RESEARCH **Open Access**

Ordered statistics-based list decoding techniques for linear binary block codes

Saif EA Alnawayseh[1] and Pavel Loskot[2*]

Abstract

The ordered statistics-based list decoding techniques for linear binary block codes of small to medium block length are investigated. The construction of a list of the test error patterns is considered. The original ordered-statistics decoding (OSD) is generalized by assuming segmentation of the most reliable independent positions (MRIPs) of the received bits. The segmentation is shown to overcome several drawbacks of the original OSD. The complexity of the ordered statistics-based decoding is further reduced by assuming a partial ordering of the received bits in order to avoid the complex Gauss elimination. The probability of the test error patterns in the decoding list is derived. The bit error rate performance and the decoding complexity trade-off of the proposed decoding algorithms is studied by computer simulations. Numerical examples show that, in some cases, the proposed decoding schemes are superior to the original OSD in terms of both the bit error rate performance as well as the decoding complexity.

Keywords: Decoding, Fading, Linear code, Performance evaluation

Introduction

A major difficulty in employing forward error correction (FEC) coding is the implementation complexity especially of the decoding at the receiver, and the associated decoding latency for long codewords. Correspondingly, the FEC coding is often designed to trade-off the bit error rate (BER) with the decoding complexity and latency [1], and even with the energy efficiency [2]. Many universal decoding algorithms have been proposed for the decoding of linear binary block codes [3]. The decoding algorithms in [4,5] are based on the testing and re-encoding of the information bits as initially considered by Dorsch [6]. In particular, a list of the likely transmitted codewords is generated using the reliabilities of the received bits, and then, the most likely codeword is selected from this list. The list of the likely transmitted codewords can be constructed from a set of the test error patterns. The test error patterns can be predefined as in [4,7], and as assumed also in this article, predefined and optimized for the channel statistics as in [8], or defined adaptively for a particular received sequence as suggested in [9]. The complexity of

the list decoding can be further reduced by the skipping and stopping rules as shown, for example, in [4,7].

Among numerous variants of the list decoding techniques, the ordered-statistics decoding (OSD) is well known [4,7]. The structural properties of the FEC code are utilized to reduce the OSD complexity in [10]. The achievable coding gain of the OSD is improved by considering the multiple information sets in [11]. The decoding proposed in [12] exploits randomly generated biases to present the decoder with the multiple received soft-decision values. The sort and match decoding of [13] divides the received sequence into two disjoint segments. The list decoding is then performed for each of the two segments independently, and the two generated lists are combined using a sort and match algorithm to decide on the most likely transmitted codeword. The box and match decoding strategy is developed in [5]. An alternative approach to the soft-decision decoding of linear binary block codes relies on the sphere decoding techniques [14,15]. For example, the input sphere decoder (ISD) considered in this article can be considered to be a trivial sphere decoding algorithm. Furthermore, the structure of the channel code can be exploited to design the channel code-specific list decoding [16].

In this article, we investigate the OSD-based decoding strategies for linear binary block codes. Our aim is

*Correspondence: p.loskot@swan.ac.uk
[2]College of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, UK
Full list of author information is available at the end of the article

to obtain low-complexity decoding schemes that provide sufficiently large or valuable coding gains, and more importantly, that are also well suited for implementation in the communication systems with limited hardware resources, e.g., at the nodes of wireless sensor networks [17]. We modify the original OSD by considering disjoint segments of the MRIPs [18]. Such segmentation of the MRIPs creates flexibility that can be exploited to fine-tune a trade-off between the BER performance and the decoding complexity. Thus, the original OSD can be considered to be a special case of the segmentation-based OSD having only one segment corresponding to the MRIPs. Furthermore, since the complexity of obtaining a row echelon form of the generator matrix for every received codeword represents a significant part of the overall decoding complexity, we examine a partial OSD (POSD) when only the systematic part of the received codeword is ordered. Finally, the simulation results presented in [18] are extended with the rigorous analysis of the probability of error for the OSD schemes which also includes the analysis how to select the optimum decoding list.

The rest of this article is organized as follows. System model is described in "System model" section. Construction of the list of the test error patterns is investigated in "List selection" section. The list decoding algorithms are developed in "List decoding algorithms" section. The performance analysis is considered in "Performance analysis" section. Numerical examples to compare the BER performance and the decoding complexity of the proposed decoding schemes are presented in "Numerical examples" section. Finally, conclusions are given in "Conclusions" section.

System model

Consider the transmission of codewords of a linear binary block code $\mathcal{C}$ over a Rayleigh fading channel with additive white Gaussian noise (AWGN). The code $\mathcal{C}$, denoted as $(N, K, d_{\min})$, has block length N, dimension K, and the minimum Hamming distance between any two codewords $d_{\min}$. Binary codewords $\mathbf{c} \in \mathbb{Z}_2^N$ where $\mathbb{Z}_2 = \{0, 1\}$ are generated from the vector of information bits $\mathbf{u} \in \mathbb{Z}_2^K$ using the generator matrix $\mathbf{G} \in \mathbb{Z}_2^{K \times N}$, i.e., $\mathbf{c} = \mathbf{u}\mathbf{G}$, and all binary operations are considered over a Galois field GF(2). If the code $\mathcal{C}$ is systematic, the generator matrix has the form, $\mathbf{G} = [\mathbf{I} \ \mathbf{P}]$, where $\mathbf{I}$ is the $K \times K$ identity matrix, and $\mathbf{P} \in \mathbb{Z}_2^{K \times (N-K)}$ is the matrix of parity checks. The codeword $\mathbf{c}$ is mapped to a binary phase shift keying (BPSK) sequence $\mathbf{x} \in \{+1, -1\}^N$ before the transmission. Assuming the mapping, $x_i = \mathcal{M}(c_i) = (-1)^{c_i}$, for $i = 1, 2, \dots, N$, we have the property,

$$\mathcal{M}(c_i \oplus c_j) = \mathcal{M}(c_i) \mathcal{M}(c_j) \tag{1}$$

for any encoded bits c_i and c_j, and $\oplus$ denotes the modulo 2 addition. Then, the encoded bit c_i can be recovered from the symbol x_i using the inverse mapping, $c_i = \mathcal{M}^{-1}(x_i) = (1 - x_i)/2$. For brevity, we also use the notation, $\mathbf{x} = \mathcal{M}(\mathbf{c})$ and $\mathbf{c} = \mathcal{M}^{-1}(\mathbf{x})$, to denote the component-wise modulation mapping and de-mapping, respectively. The codewords of code $\mathcal{C}$ are assumed to have equally probable values of the encoded bits, i.e., the probability, $\Pr\{c_i = 0\} = \Pr\{c_i = 1\} = 1/2$, for $i = 1, 2, \dots, N$. Consequently, all codewords are transmitted with the equal probability, i.e., one has the *a priori* probability, $\Pr\{\mathbf{c}\} = 2^{-K}$ for $\forall \mathbf{c} \in \mathcal{C}$.

The signal at the output of the matched filter (i.e., at the input to the decoder) at the receiver can be written as

$$y_i = h_i x_i + w_i$$

where the frequency non-selective channel fading coefficients h_i as well as the AWGN samples w_i are mutually uncorrelated zero-mean circularly symmetric complex Gaussian random variables. The variance of h_i is unity, i.e., $\mathrm{E}[|h_i|^2] = 1$ where $\mathrm{E}[\cdot]$ is expectation, and $|\cdot|$ denotes the absolute value. The samples w_i have the variance, $\mathrm{E}[|w_i|^2] = (R\gamma_c)^{-1}$, where $R = K/N$ is the encoding rate of $\mathcal{C}$, and γ_c is the signal-to-noise ratio (SNR) per transmitted encoded binary symbol. The covariance, $\mathrm{E}[h_i h_j^*] = 0$ for $i \neq j$, where $(\cdot)^*$ denotes the complex conjugate, corresponds to the case of fast fading channel with the ideal interleaving and deinterleaving. On the other hand, for slowly block-fading channel, the covariance $\mathrm{E}[h_i h_j^*] = 1$ for $\forall i, j = 1, 2, \dots, N$, whereas the fading coefficients are assumed to be uncorrelated between the transmissions of adjacent codewords.

