DC-Free Error-Control Block Codes

Fengqin Zhai, *Student Member, IEEE*, Yan Xin, *Member, IEEE*, and Ivan J. Fair, *Member, IEEE*

***Abstract*—DC-free codes and error-control (EC) codes are widely used in digital transmission and storage systems. To improve system performance in terms of code rate, bit-error rate (BER), and low-frequency suppression, and to provide a flexible tradeoff between these parameters, this paper introduces a new class of codes with both dc-control and EC capability. The new codes integrate dc-free encoding and EC encoding, and are decoded by first applying standard EC decoding techniques prior to dc-free decoding, thereby avoiding the drawbacks that arise when dc-free decoding precedes EC decoding. The dc-free code property is introduced into standard EC codes through multimode coding techniques, at the cost of minor loss in BER performance on the additive white Gaussian noise channel, and some increase in implementation complexity, particularly at the encoder. This paper demonstrates that a wide variety of EC block codes can be integrated into this dc-free coding structure, including binary cyclic codes, binary primitive BCH codes, Reed–Solomon codes, Reed–Muller codes, and some capacity-approaching EC block codes, such as low-density parity-check codes and product codes with iterative decoding. Performance of the new dc-free EC block codes is presented.**

***Index Terms*—Complementary codeword pair, dc-free codes, error-control (EC) block codes, guided scrambling (GS), multimode coding.**

I. Introduction

ERROR-CONTROL (EC) codes and constrained sequence (CS) codes are channel codes that are used to improve the performance of digital communication systems. EC codes enable detection and/or correction of errors when the coded sequence is corrupted due to an imperfect channel. CS codes make it possible for the channel-input sequences to avoid some patterns that are susceptible to corruption in practical systems.

CS codes include runlength-limited (RLL) codes and dc-free codes, in which the coded sequences have explicit constraints in the time domain and in the frequency domain, respectively. In an RLL-coded sequence, the lengths of consecutive like-valued symbols are limited to a prescribed range. In a dc-free coded sequence, the power spectral density (PSD) function has value zero at zero frequency.

DC-free codes are commonly used in wired transmission systems to improve performance that is degraded due to the use of coupling components and/or isolating transformers [1], are widely employed in optical recording to allow low-frequency noise caused by dust and fingerprints to be filtered out with minimal loss of signal [2], and have been proposed in wireless systems to assist with the insertion of pilot tones [3].

The conventional method of incorporating both EC and CS coding into a digital communication or storage system is through concatenation of an EC code as an outer code, and a CS code as an inner code [4]. In general, CS decoders exhibit error extension, expect binary input, and output hard decisions. To avoid the impact of error extension during CS decoding on the subsequent EC decoder, and to enable the use of soft-decision information during EC decoding, it is desired that EC decoding precede CS decoding [4]. Concatenation schemes with a partially reversed order of conventional CS and EC coding have been proposed in [5] and [6], and the performance of these schemes has been evaluated in [4]. Constructions for some integrated binary dc-free EC codes have been proposed in [3] and [7]–[11]; however, these codes have limited EC ability, and/or have limited flexibility regarding tradeoffs between bit-error rate (BER) and spectral performance. For example, in [7], one specific dc-free block code with minimum distance 4 was synthesized with a lookup table. In [8], dc-free block coset codes were proposed, where construction was limited to binary dc-free coset Bose–Chaudhuri–Hocquengem (BCH) codes, and in general, the number of augmenting bits for dc control is limited by the manner in which the codeword length can be factored. It remains to be determined how this coset coding technique can be applied to nonbinary codes, product codes, and low-density parity-check (LDPC) codes in order to generate dc-free EC block codes. dc-free M-ary error-correcting codes were introduced in [3]; howeve, the spectral performance of these codes diminishes when the number of parity-check bits in the EC code is relatively large. In [9]–[11], consideration focused on dc-free convolutional codes.

This paper introduces a new general approach for integrating dc-free codes and EC block codes for highly efficient and reliable digital transmission and storage systems. The main idea is to map a source word to one or multiple complementary pairs, and to encode these pairs with an EC encoder that preserves the complementary nature of words after EC encoding. Multimode selection techniques are then used to generate a dc-balanced EC sequence. In the new coding scheme, well-established EC block codes, such as binary cyclic codes, binary primitive BCH codes, Reed–Solomon (RS) codes, Reed–Muller codes, and some capacity-approaching EC block codes, including LDPC and product codes, can be employed, and the BER performance is determined mainly by the EC codes.

The remainder of the paper is organized as follows. In Section II, we briefly review the required background. In Sec-

Paper approved by G. M. Vitetta, the Editor for Equalization and Fading Channels of the IEEE Communications Society. Manuscript received May 24, 2004; revised December 3, 2004 and April 1, 2005. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, in part by the Alberta Ingenuity Fund, and in part by iCORE. This paper was presented in part at the IEEE International Symposium on Information Theory, Yokohama, Japan, June 29-July 4, 2003, and in part at the IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing, Victoria, BC, Canada, August 2005.

The authors are with Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada (e-mail: zhai@ece.ualberta.ca; xiny@ece.ualberta.ca; fair@ece.ualberta.ca).

Digital Object Identifier 10.1109/TCOMM.2005.857160

tions III and IV, we present the proposed new coding scheme, and discuss some EC block codes that are suitable for this new scheme. In Section V, we present performance evaluation of the new coding scheme, and in Section VI, we give concluding remarks.

II. Preliminaries

A. DC-Free Codes

Let binary and bipolar symbols be from the alphabets {0,1} and $\{-1, 1\}$, respectively. Let $\{X_j\}$ denote a bipolar coded sequence. The running digital sum (RDS) of $\{X_j\}$ at the lth time instant, S_l, is defined as $S_l = \sum_{j=0}^{l} x_j$. It has been shown that if S_l is bounded, the sequence $\{X_j\}$ has a spectral null at dc [12].

DC-free codes can be classified as monomode, bimode, and multimode codes [13]. When both code rate and spectrum performance are considered, dc-free multimode codes have been shown to have advantages over the other two classes of dc-free codes [13]. In a multimode code, each source word is mapped into a set of multiple representations, and a selection criterion is employed to select the "best" representation from the selection set. In the literature, three approaches have been developed to map source words to sets of potential codewords in multimode codes: dc-free coset coding [8], guided scrambling (GS) [14], and the scrambling of an RS code [15]. GS has received particular attention, since it is easy to implement and to integrate with other codes, and is well-developed [13].

The disparity of a binary codeword is defined as the difference between the number of ones and the number of zeros in the codeword. Note that the disparity value equals the word-end RDS value evaluated over the length of the codeword. In order to ensure generation of a balanced sequence in dc-free multimode codes, it is required that there exist at least one codeword with zero disparity, or that there exist codewords with opposite polarity of disparity in each selection set. Furthermore, to improve spectrum performance, it is desirable that each selection set have as many such codewords as possible. Since a pair of complementary codewords have equal and opposite disparity, in general, in a dc-free multimode code, it is advantageous if each selection set consists of complementary codeword pairs.