In general, denote as $f(\cdot)$ the probability density function (PDF) of a random variable. The reliability r_i of the received signal y_i is given by the ratio of the conditional PDFs of y_i [19], i.e.,

$$\frac{f(y_i | x_i = +1, h_i)}{f(y_i | x_i = -1, h_i)} \propto \mathrm{Re}\{h_i^* y_i\} = r_i$$

since the PDF $f(y_i | x_i, h_i)$, for $x_i = \{+1, -1\}$, is the conditionally Gaussian distribution. Since x_i are real-valued symbols, the reliability r_i can be written as

$$r_i = \mathrm{Re}\{h_i\} \mathrm{Re}\{y_i\} + \mathrm{Im}\{h_i\} \mathrm{Im}\{y_i\} = |h_i|^2 x_i + |h_i| w_i.$$

The bit-by-bit binary-quantized (i.e., hard) decisions are then defined as

$$\hat{c}_i = \mathcal{M}^{-1}(\mathrm{sign}(r_i))$$

where $\mathrm{sign}(\cdot)$ denotes the sign of a real number.

More importantly, even though the primary metric of our interest is the BER performance of the code $\mathcal{C}$, it is mathematically more convenient to design and analyze the list decoding algorithms assuming the probability of

codeword error. Thus, we can assume that the decoding algorithm designed using the probability of codeword error also achieves a good BER performance [19].

The maximum likelihood (ML) decoder minimizing the probability of codeword error provides the decision $\hat{\mathbf{c}}_{ML}$ on the most likely transmitted codeword, i.e.,

$$\hat{\mathbf{c}}_{ML} = \text{argmin}_{\mathbf{c} \in \mathcal{C}: \, \mathbf{x} = \mathcal{M}(\mathbf{c})} \left\| \mathbf{y} - \mathbf{h} \odot \mathbf{x} \right\|^2$$

$$= \text{argmax}_{\mathbf{c} \in \mathcal{C}} \sum_{i=1}^{N} \text{Re}\{y_i h_i^* x_i^*\}$$

$$\overset{\text{BPSK}}{=} \text{argmax}_{\mathbf{c} \in \mathcal{C}: \, \mathbf{x} = \mathcal{M}(\mathbf{c})} \mathbf{r} \cdot \mathbf{x} \qquad (2)$$

where $\mathbf{y}$, $\mathbf{h}$, $\mathbf{x}$, and $\mathbf{r}$ denote the N-dimensional row vectors of the received signals y_i, the channel coefficients h_i, the transmitted symbols x_i, and the reliabilities r_i within one codeword, respectively, $\|\cdot\|$ is the Euclidean norm of a vector, $\odot$ is the component-wise (Hadamard) product of vectors, and the binary operator '$\cdot$' is used to denote the dot-product of vectors. The codewords $\mathbf{c} \in \mathcal{C}$ used in (2) to search for the maximum or minimum value of the ML metric are often referred to as the test codewords. In the following section, we investigate the soft-decision decoding algorithms with small implementation complexity to replace the computationally demanding ML decoding in (2).

List decoding
We investigate the list-based decoding algorithms. For simplicity, we assume binary block codes that are linear and systematic [20]. We note that whereas the extension of the list-based decoding algorithms to non-systematic codes is straightforward, the list-based decoding of non-linear codes is complicated by the fact that the list of the test codewords is, in general, dependent on the received sequence. More importantly, we define and measure the decoding (time) complexity O of the list decoding algorithms as the size of the list given by the number of the test codewords that are examined in the decoding process. Hence, the ML decoding (2) has the complexity, $O_{ML} = 2^K$, which is prohibitive for larger values of K. Among the practical list-based decoding algorithms with the acceptable decoding complexity, we investigate the OSD-based list decoding algorithms [4] for the soft-decision decoding of linear binary block codes.

The OSD decoding resumes by reordering the received sequence of reliabilities r_i, $i = 1, 2, \ldots, N$. Thus, let,

$$|\tilde{r}'_1| \geq |\tilde{r}'_2| \geq \cdots |\tilde{r}'_N| \qquad (3)$$

so that the ordering of reliabilities can be described by the permutation, λ', i.e.,

$$\tilde{\mathbf{r}}' = (\tilde{r}'_1, \ldots, \tilde{r}'_N) = \lambda'[\mathbf{r}].$$

The permutation λ' corresponds to the generator matrix $\tilde{\mathbf{G}}' = \lambda'[\mathbf{G}]$ having the reordered columns. In order to obtain the MRIPs for the *first* K bits in the codeword, additional swapping of the columns of $\tilde{\mathbf{G}}'$ may have to be used which corresponds to the permutation λ'', and the generator matrix $\tilde{\mathbf{G}}'' = \lambda''\left[\tilde{\mathbf{G}}'\right]$. The matrix $\tilde{\mathbf{G}}''$ is then guaranteed it can be manipulated into a row (or a reduced row) echelon form using the Gauss (or the Gauss-Jordan) elimination. To simplify the notation, let $\tilde{\mathbf{r}}$ and $\tilde{\mathbf{G}}$ denote the reordered sequence of the reliabilities $\mathbf{r}$ and the reordered generator matrix $\tilde{\mathbf{G}}$ in a row (or a reduced row) echelon form, respectively, after employing the permutations λ' and λ'', that will be used to decode the received sequence $\mathbf{y}$. Thus, for $i \geq j$, the reordered sequence $\tilde{\mathbf{r}}$ has elements, $|\tilde{r}_i| \geq |\tilde{r}_j|$, for $i, j \in \{1, \ldots, K\}$, as well as for $i, j \in \{K + 1, \ldots, N\}$.

The complexity of the full ML decoding (2) of the received sequence $\mathbf{y}$ can be reduced by assuming a list of the L test codewords, so that $L \ll 2^K$. Denote such list of the test codewords of cardinality L generated by the matrix $\tilde{\mathbf{G}}$ as, $\mathcal{E}_L = \{\mathbf{e}_0, \mathbf{e}_2, \ldots, \mathbf{e}_{L-1}\} \subset \mathbb{Z}_2^N$, and let $\mathbf{e}_0 = \mathbf{0}$ be the all-zero codeword. Then, the list decoding of $\mathbf{y}$ is defined as

$$\hat{\mathbf{c}} = \text{argmax}_{\mathbf{e} \in \mathcal{E}_L: \, \mathbf{x} = \mathcal{M}(\hat{\mathbf{c}}_0 \oplus \mathbf{e})} \tilde{\mathbf{r}} \cdot \mathbf{x} \qquad (4)$$

where the systematic part of the codeword $\hat{\mathbf{c}}_0$ is given by the hard-decision decoded bits at the MRIPs. The decoding step to obtain the decision $\hat{\mathbf{c}}_0$ is referred to as the 0th order OSD reprocessing in [4]. In addition, due to linearity of $\mathcal{C}$, we have that $(\mathbf{c}_0 \oplus \mathbf{e}) \in \mathcal{C}$, and thus, the test error patterns $\mathbf{e} \in \mathcal{E}_L$ can be also referred to as the test codewords in the decoding (4). Using property (1), we can rewrite the decoding (4) as

$$\hat{\mathbf{c}} = \text{argmax}_{\mathbf{e} \in \mathcal{E}_L} \tilde{\mathbf{r}} \cdot \hat{\mathbf{x}}_0 \cdot \mathcal{M}(\mathbf{e}) = \text{argmax}_{\mathbf{e} \in \mathcal{E}_L} \tilde{\mathbf{r}}_0 \cdot \mathcal{M}(\mathbf{e}) \qquad (5)$$

where we denoted $\hat{\mathbf{x}}_0 = \mathcal{M}(\hat{\mathbf{c}}_0)$, and $\tilde{\mathbf{r}}_0 = \tilde{\mathbf{r}} \odot \hat{\mathbf{x}}_0$. The system model employing the list decoding (5) is illustrated in Figure 1. Hence, as indicated in Figure 1, the system model can be represented as an equivalent channel with the binary vector input $\mathbf{c}$ and the vector soft-output $\tilde{\mathbf{r}}_0$.