In practice, it is desired that the spectral components around zero frequency in the dc-free sequence be significantly suppressed. A performance metric that indicates the width of the spectral null at low frequencies is the cutoff frequency $\omega_0 \approx 1/(2\sigma_s^2)$, where σ_s^2 represents the variance of the RDS (sum variance in brief) [16], [17]. Another performance metric, called low-frequency spectrum weight (LFSW) [18], indicates the depth of the spectral null at low frequencies. At low frequencies, $H(\omega) \approx \xi\omega^2$, where $H(\omega)$ denotes the PSD of a dc-free sequence, and ξ denotes the LFSW which equals the zero-frequency content of the continuous PSD of the corresponding RDS sequence $\{S_l\}$. In this paper, all frequency values are normalized by the symbol rate.

B. Guided Scrambling

GS encoding involves augmenting a source word with A bits, scrambling the augmented words with a scrambling polynomial $d(x)$ to form a set of 2^A quotients, and selecting an appropriate encoded word from the quotient set [14]. Its decoding involves unscrambling the received word and discarding the augmenting bits. The scrambling and unscrambling operations can be interpreted as division and multiplication processes, respectively, from the ring of polynomials over the Galois field of order two [GF(2)] [19]. Both the scrambling polynomial and selection criteria affect the power spectrum of the coded sequence.

Polynomial representations are convenient for the description of the GS coding process. Let the components of a code vector $\boldsymbol{u} = (u_{N_u-1}, u_{N_u-2}, \ldots, u_1, u_0)$ of length N_u be coefficients of a code polynomial $u(x) = u_{N_u-1}x^{N_u-1} + u_{N_u-2}x^{N_u-2} + \cdots + u_1 x + u_0$, where the term with the highest degree represents the first bit in time. We use the terms "code vector" and "code polynomial" interchangeably in this paper. In each GS encoding interval, the source word $s(x)$ of length N_s is preceded by all A-bit binary patterns $a_i(x)$, $i = 0, 1, \ldots, 2^A - 1$, to obtain augmented words $v_i(x) = a_i(x)x^{N_s} + s(x)$ of length $N_v = A + N_s$. A quotient set is obtained by scrambling these augmented words. The quotient $q_i(x)$ of length N_v corresponding to the augmented word $v_i(x)$ is

$$q_i(x) = Q_{d(x)}\left[v_i(x)x^D\right] \tag{1}$$

where the operator $Q_{d(x)}[\cdot]$ denotes the generation of a quotient through modulo-2 division of its argument by the scrambling polynomial $d(x)$ of degree D ($D \geq 1$)

$$d(x) = x^D + d_{D-1}x^{D-1} + \cdots + d_1 x + 1$$

where $d_j = 1$ or 0, and $j = 1, 2, \ldots, D-1$. Based on a given selection criterion, a quotient $q(x)$ is selected from the quotient set as a codeword to be sent to the channel. Let $v(x)$ denote the augmented word that generated the selected quotient $q(x)$.

In GS decoding, the decoded augmented word $\hat{v}(x)$ is obtained by multiplying the received word $\hat{q}(x)$, which is the hard-decision output of the demodulator, by the scrambling polynomial $d(x)$. and discarding the D least significant bits of $\hat{q}(x)d(x)$

$$\hat{v}(x) = Q_{x^D}[\hat{q}(x)d(x)]. \tag{2}$$

If no errors are present at the output of the demodulator, the source word can be recovered correctly by removing the augmenting bits from $\hat{v}(x)$. To take errors into account, let $\hat{q}(x) = q(x) + e(x)$, where $e(x)$ is an error pattern at the output of the channel, and a coefficient of 1 for the term x^j in $e(x)$ indicates an error at position j. From (2), it follows that:

$$\begin{aligned}\hat{v}(x) &= Q_{x^D}\left[(q(x) + e(x))d(x)\right] \\ &= v(x) + Q_{x^D}\left[e(x)d(x)\right]\end{aligned} \tag{3}$$

which demonstrates that at the output of the GS decoder, one error at the input of the decoder may be extended to multiple errors during GS decoding, and that this extension is upper bounded by the weight of $d(x)$. When GS is used in a noisy channel, to prevent large error extension, it is desirable to choose a scrambling polynomial with small weight. Note that the minimum weight of $d(x)$ is two.

Let $d_e(x)$ denote an even-weight scrambling polynomial with degree not greater than A, and let $d_{2,A}(x)$ denote the weight-two scrambling polynomial of degree A (i.e., $d_{2,A}(x) = x^A + 1$). It has been shown that use of $d_e(x)$ ensures the generation of complementary codeword pairs in the quotient selection set [20]. Since, in general, a scrambling polynomial with a higher degree (up to degree A) results in larger suppression of low frequencies, and scrambling polynomials of the same degree do not result in a significant difference in spectral performance [21], we propose using $d_{2,A}(x)$ in the new scheme introduced in this paper to yield good spectral performance and minimum error extension in GS decoding.

C. *Error-Control Block Codes [19]*

Let $\boldsymbol{G}$ and $\boldsymbol{H}$ denote the generator matrix and parity-check matrix for an (n, k) linear EC block code, respectively. A codeword $\boldsymbol{c}$ of length n can be generated according to $\boldsymbol{c} = \boldsymbol{u} \cdot \boldsymbol{G}$, where $\boldsymbol{u}$ is the message word of length k. Also, a codeword $\boldsymbol{c}$ satisfies $\boldsymbol{c} \cdot \boldsymbol{H}' = \boldsymbol{0}$, where $\boldsymbol{H}'$ is the transpose of $\boldsymbol{H}$.

Let α be a symbol from a Galois field GF(2^m), where m is a positive integer. Then the set $S_{\mathrm{GF}} = \{0, 1, \alpha, \alpha^2, \ldots, \alpha^{2^m-2}\}$ includes the 2^m elements of this field. Note that $\alpha^{2^m-1} = 1$. The nonzero elements can be represented by the $2^m - 1$ distinct nonzero polynomials of α over GF(2) with degree $m-1$ or less. These polynomials are obtained by setting $p(\alpha) = 0$, where $p(x)$ is a primitive polynomial of degree m over GF(2). Addition of elements is carried out using the polynomial representations of the elements.

Let β be an element in GF(2^m), and let $b(x)$ be a polynomial with coefficients from GF(2). If β is a root of $b(x)$, then β^{2^i} is also a root of $b(x)$, where i is a positive integer. The element β^{2^i} is called a conjugate of β. The elements of GF(2^m) form all the roots of $x^{2^m} + x$. β may be a root of a polynomial over GF(2) with a degree less than 2^m. Let $\phi(x)$ be the polynomial of smallest degree over GF(2), such that $\phi(\beta) = 0$. This $\phi(x)$ is called the minimal polynomial of β; it is irreducible, and it divides $x^{2^m} + x$.

III. New Coding Scheme

It has been shown that in binary multimode encoding, the existence of at least one complementary codeword pair in each codeword selection set guarantees the ability to generate a dc-free-coded sequence [14], and that use of scrambling polynomials $d_e(x)$ in GS ensures the generation of complementary GS codeword pair(s) [20]. It is also straightforward to show that it is possible to use linear EC block codes to generate complementary EC codewords when input words to the EC encoder are complementary. Note that the sum of complementary words is the all-one word, and that a linear systematic EC code which contains the all-one codeword will map the all-one input word to the all-one codeword (note that this is also possible with some nonsystematic codes). Owing to the linear nature of this EC code, it will also map complementary input words to complementary codewords. Therefore, it is sufficient for the all-one word to be an EC codeword, in order to ensure the presence of complementary EC codewords for selection when the inputs to an EC encoder are complementary words. For example, a systematic (7, 4) Hamming code has an all-one codeword. This code has eight pairs of complementary codewords, and the encoder will map a pair of complementary input words to a pair of complementary codewords.

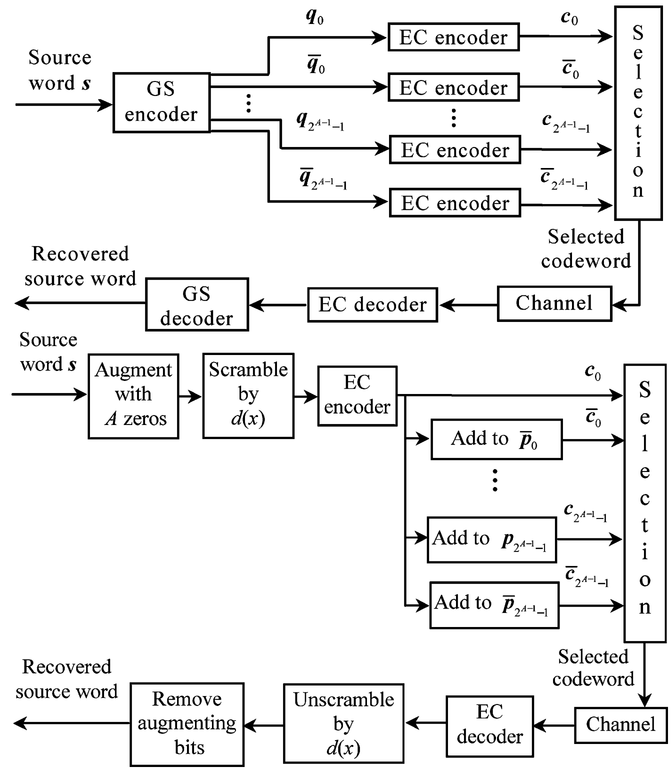

Fig. 1. (a). Block diagram of the new dc-free EC block code. (b). Block diagram of the equivalent new dc-free EC block code.

We consider integration of a GS encoder, which uses A ($A \geq 1$) augmenting bits and $d_{2,A}(x)$, with an EC block code which contains the all-one codeword. Fig. 1(a) shows our new coding structure for dc-free EC block codes. To encode, a source word $\boldsymbol{s}$ is mapped to a set of 2^{A-1} pairs of complementary EC codewords through concatenation of a GS encoder with 2^A identical EC encoders. The GS encoder generates 2^{A-1} pairs of complementary words $\boldsymbol{q}_j$ and $\overline{\boldsymbol{q}}_j$, $j = 0, 1, \ldots, 2^{A-1} - 1$, and the EC encoders encode these complementary words to form the complementary EC words $\boldsymbol{c}_j$ and $\overline{\boldsymbol{c}}_j$, $j = 0, 1, \ldots, 2^{A-1} - 1$. Based on a predetermined selection criterion, an EC codeword from the selection set is selected to ensure that the RDS of the coded sequence is bounded. Therefore, the input to the channel is a fully EC-protected dc-free sequence.

Several selection criteria have been developed for multimode codes. They include minimum word-end running digital sum (MRDS), which selects the word from the selection set which results in the minimum absolute word-end RDS value [14], and minimum squared weight (MSW), which selects the word with the minimum sum of the squared RDS values at each bit position within the word [17]. It has been shown that the MSW criterion yields excellent spectral performance for dc-free multimode codes when $A > 1$, and provides approximately the same performance as that of MRDS when $A = 1$ [13]. Since MRDS is simpler than MSW, in this paper, we consider use of MRDS when $A = 1$ and MSW when $A > 1$.

It can be shown that the all-one binary codeword exists in most linear EC block codes, including binary cyclic codes, binary primitive BCH codes, the Golay code, RS codes, and Reed–Muller codes. For LDPC codes [22], if each row of the parity-check matrix has even weight, the all-one word is a codeword. Furthermore, product codes [23] with the above codes as component codes also have the all-one codeword. We consider appropriate EC codes in more detail in Section IV.

Note that the use of EC codes that map the all-one input word to the all-one codeword is not strictly necessary in order to ensure valid operation of this new encoding approach. In general, in order to ensure that the RDS of the coded sequence can be bounded, we must ensure that in each selection set, there exist words of opposing disparity, or at least one word with zero disparity. However, it is not clear how to construct a dc-free EC code with this general requirement. Instead of considering this general construction of dc-free EC codes, we propose selecting GS and EC code parameters, as outlined above, to ensure the presence of complementary words in each selection set. It is well known how to generate complementary words at the output of the GS encoder through use of $d_e(x)$, and that an EC encoder with an all-one codeword will preserve this complementary nature. Alternatively, if the EC encoder maps some input word $\boldsymbol{h}$ rather than the all-one word to the all-one codeword, the GS encoder could be designed to guarantee that words in the GS selection set are related through elementwise modulo-2 addition with $\boldsymbol{h}$. This would involve construction of the appropriate GS scrambling polynomial whose weight would be greater than two [24].

An equivalent form of the new coding scheme is shown in Fig. 1(b). The difference between these two structures is the manner in which encoding is implemented. Since both the GS encoder and the EC encoders in Fig. 1(a) are linear, there are predetermined additive patterns between the codewords $\{\boldsymbol{c}_0, \overline{\boldsymbol{c}}_0, \ldots, \boldsymbol{c}_{2^{A-1}-1}, \overline{\boldsymbol{c}}_{2^{A-1}-1}\}$ which depend only on the GS scrambling polynomial and the EC encoder. In order to determine these additive patterns, one can augment the all-zero source word with all patterns of the augmenting bits $\boldsymbol{a}_i, i = 0, 1, \ldots, 2^A - 1$, scramble these augmented words with the scrambling polynomial, and encode these scrambled words to EC codewords $\{\boldsymbol{p}_0, \overline{\boldsymbol{p}}_0, \ldots, \boldsymbol{p}_{2^{A-1}-1}, \overline{\boldsymbol{p}}_{2^{A-1}-1}\}$. Then the codeword candidates $\{\boldsymbol{c}_0, \overline{\boldsymbol{c}}_0, \ldots, \boldsymbol{c}_{2^{A-1}-1}, \overline{\boldsymbol{c}}_{2^{A-1}-1}\}$ in each encoding interval are related by $\boldsymbol{c}_j = \boldsymbol{c}_0 + \boldsymbol{p}_j$ and $\overline{\boldsymbol{c}}_j = \boldsymbol{c}_0 + \overline{\boldsymbol{p}}_j$, $j = 0, 1, \ldots, 2^{A-1} - 1$. Fig. 1(b) demonstrates that the same codeword set as that generated in Fig. 1(a) can be constructed by generating $\boldsymbol{c}_0$, and then forming the rest of the codeword alternatives through addition with the predetermined additive patterns stored within the encoder. Note that since $\boldsymbol{p}_0$ is the all-zero sequence, it is not indicated in Fig. 1(b), and that since the predetermined additive patterns are complementary, only half of them need to be stored.

Fig. 1(a) and (b) show that decoding of the new coding scheme is completed first through EC decoding and then through GS decoding, and that the EC decoding is independent of the GS decoding. Therefore, 1) soft-decision information available at the output of the demodulator can be used by the EC decoder, resulting in improved BER performance, and 2) error extension that occurs during GS decoding follows the EC decoder, and therefore, has no detrimental impact on the performance of the EC decoder.

The BER performance of this new technique is governed largely by the characteristics of the EC code. When $d_{2,A}(x) = x^A + 1$ is used for GS, subsequent error extension in GS decoding is upper bounded by two.