List selection
The selection of the test error patterns $\mathbf{e}$ to be included in the list $\mathcal{E}_L$ as well as the list size L critically affect the probability of incorrect codeword decision in the list decoding. Denote such probability of codeword error as P_e, and let $\mathbf{c}_{Tx}$ be the codeword transmitted. In [14], the probability P_e is expanded as

$$P_e = \text{Pr}\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\} \text{Pr}\{\hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\}$$

$$+ \text{Pr}\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\} \text{Pr}\{\hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\}$$
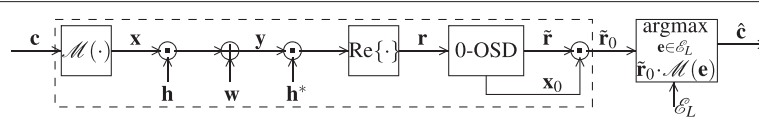
Figure 1 Equivalent system model. The system model and equivalent vector channel with the binary vector input **c** and the vector soft-output $\tilde{\mathbf{r}}_0$.

where the decision $\hat{\mathbf{c}}$ is obtained by the decoding (5), and the condition, $\hat{\mathbf{c}}_{\mathrm{ML}} \neq \mathbf{c}_{\mathrm{Tx}}$, is satisfied provided that the vectors $\hat{\mathbf{c}}_{\mathrm{ML}}$ and $\mathbf{c}_{\mathrm{Tx}}$ differ in at least one component, i.e., $\hat{\mathbf{c}}_{\mathrm{ML}} = \mathbf{c}_{\mathrm{Tx}}$ if and only if all the components of these vectors are equal. Since, for any list $\mathcal{E}_L$, the probability, $\Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{\mathrm{Tx}} | \hat{\mathbf{c}}_{\mathrm{ML}} \neq \mathbf{c}_{\mathrm{Tx}}\} = 1$, and usually, the probability, $\Pr\{\hat{\mathbf{c}}_{\mathrm{ML}} = \mathbf{c}_{\mathrm{Tx}}\}$ is close to 1, the probability P_{e} can be tightly upper-bounded as

$$\mathrm{P}_{\mathrm{e}} \leq \Pr\{\hat{\mathbf{c}}_{\mathrm{ML}} \neq \mathbf{c}_{\mathrm{Tx}}\} + \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{\mathrm{Tx}} | \hat{\mathbf{c}}_{\mathrm{ML}} = \mathbf{c}_{\mathrm{Tx}}\}. \quad (6)$$

The bound (6) is useful to analyze the performance of the list decoding (5). The first term on the right-hand side of (6) is the codeword error probability of the ML decoding, and the second term is the conditional codeword error probability of the list decoding. Furthermore, the probability $\Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{\mathrm{Tx}} | \hat{\mathbf{c}}_{\mathrm{ML}} = \mathbf{c}_{\mathrm{Tx}}\}$ is decreasing with the list size. In the limit of the maximum list size when the list decoding becomes the ML decoding, the bound (6) becomes $\mathrm{P}_{\mathrm{e}} = \Pr\{\hat{\mathbf{c}}_{\mathrm{ML}} \neq \mathbf{c}_{\mathrm{Tx}}\}$. However, in order to construct the list of the test error patterns, we consider the following alternative expansion of the probability P_{e}, i.e.,

$$\mathrm{P}_{\mathrm{e}} = \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{\mathrm{Tx}} | (\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\} \Pr\{(\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\}$$
$$+ \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{\mathrm{Tx}} | (\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \notin \mathcal{E}_L\} \Pr\{(\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \notin \mathcal{E}_L\}$$
$$= 1 - \underbrace{\Pr\{\hat{\mathbf{c}} = \mathbf{c}_{\mathrm{Tx}} | (\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\}}_{\mathrm{P_I}} \underbrace{\Pr\{(\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\}}_{\mathrm{P_{II}}}.$$

Using (4) and (5), the probability $\mathrm{P_I}$ that the list decoding (5) selects the transmitted codeword provided that such codeword is in the list (more precisely, provided that the error pattern $\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0$ is in the list) can be expressed as

$$\mathrm{P_I} = \Pr\{\tilde{\mathbf{r}} \cdot \mathcal{M}(\mathbf{c}_{\mathrm{Tx}} \oplus \hat{\mathbf{c}}_0) \geq \tilde{\mathbf{r}}_0 \cdot \mathcal{M}(\mathbf{e}), \forall \mathbf{e} \in \mathcal{E}_L\}. \quad (7)$$

The probability (7) decreases with the list size, and, in the limit of the maximum list size $L = 2^K$, $\mathrm{P_I} = 1 - \mathrm{P}_{\mathrm{e}}$. On the other hand, the probability $\mathrm{P_{II}}$ that the transmitted codeword is in the decoding list increases with the list size, and $\mathrm{P_{II}} = 1$, for $L = 2^K$.

Since the coding $\mathcal{C}$ as well as the communication channel are linear, then, without loss of generality, we can assume that the all-zero codeword, $\mathbf{c}_{\mathrm{Tx}} = \mathbf{0}$, is transmitted. Consequently, given the list decoding complexity L, the optimum list $\mathcal{E}_L^*$ minimizing the probability P_{e} is constructed as

$$\mathcal{E}_L^* = \mathrm{argmax}_{\mathcal{E}: \, |\mathcal{E}|=L} \Pr\{\hat{\mathbf{c}} = \mathbf{0} | \hat{\mathbf{c}}_0 \in \mathcal{E}\} \Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}\} \quad (8)$$

where $|\mathcal{E}|$ is the cardinality of the list $\mathcal{E}$, and the hard-decision codeword $\hat{\mathbf{c}}_0 \in \mathcal{C}$ represents the error pattern observed at the receiver after the transmission of the codeword $\mathbf{c}_{\mathrm{Tx}} = \mathbf{0}$. For given list of the test error patterns $\mathcal{E}$ in (8), and assuming the system model in "System model" section with asymptotically large SNR, the probability $\mathrm{P_I} = \Pr\{\hat{\mathbf{c}} = \mathbf{0} | \hat{\mathbf{c}}_0 \in \mathcal{E}\}$ is dominated by the error events corresponding to the error patterns with the smallest Hamming distances. Since the error patterns in the list are also codewords of $\mathcal{C}$, the smallest Hamming distance between any two error patterns in the list $\mathcal{E}$ is at least $d_{\min}$. Furthermore, assuming that the search in (8) is constrained to the lists $\mathcal{E}$ having the minimum Hamming distance between any of the two error patterns exactly equal to $d_{\min}$, the probability $\mathrm{P_I}$ is approximately constant for all these lists $\mathcal{E}$. Consequently, we can consider the sub-optimum list construction

$$\mathcal{E}_L = \mathrm{argmax}_{\mathcal{E}: \, |\mathcal{E}|=L} \Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}\}. \quad (9)$$

The list construction (9) is recursive in its nature, since the list $\mathcal{E}$ maximizing (9) consists of the L most probable error patterns. However, in order to achieve the small probability of decoding error P_{e} and approach the probability of decoding error, $\Pr\{\hat{\mathbf{c}}_{\mathrm{ML}} \neq \mathbf{c}_{\mathrm{Tx}}\}$, of the ML decoding, the list size L must be large. Hence, we can obtain a practical list construction by assuming the L sufficiently probable error patterns rather than assuming the L most likely error patterns. We restate Theorems 1 and 2 in [4] to obtain the likely error patterns and to define the practical list decoding algorithms.