IV. Error-Control Block Codes Suitable for the New Coding Scheme

Let $w_{n-1}(x)$ denote the all-one word of length n. If $w_{n-1}(x)$ is a codeword, nonsystematic codes, in general, map a nonall-one input word to $w_{n-1}(x)$. The $d(x)$ constructed by using this nonall-one word as the relationship sequence likely has weight greater than two. In systematic codes, however, the all-one input word is mapped to $w_{n-1}(x)$, and $d(x) = x^A + 1$ will generate this all-one relationship sequence. Since low weight $d(x)$ is preferred to decrease the error extension during GS decoding, systematic EC codes are considered in the remainder of this paper.

We now show the existence of $w_{n-1}(x)$ in a number of different EC block codes, in order to demonstrate that these EC codes can be integrated into our new dc-free EC block coding scheme.

A. Binary Cyclic Codes

In an (n, k) binary cyclic code, a word $c(x)$ is a codeword if and only if $c(x)$ is divisible by the generator polynomial $g_C(x)$ of degree $(n-k)$ [19]. Also, $g_C(x)$ divides $x^n + 1$. It is straightforward to verify that $x^n + 1 = (x+1)w_{n-1}(x)$. If $g_C(x)$ is selected from factors of $x^n + 1$ except $x + 1$ (which is often the case), the all-one word is a codeword.

B. Binary Primitive BCH Codes

BCH codes are cyclic codes. For an (n, k) binary primitive t-error-correcting BCH code, the generator polynomial $g_{\mathrm{BCH}}(x)$ is the least common multiple of the minimal polynomial of α^i, $i = 1, 2, \ldots, 2t$. Since $w_{n-1}(x)$ is a multiple of all the minimum polynomials except for those of elements zero and one [19], $w_{n-1}(x)$ is a multiple of $g_{\mathrm{BCH}}(x)$, and is, therefore, a codeword.

For shortened BCH codes, unfortunately, the all-one codeword is not available in BCH codes shortened in the conventional manner [19]. Instead, we develop a new approach to shorten the code. Consider the codewords in the (n, k) code that have all-one parity check of length $n - k$. Note that if $2k > n$, there are $2^k/2^{n-k} = 2^{2k-n}$ of them. These codewords are called shortening patterns. For example, $\boldsymbol{c}_s = 00110111111, 1111$ is such a codeword in the (15, 11) BCH code ($\boldsymbol{g}_{\mathrm{BCH}} = 11001$). This codeword indicates that the (15, 11) code can be shortened to a (12, 8) code in a different manner. Zeros in $\boldsymbol{c}_s$ indicate the positions in which data bits in all words must be ensured to be zero. To construct the codewords, insert three zeros in the 8-b message word $\boldsymbol{u} = u_7u_6u_5u_4u_3u_2u_1u_0$ to obtain the 11-b word $\tilde{\boldsymbol{u}} = 00u_7u_60u_5u_4u_3u_2u_1u_0$. This word $\tilde{\boldsymbol{u}}$ is processed as the input word of the (15, 11) encoder, and the three zeros are removed after encoding to create a (12, 8) shortened code.

Based on the shortening pattern, it is straightforward to construct the shortened GS-BCH scheme by modifying Fig. 1(a) or

(b) to include appropriate insertion of zeros before the BCH encoder and removal of these zeros after the BCH encoder, and insertion of zeros (or appropriate soft-decision values) before the BCH decoder and removal of these zeros after the BCH decoder.

Shortening patterns can be constructed by simply adding appropriately left-shifted versions of $\boldsymbol{g}_{\mathrm{BCH}}$, which is equivalent to evaluating a multiple of $\boldsymbol{g}_{\mathrm{BCH}}$, and therefore, constructing a valid codeword in the code. Alternatively, to find the shortening pattern for the (12, 8) code, appending the all-zero vector of length 10 to $\boldsymbol{g}_{\mathrm{BCH}}$ yields the codeword $\overline{\boldsymbol{c}}_s$. The complement of $\overline{\boldsymbol{c}}_s$, the desired pattern $\boldsymbol{c}_s$, is also a codeword, because the all-one word of length n is a codeword. Based on these approaches, the (15, 11) code can be shortened to $(15 - z, 11 - z)$ codes, $z = 3, 4, \ldots, 10$.

The most typical extension of a binary code is the inclusion of an overall parity check, which is the modulo-2 sum of all other bits in the word. Since the codeword length n is odd in BCH codes [19], it is straightforward to verify that the all-one codeword exists in extended BCH codes.

C. Reed–Solomon Codes

RS codes are cyclic codes. For (n, k) primitive t-error-correcting RS codes with $k = n - 2t$ and $n = 2^m - 1$, the generator polynomial $g_{\mathrm{RS}}(x)$ has $\alpha, \alpha^2, \alpha^3, \ldots, \alpha^{2t}$ as all its roots. Since α^j from GF(2^m), $1 \le j \le n-1$, form all the roots of $w_{n-1}(x)$, $w_{n-1}(x)$ is a multiple of $g_{\mathrm{RS}}(x)$. However, the binary representation of $w_{n-1}(x)$ is not all-one when $m > 1$. Let α^γ $(m < \gamma \le 2^m - 2)$ from GF(2^m) be the element whose binary vector representation is an m-bit all-one vector. Since in a linear code over GF(2^m), the product of any codeword by a field element is a codeword [25], in RS codes, there exists $w'_{n-1}(x) = \alpha^\gamma w_{n-1}(x)$, whose corresponding binary representation is the all-one word.

As with shortened BCH codes, the all-one codeword is not available in RS codes shortened in the usual manner. The method used for BCH codes can be employed to find shortening patterns for RS codes, however, it is not as straightforward because of the nonbinary elements. As an example, Fig. 2 shows how to obtain the shortening patterns for codes shortened from a (7, 5) RS code.

From Fig. 2, the shortening pattern for the (3, 1) code, when written with coefficient 1 instead of α^5 and omitting the first three zeros, is $x^3 + x + 1$, which is the $g_{\mathrm{BCH}}(x)$ of the (7, 4) binary BCH code. This observation provides an easier approach to construct the shortening patterns for RS codes based on $g_{\mathrm{BCH}}(x)$, which are given in many EC text books. Consider binary (n, k_B) BCH codes and (n, k_R) RS codes, such that both codes are constructed over GF(2^m) and have the same t. Since $g_{\mathrm{BCH}}(x)$ has $\alpha, \alpha^2, \ldots, \alpha^{2t}$ and their conjugates as all its roots, and $g_{\mathrm{RS}}(x)$ has $\alpha, \alpha^2, \ldots, \alpha^{2t}$ as all its roots [19], $g_{\mathrm{BCH}}(x)$ is a multiple of $g_{\mathrm{RS}}(x)$ and is a codeword of the RS code. Therefore, the shortening patterns for the RS codes can be obtained through summation of shifted versions of the corresponding $g_{\mathrm{BCH}}(x)$, and subsequent replacement of the coefficients of value 1 by α^γ.

		x^6	x^5	x^4	x^3	x^2	x	1	
$g_{RS}(x)$						1	α^4	α^3	
$\alpha^3 x g_{RS}(x)$	$+$				α^3	α^7	α^6		
$(1+\alpha^3 x) g_{RS}(x) = c_0(x)$					α^3	0	α^3	α^3	
$\alpha^2 c_0(x) = c_{s1}(x)$		0	0	0	α^5	0	α^5	α^5	✓
$\alpha^2 c_{s1}(x) = c_2(x)$					1	0	1	1	
$x^2 c_2(x)$	$+$		1	0	1	1			
$(1+x^2) c_2(x) = c_3(x)$			1	0	0	1	1	1	
$\alpha^5 c_3(x) = c_{s2}(x)$		0	α^5	0	0	α^5	α^5	α^5	✓

Fig. 2. Construction of the shortening patterns $c_{s1}(x)$ and $c_{s2}(x)$ for a (7, 5) RS code shortened to (3, 1) and (4, 2) RS codes, respectively.

D. Reed–Muller Codes

It follows from observation of the generator matrix of binary rth-order Reed–Muller codes, which are nonsystematic, that the all-one codeword exists [25]. The codes can be converted to a systematic form through elementary row operations and/or column permutations on the generator matrix [25]. If there exists the all-one codeword, it remains a codeword after any of the above transformations of the generator matrix.

E. LDPC Codes

Length-n LDPC codes are specified by the sparse nonsystematic parity-check matrix $\boldsymbol{H}$ [22]. For regular LDPC codes, there is a fixed row weight w_r and fixed column weight w_c. These codes are usually denoted as (n, w_c, w_r). If all rows of $\boldsymbol{H}$ have even weight, the all-one word is a codeword. As an example, all rows of $\boldsymbol{H}$ in the (12, 3, 6) LDPC code constructed in [26] have even weight, and it is straightforward to verify that the all-one word is a codeword.

F. Product Codes

An $(n_1 n_2, k_1 k_2)$ product code is constructed using two component codes, an (n_1, k_1) inner block code and an (n_2, k_2) outer block code [19]. If all-one binary codewords exist for both component codes, it can be shown that an all-one message word of length $k_1 k_2$ results in the all-one codeword of length $n_1 n_2$. All codes discussed above can be used as component codes in these product codes.

V. Performance Evaluation

In this section, we evaluate the performance of our new scheme, and compare it with dc-free coset coding and the conventional approach of concatenating EC and constrained codes. Note that for the simulation results reported in this section, independent and equiprobable information bits are assumed unless otherwise specified.

Due to the complexity of GS dc-free multimode codes, it is not yet known how to fully analyze their dc-free performance in a practical manner [13]. Instead, the spectral performance of these codes is obtained through simulation or approximations. In both cases, metrics such as sum variance and LFSW are used to assess the performance of these codes. A random-drawing

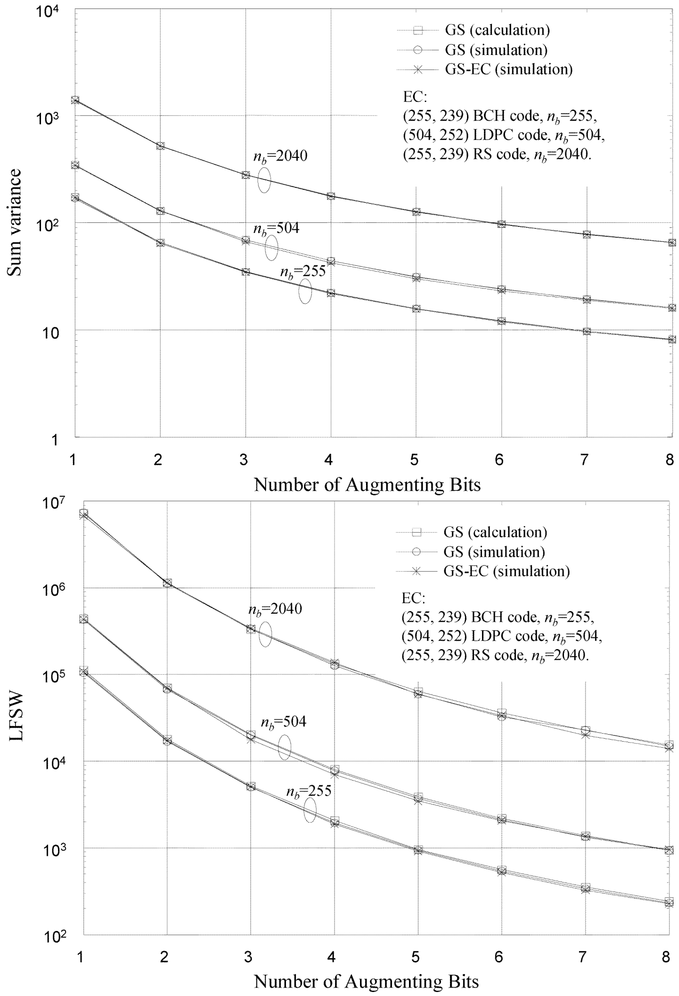

Fig. 3. (a) Sum variance of GS dc-free multimode codes (calculation and simulation), and simulated sum variance of three different dc-free GS-EC codes. (b) LFSW of GS dc-free multimode codes (calculation and simulation), and simulated LFSW of three different dc-free GS-EC codes.

model was proposed to estimate the sum variance of multimode codes which used the MRDS selection criterion [13], and a method has been developed to estimate the sum variance and the LFSW of GS dc-free multimode codes which use $d_{2,A}(x)$ and the MSW criterion [21].

As described in [21], the sum variance σ_s^2 and LFSW ξ of GS dc-free multimode codes which use the MSW criterion and $d_{2,A}(x)$ can be approximated as

$$\sigma_s^2 \approx 0.2326 \cdot S_A \cdot \frac{n_b}{A} \tag{4}$$

$$\xi \approx 0.2225 \cdot L_A \cdot \left(\frac{n_b}{A}\right)^2 \tag{5}$$

where n_b is the binary codeword length and S_A and L_A are factors which are dependent only on the number of augmenting bits A. Values of S_A and L_A are reported in [21].

Fig. 3(a) and (b) compare the sum variance and LFSW calculated from (4) and (5) with simulated sum variance and LFSW for GS dc-free multimode codes. These figures also show simulated sum variance and LFSW for our new dc-free GS-EC block codes, where the (255, 239) BCH code, (504, 252) LDPC code, and (255, 239) RS codes are used as EC codes. These figures show that relations (4) and (5) can also be applied to our GS-EC block codes to accurately estimate sum variance and LFSW values.

In [8], a dc-free coset code is constructed from a binary BCH code through modification of the nonsystematic generator matrix of the BCH code. We call this approach a dc-free coset BCH code. For a binary t-error-correcting BCH code of length $n = 2^m - 1$, let A_c and n_p be two odd integers such that $A_c \cdot n_p = n$ and $n_p \geq 2t + 1$. As described in [8], the dc-free coset BCH code of length n can be generated with A_c augmenting bits. Due to the similarity of the approach to dc control in the dc-free coset BCH code [8] of length n and a polarity switch code of length n_p, we propose estimating the sum variance of the dc-free coset BCH code with the analytical method developed for the polarity-switch code [17]. By extending the approach outlined in [17], the sum variance of the dc-free coset BCH code can be approximated as

$$\sigma_s^2 \approx \frac{(2 \cdot n_p - 1)}{3}. \tag{6}$$

Also, based on a closed-form expression for the approximate PSD of the polarity-switch code [26], it follows that the LFSW ξ of the polarity-switch code can be approximated as

$$\xi \approx 1.03 n_p^2 - 1.35 n_p - 0.08. \tag{7}$$

Therefore, the LFSW of the dc-free coset BCH code can be estimated from (7).