Denote as $\mathrm{P}(i_1, i_2, \ldots, i_n)$ the nth-order joint probability of bit errors at the bit positions $1 \leq i_1 < i_2 < \ldots < i_n \leq N$ in the received codeword after the ordering λ' and λ'', and before the decoding. Since the test error pattern $\mathbf{e}$ is a codeword of $\mathcal{C}$, the probability $\mathrm{P}(i_1, i_2, \ldots, i_n)$, for $i_n \leq K$, is equal to the probability $\Pr\{\mathbf{e} = \hat{\mathbf{c}}_0\}$ assuming that n bit errors occurred during the transmission corresponding to the positions (after the ordering) $i_1, i_2, \ldots, i_n$. We have the following lemma.

Lemma 1. *For any bit positions $\mathcal{I}_1 \subseteq \mathcal{I} \subseteq \{1, 2, \ldots, N\}$,*

$$\mathrm{P}(\mathcal{I}) \leq \mathrm{P}(\mathcal{I}_1).$$

Proof. The lemma is proved by conditioning, i.e., note that, $\mathrm{P}(\mathcal{I}) = \mathrm{P}(\mathcal{I}_1, \mathcal{I} \backslash \mathcal{I}_1) = \mathrm{P}(\mathcal{I} \backslash \mathcal{I}_1 | \mathcal{I}_1) \mathrm{P}(\mathcal{I}_1) \leq$

$\leq \min\{P(\mathcal{I}_1), P(\mathcal{I} \backslash \mathcal{I}_1 | \mathcal{I}_1)\} \leq P(\mathcal{I}_1)$ where $\mathcal{I} \backslash \mathcal{I}_1$ denotes the difference of the two sets.

Using Lemma 1, we can show, for example, that $P(i,j) \leq P(i)$, and $P(i,j) \leq P(j)$. We can now restate Theorems 1 and 2 in [4] as follows.

Theorem 1. *Assume bit positions* $1 \leq i < j < k \leq N$, *and let the corresponding reliabilities be* $|\tilde{r}_i| \geq |\tilde{r}_j| \geq |\tilde{r}_k|$. *Then, the bit error probabilities*

$$P(i) \leq P(j)$$
$$P(i,j) \leq P(i,k).$$

Proof. Without loss of generality, we assume that the symbols $x_i = -1$, $x_j = -1$, and $x_k = -1$ have been transmitted. Then, before the decoding, the received bits would be decided erroneously if the reliabilities $\tilde{r}_i > 0$, $\tilde{r}_j > 0$, and $\tilde{r}_k > 0$. Conditioned on the particular transmitted symbols x_i, x_j, and x_k, let $f(\cdot)$ denote the conditional PDF of the ordered reliabilities $\tilde{r}_i$, $\tilde{r}_j$, and $\tilde{r}_k$.

Consider first the inequality $P(i) \leq P(j)$. Since, for $\tilde{r}_i > 0$, $f(\tilde{r}_i) < f(-\tilde{r}_i)$, using $f(\tilde{r}_i, \tilde{r}_j) = f(\tilde{r}_i | \tilde{r}_j) f(\tilde{r}_j)$, we can show that, for $\tilde{r}_i > 0$ and any $\tilde{r}_j$, $f(\tilde{r}_i, \tilde{r}_j) < f(-\tilde{r}_i, \tilde{r}_j)$. Similarly, using $f(-\tilde{r}_i, \tilde{r}_j) = f(\tilde{r}_j | -\tilde{r}_i) f(-\tilde{r}_i)$, we can show that, for $\tilde{r}_j > 0$ and any $\tilde{r}_i$, $f(-\tilde{r}_i, \tilde{r}_j) < f(-\tilde{r}_i, -\tilde{r}_j)$. Then, the probability of error for bits i and j, respectively, is

$$P(i) = \int_0^\infty \int_{-\tilde{r}_i}^{\tilde{r}_i} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i$$
$$= \int_0^\infty \int_0^{\tilde{r}_i} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i + \int_0^\infty \int_{-\tilde{r}_i}^0 f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i$$
$$P(j) = \int_{-\infty}^\infty \int_0^{|\tilde{r}_i|} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i$$
$$= \int_0^\infty \int_0^{\tilde{r}_i} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i + \int_0^\infty \int_{-\tilde{r}_i}^0 f(-\tilde{r}_i, -\tilde{r}_j) d\tilde{r}_j d\tilde{r}_i$$

and thus, $P(i) \leq P(j)$.

The second inequality, $P(i,j) \leq P(i,k)$, can be proved by assuming conditioning, $P(i,j) = P(j|i)P(i)$, $P(i,k) = P(k|i)P(i)$, and $f(\tilde{r}_i, \tilde{r}_j, \tilde{r}_k) = f(\tilde{r}_j, \tilde{r}_k | \tilde{r}_i) f(\tilde{r}_i)$, and by using inequality $P(i) \leq P(j)$, and following the steps in the first part of this proof.

List decoding algorithms

Using Theorems 1 and 2 in [4], the original OSD assumes the following list of the test error patterns,

$$\mathcal{E}_L = \{\mathbf{e}\mathbf{G} : 0 \leq w_H[\mathbf{e}] \leq I, \mathbf{e} \in \mathbb{Z}_2^K\} \quad (10)$$

where I is the so-called reprocessing order of the OSD, and $w_H[\mathbf{e}]$ is the Hamming weight of the vector $\mathbf{e}$. Thus, the list (10) uses a K-dimensional sphere of radius I defined about the origin $\mathbf{0} = (0, \ldots, 0)$ in $\mathbb{Z}_2^K$. The decoding complexity

for the list (10) is $L = \sum_{l=0}^I \binom{K}{l}$ where l is referred to as the phase of the order I reprocessing in [4]. Assuming an AWGN channel, the recommended reprocessing order is $I = \lceil d_{\min}/4 \rceil \ll K$ where $\lceil \cdot \rceil$ is the ceiling function. Since the OSD algorithm may become too complex for larger values of I and K, a stopping criterion for searching the list $\mathcal{E}_L$ was developed in [10].

We can identify the following inefficiencies of the original OSD algorithm. First, provided that no stopping or skipping rules for searching the list of the test error patterns are used, once the MRIPs are found, the ordering of bits within the MRIPs according to their reliabilities becomes irrelevant. Second, whereas the BER performance of the OSD is modestly improving with the reprocessing order I, the complexity of the OSD increases rapidly with I [10]. Thus, for given K, the maximum value of I is limited by the acceptable OSD complexity to achieve a certain target BER. We can address these inefficiencies of the original OSD by more carefully exploiting the properties of the joint probability of bit errors given by Lemma 1 and Theorem 1. Hence, our aim is to construct a well-defined list of the test error patterns *without* considering the stopping and skipping criteria to search this list.

Recall that the test error patterns are uniquely specified by bits within the MRIPs, whereas the bits outside the MRIPs are obtained using the parity check matrix of the code. In order to generate a list of the test error patterns independently of the particular generator matrix (i.e., independently of the particular code) as well as independently of the particular received sequence, we consider only the bit errors within the MRIPs. Hence, we assume that, for all test error patterns, the bit errors outside the MRIPs affect the value of the metric in (5) equally. More importantly, in order to reduce the list decoding complexity while improving the BER performance, we consider partitioning of the MRIPs into disjoint segments. This decoding strategy employing segments of the MRIPs is investigated next.

Segmentation-based OSD

Assuming Q disjoint segments of the MRIPs, the test error pattern $\mathbf{e}$ corresponding to the K MRIPs can be expressed as a concatenation of the Q error patterns $\mathbf{e}^{(q)}$ of length K_q bits, $q = 1, \ldots, Q$, i.e.,

$$\mathbf{e} = (\mathbf{e}^{(1)}, \ldots, \mathbf{e}^{(Q)}) \in \mathbb{Z}_2^K$$

so that $\sum_{q=1}^Q K_q = K$, and $w_H[\mathbf{e}] = w_H[\mathbf{e}^{(1)}] + \cdots + w_H[\mathbf{e}^{(Q)}]$. As indicated by Lemma 1 and Theorem 1, more likely error patterns have smaller Hamming weights, and they correct the bit positions having smaller reliabilities. In addition, the decoding complexity given by the total number of the test error patterns in the list should grow linearly with the number of segments Q. Consequently, for

a small number of segments Q, it is expected that a good decoding strategy is to decode each segment independently, and then, the final decision is obtained by selecting the best error (correcting) pattern from these segments decodings. In this article, we refine this strategy for $Q = 2$ segments as a generalization of the conventional OSD having only $Q = 1$ segment.

Assuming that the two segments of the MRIPs are decoded independently, the list of the test error patterns can be written as

$$\mathcal{E}_L = \mathcal{E}_{L_1}^{(1)} \cup \mathcal{E}_{L_2}^{(2)} \qquad (11)$$

where $\mathcal{E}_{L_1}^{(1)}$ and $\mathcal{E}_{L_2}^{(2)}$ are the sublists of the test error patterns corresponding to the list decoding of the first segment and of the second segment, respectively, and $L = L_1 + L_2$. Obviously, fewer errors, and thus, fewer error patterns can be assumed for shorter segments, and the segments with larger reliabilities of the received bits. Similar to the case of the conventional OSD having only one segment, for $Q = 2$ segments of the MRIPs considered, we assume all the test error patterns up to the maximum Hamming weight I_q, $q = 1, 2$. Then, the sublists of the test error patterns in (11) are defined as

$$\mathcal{E}_{L_1}^{(1)} = \{(\mathbf{e}, \mathbf{0})\mathbf{G} : 0 \le w_H[\mathbf{e}] \le I_1, \ \mathbf{e} \in \mathbb{Z}_2^{K_1}\}$$
$$\mathcal{E}_{L_2}^{(2)} = \{(\mathbf{0}, \mathbf{e})\mathbf{G} : 0 \le w_H[\mathbf{e}] \le I_2, \ \mathbf{e} \in \mathbb{Z}_2^{K_2}\}. \qquad (12)$$

Hence, the overall decoding complexity of the segmentation-based OSD with the sublists in (12) is

$$L = \sum_{l=0}^{I_1} \binom{K_1}{l} + \sum_{l=0}^{I_2} \binom{K_2}{l}$$

where $K = K_1 + K_2$, and we assume that, $I_1 \ll K_1$, and $I_2 \ll K_2$.

Recall that the original OSD, denoted as OSD(I) or OSD($I|K$), has one segment of length K bits, and that the maximum number of bit errors assumed in this segment is I. The segmentation-based OSD is denoted as OSD(I_1, I_2) or OSD($I_1|K_1, I_2|K_2$), and it is parameterized by the segments length K_1, and K_2, and the maximum number of bit errors in these segments I_1 and I_2, respectively. The segment sizes K_1 and K_2 are chosen empirically to minimize the BER for a given decoding complexity L and the set of codes under consideration. In particular, for systematic block codes of block length $N < 128$ and rate $R \ge 1/2$, it was found that the best BER performance is achieved if the length of the first segment is

$$K_1 \approx \lceil 0.35K \rceil$$

so that the second segment length is $K_2 = K - K_1$. The maximum number of bit errors I_1 and I_2 in the two segments are then selected empirically to fine-tune the BER performance and the decoding complexity trade-off. For

instance, we can set the parameters of the segmentation-based list decoding to have the BER performance as well as the decoding complexity between those corresponding to the original decoding schemes OSD(I) and OSD($I + 1$).

Finally, we note that it is straightforward to develop the skipping criteria for efficient search of the list of the test error patterns in the OSD-based decoding schemes. For instance, one can consider the Hamming distances for one or more segments of the MRIPs between the received hard decisions (before the decoding) and the temporary decisions obtained using the test error patterns from the list. If any or all of the Hamming distances are above given thresholds, the test error pattern can be discarded without re-encoding and calculating its Euclidean distance. For the $Q = 2$ segments OSD being considered, our empirical results indicate that the thresholds of the number of bit errors in the first and the second segments should be $\lceil 0.35\, d_{\min} \rceil$ and $d_{\min}$, respectively.

POSD

The Gauss (or the Gauss-Jordan) elimination employed in the OSD-based decoding algorithms represents a significant portion of the overall implementation complexity. A new row (or a reduced row) echelon form of the generator matrix must be obtained after every permutation λ'' until the MRIPs are found. Hence, we can devise a POSD that completely avoids the Gauss elimination, and thus, it further reduces the decoding complexity of the OSD-based decoding. The main idea of the POSD is to order only the first K received bits according to their reliabilities, so that the generator matrix remains in its systematic form. It is clear that such decoding strategy can only provide the coding gain if more than one segment of the information bits is considered. Thus, we assume $Q = 2$ segments, and denote this decoding as POSD(I_1, I_2) or POSD($I_1|K_1, I_2|K_2$). The parameters K_1, K_2, I_1, and I_2 can be optimized empirically as in the case of OSD(I_1, I_2) to fine-tune the BER performance versus the implementation complexity; it is recommended to use the same parameters as for the OSD(I_1, I_2) decoding. Moreover, even though the partial ordering of the first K out of N received bits is irrelevant for the OSD decoding using one segment only, we note that the POSD(I) decoding can be referred to as the input-sphere decoding ISD(I).

Implementation complexity

We compare the number of binary operations (BOPS) and the number of floating point operations (FLOPS) required to execute the decoding algorithms proposed in this article. Assuming a $(N, K, d_{\min})$ code, the complexity of the OSD and the POSD are compared in Tables 1 and 2. The implementation complexity expressions in Table 1 for OSD(I) are from the reference [4]. For example, the OSD decoding of the BCH code $(128, 64, 22)$ requires at least

Table 1 The implementation complexity of the OSD and the POSD

Operation	Complexity
OSD(I_1) and OSD(I_1, I_2)	
$\mathbf{r}$	$2N$ FLOPS
$\tilde{\mathbf{r}}'$	$N \log_2(N)$ FLOPS
Gauss el. $\mathbf{G}'$	$N \min(K, N-K)^2$ BOPS
$\tilde{\mathbf{r}}''$	$K + K(N-K)$ BOPS
POSD(I_1) $\equiv$ ISD(I_1)	
$\mathbf{r}$	$2N$ FLOPS
$\tilde{\mathbf{r}}'$	0 BOPS
POSD(I_1, I_2)	
$\mathbf{r}$	$2N$ FLOPS
$\tilde{\mathbf{r}}'$	$K \log_2(K)$ FLOPS

1, 152 FLOPS and 528, 448 BOPS to find the MRIPs and to obtain the corresponding equivalent generator matrix in a row echelon form. All this complexity can be completely avoided by assuming the POSD decoding. The number of the test error patterns is $L = 2080$ for OSD(2), and $L = 1177$ for OSD(2, 2) with $K_1 = 21$ and $K_2 = 43$ whereas the coding gain of OSD(2) is only slightly better than the coding gain of OSD(2, 2) (see Figure 2). Hence, the overall complexity of the OSD-based schemes can be substantially reduced by avoiding the Gauss (Gauss-Jordan) elimination.

Performance analysis

Recall that we assume a memoryless communication channel as described in "System model" section. We derive the probability $\Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}_L\}$ in (9) that the error pattern $\hat{\mathbf{c}}_0$ observed at the receiver after the transmission of the codeword $\mathbf{c}_{\mathrm{Tx}} = \mathbf{0}$ is an element of the decoding list $\mathcal{E}_L$. The derivation relies on the following generalization of Lemma 3 in [4].