In order to compare the performance of our GS-BCH code with the dc-free coset BCH code, consider the (255, 239) BCH code with $\boldsymbol{g}_{\mathrm{BCH}} = 267\,543$ (in octal) [19] and $A = A_c = 5$ augmenting bits in both the GS-BCH code and the dc-free coset BCH code; in this coset BCH code, $n_p = 51$. Note that the BCH code used in our GS-BCH code is a standard systematic BCH code, and the code used in the dc-free coset BCH code is a nonsystematic modified BCH code [8]. From (4) through (7), noting that $S_A = 1.3367$ and $L_A = 1.7079$ when $A = 5$ [21], we obtain $\sigma_s^2 \approx 15.86$ and $\xi \approx 988.40$ for the GS-BCH code, and $\sigma_s^2 \approx 33.67$ and $\xi \approx 2610.10$ for the dc-free coset BCH code. Our simulated σ_s^2 and ξ values equal 15.58 and 912.15 for the GS-BCH code, and 33.67 and 2905.22 for dc-free coset BCH code, respectively.

Fig. 4 shows simulated PSDs for both these codes. These results demonstrate that with the same number of augmenting bits, the GS-BCH code results in significant improvement in spectral performance, compared with the dc-free coset BCH code. Fig. 5 illustrates the simulated distribution of RDS values for these two codes. From [8], the RDS of this dc-free coset BCH code can be upper bounded by $|\mathrm{RDS}| \leq 51 + \lfloor 51/2 \rfloor = 76$. Fig. 5 shows that for this code, RDS values with probability 10^{-6} are still far from this bound. We also note that if the Berlekamp–Massey algorithm [19] is used to decode the dc-free coset BCH code, error extension cannot be avoided, since the source word is recovered from the decoded codeword by solving equations based on the nonsystematic generator matrix.

In the remainder of this section, we show other advantages of our new coding scheme. We report performance results when an RS code is used as the component EC code to show that our new coding scheme yields coding gain when compared with the conventional concatenation of RS and GS codes. We also present results when an LDPC code is used as the component

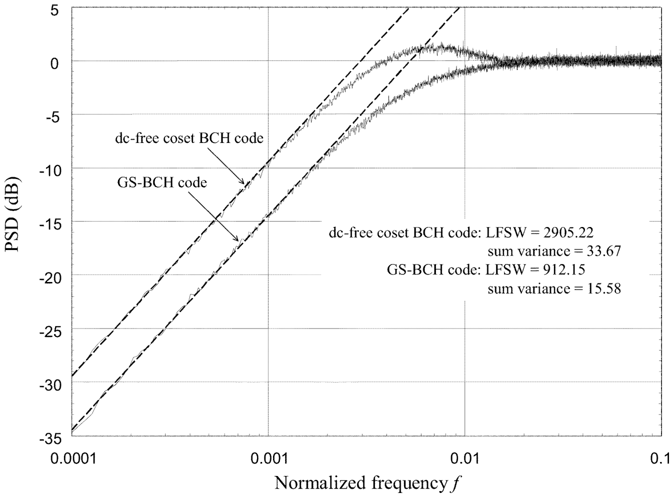

Fig. 4. Simulated PSDs (solid curves) and approximate PSDs (dashed lines, based on the simulated LFSW) of the dc-free coset BCH code and the GS-BCH code employing the (255, 239) BCH code with $A = A_c = 5$.

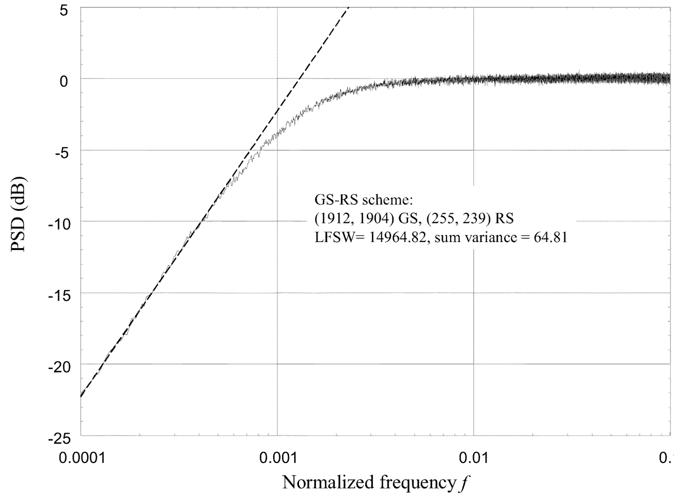

Fig. 6. Simulated PSD (solid curve) and approximate PSD (dashed line, based on the simulated LFSW) for a GS-RS coding scheme.

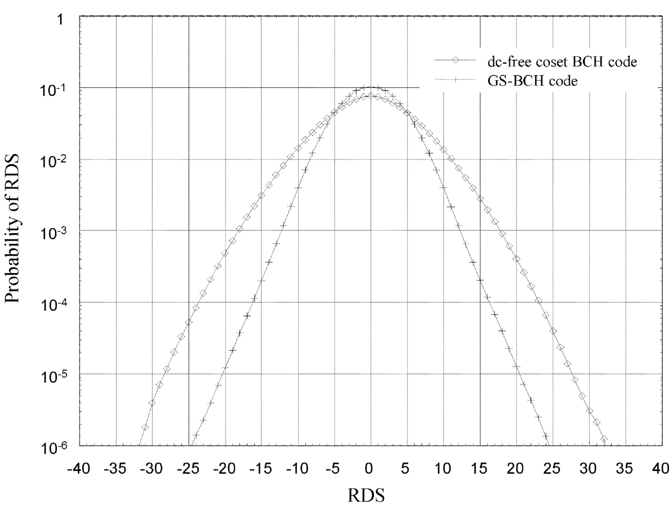

Fig. 5. Simulated distribution of RDS values for the dc-free coset BCH code and the GS-BCH code employing the (255, 239) BCH code with $A = A_c = 5$.

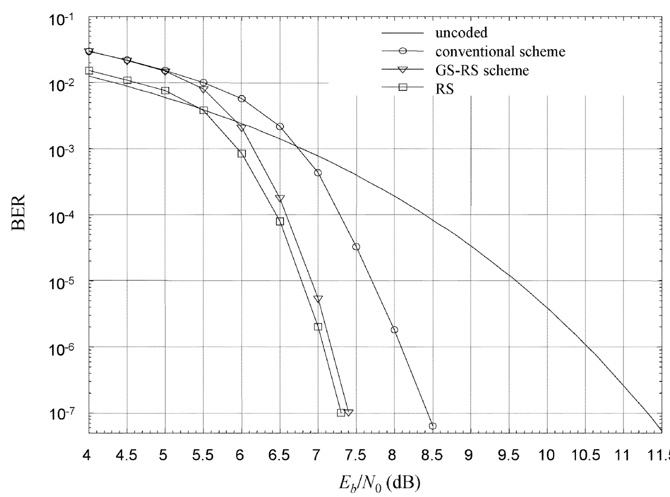

Fig. 7. BER performance over the AWGN channel.

EC code to illustrate integration of soft-decision EC decoding, which is not possible, in general, in the conventional scheme. Note that in the following results, we use $d_{2,A}(x)$ and the MSW selection criterion in our codes.