Lemma 2. *For any ordering of the N received bits, consider the I bit positions $\mathcal{I} \subseteq \{1, 2, \ldots, N\}$, and the $\binom{I}{I_1}$ subsets $\mathcal{I}_1 \subseteq \mathcal{I}$ of I_1 bit positions, such that $I_1 \leq I \leq N$. Then, the total probability of I_1 bit errors within the I bits can be calculated as*

$$\sum_{\mathcal{I}_1: |\mathcal{I}_1| = I_1} \mathrm{P}(\mathcal{I}_1) = \binom{I}{I_1} p_0^{I_1}$$

Table 2 The decoding list sizes for the OSD and the POSD

OSD(I)	$\sum_{l=0}^{I} \binom{K}{l}$
OSD(I_1, I_2)	$\sum_{l=0}^{I_1} \binom{K_1}{l} + \sum_{l=0}^{I_2} \binom{K_2}{l}$
POSD(I) $\equiv$ ISD(I)	$\sum_{l=0}^{I} \binom{K}{l}$
POSD(I_1, I_2)	$\sum_{l=0}^{I_1} \binom{K_1}{l} + \sum_{l=0}^{I_2} \binom{K_2}{l}$

where p_0 is the probability of bit error corresponding to the bit positions $\mathcal{I}$ before the decoding.

Proof. The ordering of the chosen I bits in the given set $\mathcal{I}$ is irrelevant since *all* subsets $\mathcal{I}_1$ of I_1 errors within the I bits $\mathcal{I}$ are considered. Consequently, the bit errors in the set $\mathcal{I}$ can be considered to be independent with the equal probability denoted as p_0.

Using Lemma 2, we observe that the lists of the test error patterns (10) and (12) are constructed, so that the ordering of bits within the segments is irrelevant. Then, the bit errors within the segments can be considered to be mutually independent. This observation is formulated as the following corollary of Lemma 2.

Corollary 1. *For the* OSD(I) *and the list of the test error patterns (10), the bit errors in the MRIPs can be considered as conditionally independent. Similarly, for the* POSD(I_1, I_2) *and the list of the test error patterns (12), the bit errors in the two segments can be considered to be conditionally independent.*

Thus, the bit errors in Corollary 1 are independent conditioned on the particular segment considered as shown next.

Let P_0 be the bit error probability corresponding to the MRIPs in the OSD(I) decoding. Similarly, let P_1 and P_2 be the bit error probabilities in the first and the second segments in the OSD(I_1, I_2) decoding, respectively. Denote the auxiliary variables, $v_1 = |\tilde{r}_{K_1}|$, $v_2 = |\tilde{r}_{K_1+1}|$, and $v_3 = |\tilde{r}_{K+1}|$ of the ordered statistics (3), and let $u \equiv |r_i|$, $i = 1, 2, \ldots, K$. Hence, always, $v_1 \geq v_2$, and, for simplicity, ignoring the second permutation λ'', also, $v_2 \geq v_3$. The probability of bit error P_0 corresponding to the MRIPs is calculated as

$$\mathrm{P}_0 = \mathrm{E}_u \left[\int_0^u \frac{f_{v_3}(v)}{1 - F_u(v)} \mathrm{d}v \right]$$

where $\mathrm{E}_u[\cdot]$ denotes expectation w.r.t. (with respect to) u, $f_{v_3}(v)$ is the PDF of the $(K + 1)$th-ordered statistic in (3), and $F_u(v)$ is the cumulative distribution function (CDF) of the absolute value of the received reliability before the ordering. Similarly, the probability of bit error P_1 for the first segment is calculated as

$$\mathrm{P}_1 = \mathrm{E}_u \left[\int_0^u \frac{f_{v_2}(v)}{1 - F_u(v)} \mathrm{d}v \right]$$

where $f_{v_2}(v)$ is the PDF of the $(K_1 + 1)$th-ordered statistic in (3). The probability of bit error P_2 for the second segment is calculated as

$$\mathrm{P}_2 = \mathrm{E}_u \left[\int_0^u \int_u^\infty \frac{f_{v_1}(v) f_{v_3}(v')}{(F_u(v) - F_u(v'))(1 - F_{v_1}(v'))} \mathrm{d}v \mathrm{d}v' \right]$$
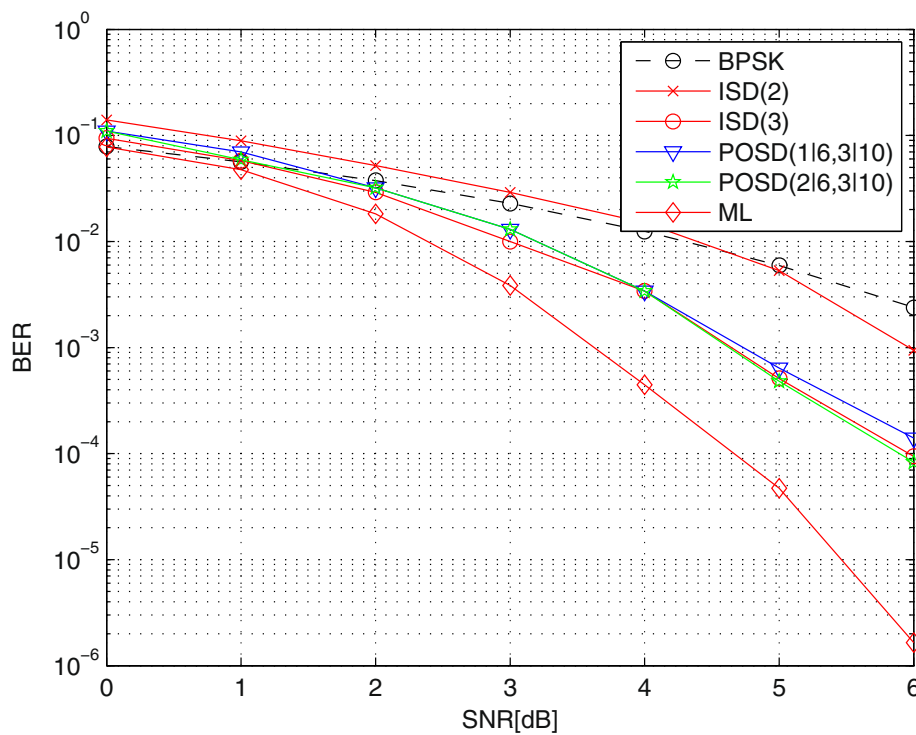
Figure 2 BER: BCH (128, 64, 22), AWGN channel. The BER of the **(128, 64, 22)** BCH code over an AWGN channel.

where $f_{\nu_1}(\nu)$ and $F_{\nu_1}(\nu')$ are the PDF and the CDF of the K_1th-ordered statistic in (3), respectively. The values of the probabilities P_0, P_1, and P_2 have to be evaluated numerically. Finally, we can substitute the probabilities P_0, P_1, and P_2 for p_0 in Lemma 2 to calculate the probability $\Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}_L\}$ of the test error patterns in the list $\mathcal{E}_L$.

Numerical examples

We use computer simulations to compare the BER performances of the proposed soft-decision decoding schemes. Recall that all block codes considered are linear and systematic.

The BER of the $(128, 64, 22)$ BCH code over an AWGN channel is shown in Figure 2 assuming OSD(1) and OSD(2) with $K = 64$, and assuming OSD(2, 2) with $K_1 = 21$ and $K_2 = 43$. The number of the test error patterns for the OSD(1), OSD(2), and OSD(2, 2) decodings are 64, 2,081, and 1,179, respectively. A truncated union bound of the BER in Figure 2 is used to indicate the ML performance [19, Ch. 10]. We observe that both OSD(2) and OSD(2, 2) have the same BER performance for the BER values larger than 10^{-3}, and OSD(2) outperforms OSD(2, 2) by at most 0.5 dB for small values of the SNR. Our numerical results show that, in general, the OSD(2, 2) decoding can achieve approximately the same BER as OSD(2) for small to medium SNR values while using about 50% less test error patterns. Thus, a slightly smaller coding gain (less than 0.5 dB) of OSD(2, 2)

in comparison with OSD(2) at larger values of the SNR is well compensated for by the reduced decoding complexity. More importantly, OSD(2, 2) can trade-off the BER performance and the decoding complexity between those provided by OSD(1) and OSD(2), especially at larger values of the SNR.

The BER of the $(31, 16, 7)$ BCH code over an AWGN channel is shown in Figure 3 assuming ISD(2) and ISD(3) with $K = 16$ having the 137 and 697 test error patterns, respectively, and assuming POSD(1, 3) and POSD(2, 3) with $K_1 = 6$ and $K_2 = 10$ having the 183 and 198 test error patterns, respectively. We observe that POSD(1, 3) achieves the same BER as ISD(3) while using much less error patterns which represents the gain of ordering the received information bits into two segments. At the BER of 10^{-4}, POSD(1, 3) outperforms ISD(2) by 1.1 dB using approximately 50% more test error patterns. Thus, the POSD(1, 3) decoding provides 2.3 dB coding gain, and has a small implementation complexity at the expense of 2 dB loss compared to the ML decoding.

Figure 4 shows the BER of the $(63, 45, 14)$ BCH code over an AWGN channel. The number of the test error patterns for the ISD(2), ISD(3), POSD(1, 3), and OSD(2) decodings are 1,036, 15,226, 5,503, and 1,036, respectively. We observe from Figure 4 that ISD(3) has the same BER as POSD(1, 3) with two segments of $K_1 = 13$ and $K_2 = 32$ bits. However, especially for high rate codes (i.e., having rates greater than 1/2), one has to also consider
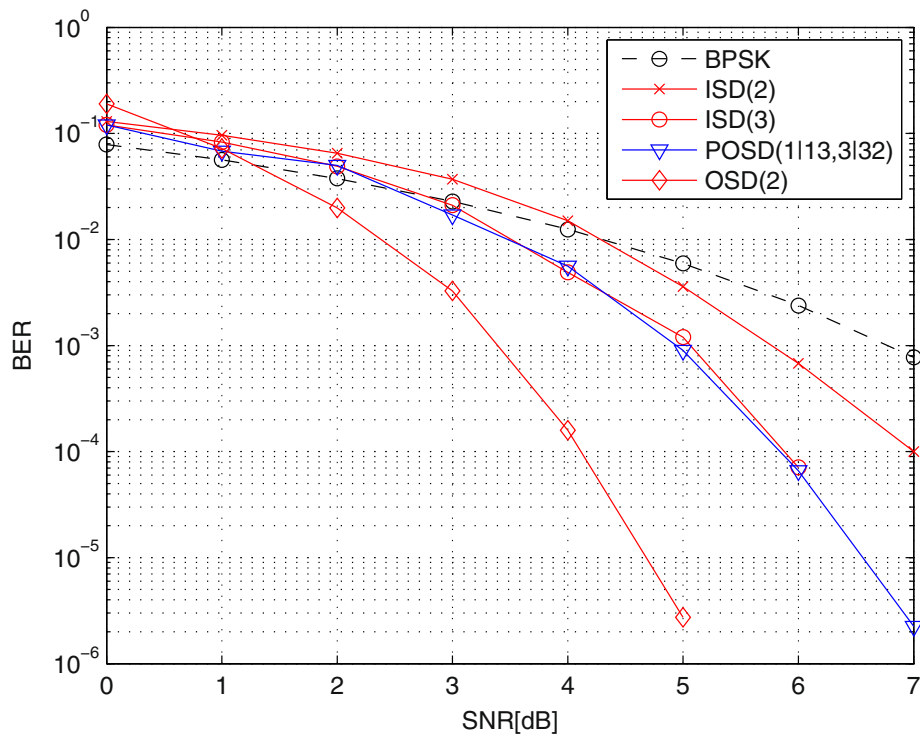
Figure 3 BER: BCH $(31, 16, 7)$, AWGN channel. The BER of the $(31, 16, 7)$ BCH code over an AWGN channel.
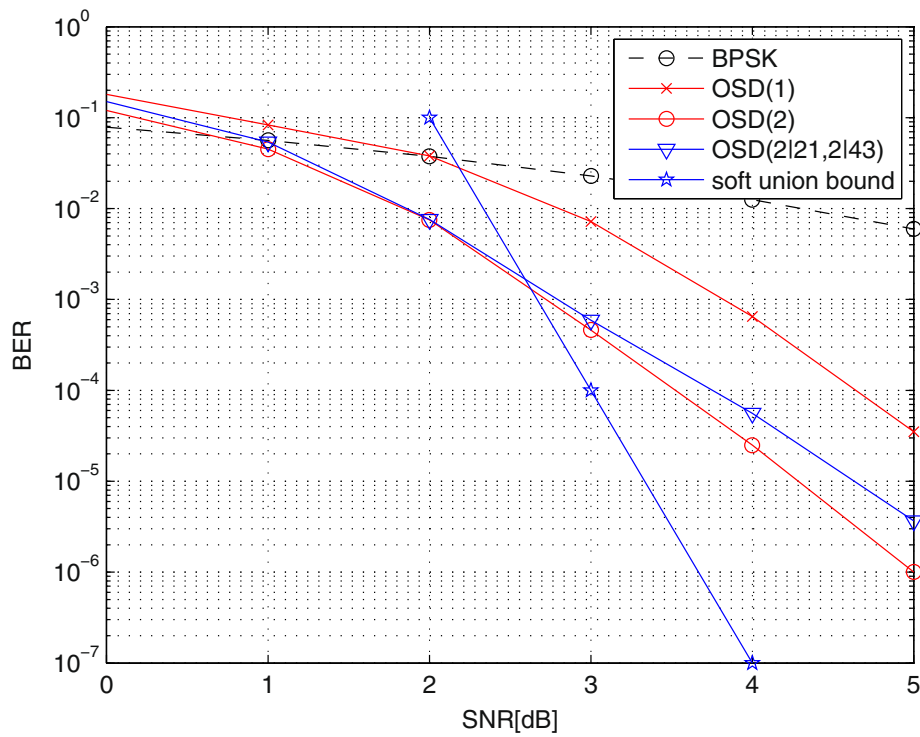

Figure 4 BER: BCH $(63, 45, 14)$, AWGN channel. The BER of the $(63, 45, 14)$ BCH code over an AWGN channel.

the complexity of the Gauss elimination to obtain the row echelon form of the generator matrix for the OSD. For example, the Gauss elimination for the $(63, 45, 14)$ code requires approximately $20, 400$ BOPS (cf. Table 1).

The BER of the $(31, 16, 7)$ BCH code over a fast Rayleigh fading channel is shown in Figure 5. We assume the same decoding schemes as in Figure 3. The POSD$(1, 3)$ decoding with the 183 test error patterns achieves the coding gain of 17 dB over an uncoded system, the coding gain of 4 dB over ISD(2) with the 137 test error patterns, while it has the same BER as OSD(3) with the 697 test error patterns. The BER of the high-rate BCH code $(64, 57, 4)$ over a fast Rayleigh channel is shown in Figure 6. In this case, the number of the test error patterns for the ISD(2), ISD(3), POSD$(2, 3)$, and OSD(2) decoding is $1, 654$, $30, 914$, $8, 685$, and $1, 654$, respectively. We observe that, for small to medium SNR values, POSD$(2, 3)$ which does not require the Gauss elimination (corresponding to approximately $3, 000$ BOPS) outperforms OSD(2) by 1 dB whereas, for large SNR values, these two decoding schemes achieve approximately the same BER performance.