The performance of the proposed new GS and RS integrated coding scheme (GS-RS scheme) is shown in Figs. 6–8. Our simulation parameters include: (1912, 1904) GS code with $A = 8$; (255, 239) RS code with $\boldsymbol{p} = 435$ (in octal) for generating GF(2^8). For comparison, we also give the BER performance for uncoded signaling for (255, 239) RS coding, and for conventional concatenation of RS and GS codes. In this conventional code, the (255, 239) RS code was used as the outer code, and a (2048, 2040) GS code was used as the inner code.

Fig. 6 depicts the PSD performance of the GS-RS scheme. The corresponding conventional scheme provides almost the same low-frequency suppression as that of the new scheme; RS coding alone does not result in suppression of low frequencies.

Fig. 7 presents the BER performance of this new GS-RS scheme over the additive white Gaussian noise (AWGN) channel, and compares this with the performance of uncoded signaling, the conventional scheme, and RS coding without spectrum control. E_b denotes the average energy per information bit, N_0 is the single-sided PSD of the white Gaussian noise, and matched filtering is used. RS decoding is implemented using the hard-decision Berlekamp–Massey algorithm [19]. As indicated in this figure, under these conditions, our GS-RS scheme offers 1-dB gain at BER $= 10^{-7}$ when compared with conventional concatenation of RS and GS codes.

In order to obtain a general indication of error performance that can be expected on a channel with low-frequency constraints, we model the noisy dc-constrained channel using a simple first-order RC high-pass filter (HPF) and AWGN (HP-AWGN channel). The HPF can be used to model a specific dc constraint through selection of the value of the normalized time constant $\tau = RC/T$, where T denotes symbol duration.

Fig. 8 illustrates the BER performance of our new GS-RS coding scheme compared with the other three schemes for different source-logic probabilities when $\tau = 120$ and a rectangular pulse shape and a rectangular receiving filter are used. This figure presents results when the probability of a logic 0 in the source sequence, p_0, is 0.5, 0.45, and 0.4. It can be seen that the

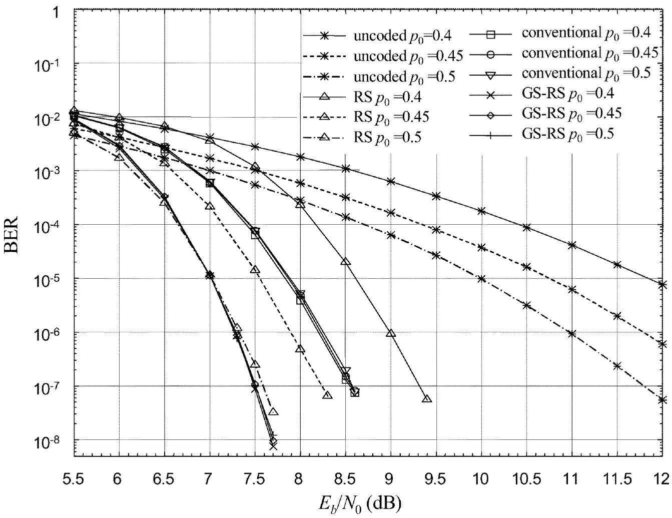

Fig. 8. BER performance with different source-symbol probabilities over the HP-AWGN channel.

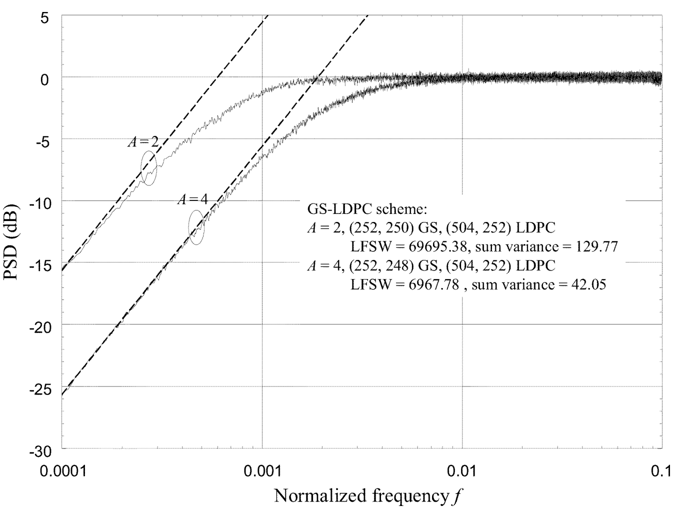

Fig. 9. Simulated PSDs (solid curves) and approximate PSDs (dashed lines, based on the simulated LFSW) of two GS-LDPC codes.

change of this probability does not affect the performance of the GS-RS scheme and the conventional scheme significantly, but has a large, negative effect on the performance of the uncoded and RS codes, which justifies the use of dc-constrained codes on this channel. While these results are for the simple first-order HP-AWGN channel model, it is expected that these trends would also be apparent in more realistic noisy dc-free constrained systems.

The performance of the proposed new GS and LDPC integrated coding scheme (GS-LDPC) for $A = 2$ and $A = 4$, in terms of PSD and BER, is shown in Figs. 9 and 10. The rate-1/2 LDPC code uses a parity-check matrix $\boldsymbol{H}$ from [28] with column weight 3 and row weight 6, and LDPC decoding is performed using the iterative sum-product algorithm [22] with a maximum of 1000 iterations.

Fig. 9 illustrates the PSDs for this GS-LDPC scheme with $A = 2$ and $A = 4$. Clearly, a 10-dB improvement in performance is obtained at $f = 10^{-4}$ when $A = 4$, compared with the case with $A = 2$. This is in agreement with the results obtained with increasing A in GS dc-free multimode codes without provision for EC [13]. Fig. 10 shows the BER performance in the AWGN channel for the GS-LDPC scheme with $A = 2$ and $A = 4$, and the BER performance for LDPC codes without the dc constraint. It is evident that when the BER is less than 10^{-5}, the BER performance of the new GS-LDPC scheme is within 0.15–0.2 dB of that of the LDPC code without the dc constraint. This loss in performance in the AWGN channel is due to the error extension in GS decoding and the rate penalty associated with including GS augmenting bits. As with the GS-RS scheme, however, the performance of the GS-LDPC scheme will be superior on channels with a dc constraint.

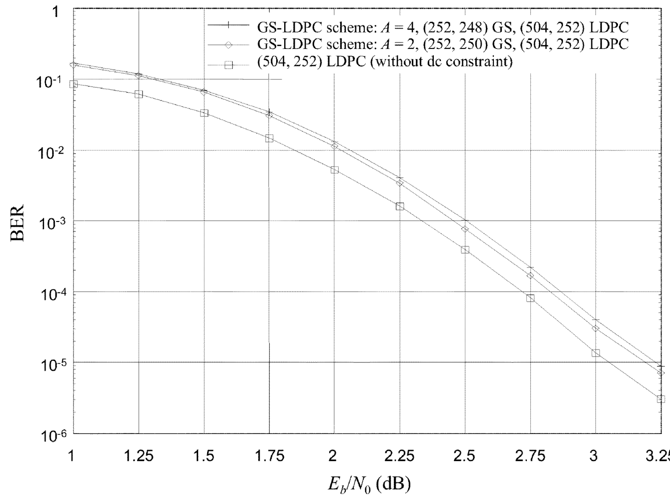

Fig. 10. BER performance of the GS-LDPC coding scheme over the AWGN channel.

VI. Conclusion

A new general technique for integrating dc-free CS codes and EC block codes has been reported. Our new coding method proposes concatenation of GS multimode coding and EC block coding. In this new scheme, the codeword selection set consists of EC block codewords. Selection criteria previously developed for dc-free multimode codes can be used. Well-established standard EC block decoders can be employed to complete the first stage of decoding of these new codes, allowing for the use of soft-decision information and iterative decoding. Many well-known block codes can be used in this new coding scheme.

We have introduced a method for estimating the spectral performance of the new codes, and have shown that this new coding scheme demonstrates excellent performance in terms of low-frequency suppression. We have also demonstrated that, compared with conventional concatenation of EC and dc-free codes, this scheme provides superior BER performance over a simple HP-AWGN channel.

Acknowledgment

The authors would like to thank the anonymous reviewers, whose comments resulted in significant improvement in this paper.

REFERENCES

[1] K. W. Cattermole, "Principles of digital line coding," *Int. J. Electron.*, vol. 55, pp. 3–33, Jul. 1983.

[2] K. A. S. Immink, P. H. Siegel, and J. K. Wolf, "Codes for digital recorders," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2260–2299, Oct. 1998.

[3] A. Kokkos, A. Popplewell, and J. J. O'Reilly, "A power efficient coding scheme for low-frequency spectral suppression," *IEEE Trans. Commun.*, vol. 41, no. 11, pp. 1598–1601, Nov. 1993.

[4] J. L. Fan and A. R. Calderbank, "A modified concatenated coding scheme with applications to magnetic data storage," *IEEE Trans. Inf. Theory*, vol. 44, no. 4, pp. 1565–1574, Jul. 1998.

[5] W. G. Bliss, "Circuitry for performance error correction calculations on baseband encoded data to eliminate error propagation," *IBM Tech. Discl. Bull.*, vol. 23, pp. 4633–4634, 1981.

[6] K. A. S. Immink, "A practical method for approaching the channel capacity of constrained channels," *IEEE Trans. Inf. Theory*, vol. 43, no. 5, pp. 1389–1399, Sep. 1997.

[7] H. C. Ferreira, "Lower bounds on the minimum Hamming distance achievable with runlength constrained or dc-free block codes and the synthesis of a (16, 8) $d = 4$ dc-free block code," *IEEE Trans. Magn.*, vol. MAG-20, no. 9, pp. 881–883, Sep. 1984.

[8] R. H. Deng and M. A. Herro, "DC-free coset codes," *IEEE Trans. Inf. Theory*, vol. 34, no. 4, pp. 786–792, Jul. 1988.

[9] M. C. Chiu, "DC-free error-correcting codes based on convolutional codes," *IEEE Trans. Commun.*, vol. 49, no. 4, pp. 609–619, Apr. 2001.

[10] T. Wadayama and A. J. H. Vinck, "DC-free binary convolutional coding," *IEEE Trans. Inf. Theory*, vol. 48, no. 1, pp. 162–173, Jan. 2002.

[11] I. J. Fair and D. R. Bull, "DC-free error control coding through guided convolutional coding," in *Proc. IEEE Int. Symp. Inf. Theory*, 2002, p. 297.

[12] G. L. Pierobon, "Codes for zero spectral density at zero frequency," *IEEE Trans. Inf. Theory*, vol. IT-30, no. 2, pp. 435–439, Mar. 1984.

[13] K. A. S. Immink and L. Patrovics, "Performance assessment of dc-free multimode codes," *IEEE Trans. Commun.*, vol. 45, no. 3, pp. 293–299, Mar. 1997.

[14] I. J. Fair, W. D. Grover, W. A. Krzymien, and R. I. MacDonald, "Guided scrambling: A new line coding technique for high-bit-rate fiber optic transmission systems," *IEEE Trans. Commun.*, vol. 39, no. 2, pp. 289–297, Feb. 1991.

[15] A. Kunisa, S. Takahasi, and N. Itoh, "Digital modulation method for recordable digital video disc," *IEEE Trans. Consum. Electron.*, vol. 42, no. 3, pp. 820–825, Aug. 1996.

[16] J. Justesen, "Information rates and power spectra of digital codes," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 3, pp. 457–472, May 1982.

[17] K. A. S. Immink, *Codes for Mass Data Storage Systems*. Rotterdam, The Netherlands: Shannon Foundation, 1999.

[18] Y. Xin and I. J. Fair, "A performance metric for codes with a high-order spectral null at zero frequency," *IEEE Trans. Inf. Theory*, vol. 50, no. 2, pp. 385–394, Feb. 2004.

[19] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[20] Y. Xin and I. J. Fair, "Polynomials for generating selection sets with complementary quotients in guided scrambling line coding," *Electron. Lett.*, vol. 37, pp. 365–366, Mar. 2001.

[21] ——, "Low-frequency performance of guided scrambling dc-free codes," *IEEE Commun. Lett.*, vol. 9, no. 6, pp. 537–539, Jun. 2005.

[22] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[23] R. M. Pyndiah, "Near-optimum decoding of product codes: Block turbo codes," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1003–1010, Aug. 1998.

[24] I. J. Fair, Q. Wang, and V. K. Bhargava, "Polynomials for guided scrambling line codes," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 4, pp. 499–509, Apr. 1995.

[25] R. E. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley, 1983.

[26] P. Sweeney, *Error Control Coding: From Theory to Practice*. Chichester, U.K.: Wiley, 2002.

[27] L. J. Greenstein, "Spectrum of a binary signal block coded for dc suppression," *Bell Syst. Tech. J.*, vol. 53, pp. 1103–1126, Jul.–Aug. 1974.

[28] D. J. C. MacKay. Online code resources of low-density parity-check codes. [Online]. Available: http://www.inference.phy.cam.ac.uk/mackay/codes/data.html

Fengqin Zhai (S'98) was born in Yangquan, Shanxi, China. She received the M.Sc. degree in electrical and computer engineering in 2001 from the University of Alberta, Edmonton, AB, Canada, where she is currently working toward the Ph.D. degree.

Her current research interests include error-control coding and constrained coding in wireless communication, digital transmission, and data-storage systems.

Yan Xin (S'97–M'03) was born in Shenyang, Liaoning, China. He received the Ph.D. degree in electrical and computer engineering from the University of Alberta, Edmonton, AB, Canada, in 2002.

Since 2002, he has been a Postdoctoral Fellow, holding an Alberta Ingenuity Associateship at the University of Alberta. His current research interests include error-control coding and constrained coding in digital transmission, data storage, and multicarrier communications systems.

Ivan J. Fair (S'91–M'95) received the B.Sc. and M.Sc. degrees from the University of Alberta, Edmonton, AB, Canada, in 1985 and 1989, respectively, and the Ph.D. degree from the University of Victoria, Victoria, BC, Canada, in 1995.

He was employed with Bell Northern Research Ltd. (now Nortel Networks) from 1985 to 1987, and with MPR TelTech Ltd. from 1989 to 1991, working on various aspects of communication system design and implementation. In 1995, he joined the Technical University of Nova Scotia (since amalgamated with Dalhousie University), Halifax, NS, Canada, as an Assistant Professor, and was promoted to Associate Professor before joining the University of Alberta in 1998, where he is now a Professor. From 2001 to 2004, he was the Associate Chair for Undergraduate Studies and Acting Director of Computer Engineering in the Department of Electrical and Computer Engineering at the University of Alberta. He is presently an Associate Editor for IEEE COMMUNICATIONS LETTERS.