Conclusions

The low-complexity soft-decision decoding techniques employing the list of the test error patterns for linear binary block codes of small to medium block length were investigated. The optimum and sub-optimum construction of the list of the test error patterns was developed. Several properties of the joint probability of bit errors after the ordering were derived. The original OSD algorithm was generalized by assuming segmentation of the MRIPs. The segmentation of the MRIPs was shown to overcome several drawbacks of the original OSD, and it also enables flexibility to devise new decoding strategies. The decoding complexity of the OSD-based decoding algorithms was further reduced by avoiding the Gauss (or the Gauss-Jordan) elimination using a partial ordering of the received bits in the POSD decoding. The performance analysis was concerned with the problem of finding the probability of the test error patterns contained in the decoding list. The BER performance and the decoding complexity of the proposed decoding schemes were compared by extensive computer simulations. Numerical examples demonstrated excellent flexibility of the proposed decoding schemes to trade-off the BER performance and the decoding complexity. In some cases, both the BER performance as well as the decoding complexity of the segmentation-based OSD were found to be improved compared to the original OSD.

Appendix

We derive the probabilities P_0, P_1, and P_2 in "Performance analysis" section. Without loss of generality, we assume
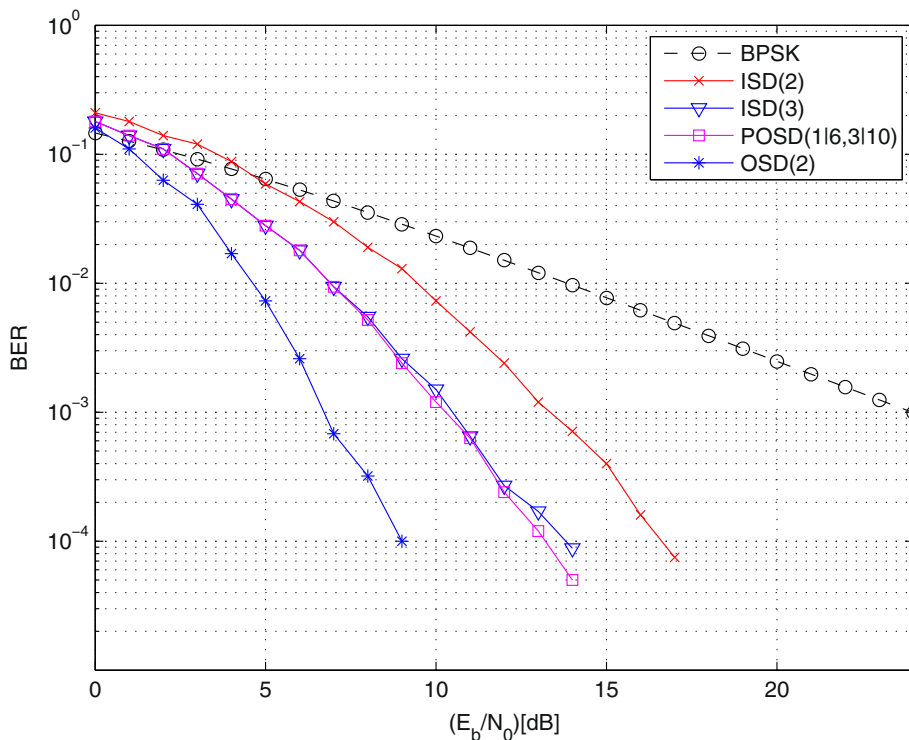

Figure 5 BER: BCH $(31, 16, 7)$, Rayleigh fading channel. The BER of the $(31, 16, 7)$ BCH code over a Rayleigh fading channel.
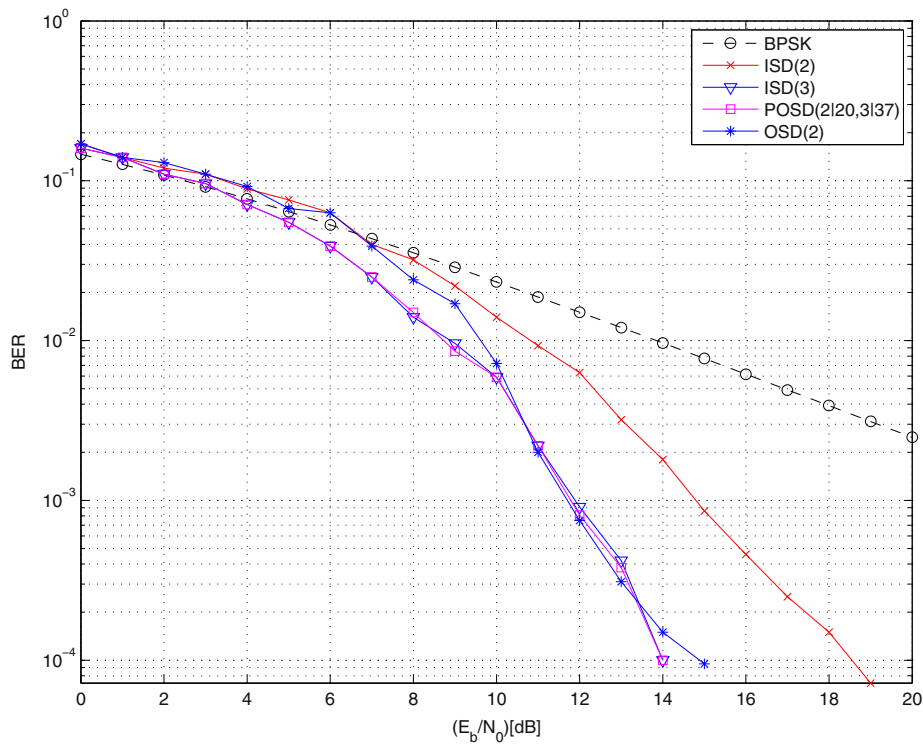
Figure 6 BER: BCH (64, 57, 4), Rayleigh fading channel. The BER of the (64, 57, 4) BCH coded over a Rayleigh fading channel.

that the all-ones codeword was transmitted, i.e., $x_i = -1$ for $\forall i$. Then, after the ordering, the ith received bit, $i = 1, 2, \ldots, N$, is in error, provided that $\tilde{r}_i > 0$. The probability of bit error P_0 for bits within the MRIPs is calculated as

$$P_0 = \int_0^\infty \int_0^\infty f_u(u|u \geq v_3) f_{v_3}(v_3) dv_3 du$$

where the conditional PDF [21],

$$f_u(u|u \geq v_3) = \begin{cases} \dfrac{f_u(u)}{1 - F_u(v_3)} & u \geq v_3 \\ 0 & u < v_3 \end{cases}$$

and $f_u(u)$ and $F_u(v_3)$ are the PDF and the CDF of the reliabilities of the received bits, respectively, and thus,

$$P_0 = \int_0^\infty f_u(u) \int_0^u \frac{f_{v_3}(v_3)}{1 - F_u(v_3)} dv_3 du.$$

Similarly, after the ordering, the probability of bit error P_1, for the received bits in the first segment, is calculated as

$$P_1 = \int_0^\infty \int_0^\infty f_u(u|u \geq v_2) f_{v_2}(v_2) dv_2 du$$

$$= \int_0^\infty f_u(u) \int_0^u \frac{f_{v_2}(v_2)}{1 - F_u(v_2)} dv_2 du.$$

The probability of bit error P_2 after the ordering, for the received bits in the second segment, is calculated as

$$P_2 = \int_0^\infty \int_0^\infty \int_0^\infty f_u(u|v_1 \geq u \geq v_3) f_{v_1, v_3}(v_1, v_3) dv_1 dv_3 du$$

where the conditional PDF,

$$f_u(u|v_1 \geq u \geq v_3) = \begin{cases} \dfrac{f_u(u)}{F_u(v_1) - F_u(v_3)} & v_1 \geq u \geq v_3 \\ 0 & \text{otherwise} \end{cases}$$

and the joint PDF of the ordered statistics $v_1 \geq v_3$ is

$$f_{v_1, v_3}(v_1, v_3) = \begin{cases} \dfrac{f_{v_1}(v_1)}{1 - F_{v_1}(v_3)} f_{v_3}(v_3) & v_1 \geq v_3 \\ 0 & v_1 < v_3 \end{cases}$$

and thus,

$$P_2 = \int_0^\infty f_u(u) \int_0^u \int_u^\infty \frac{f_{v_1}(v) f_{v_3}(v')}{(F_u(v) - F_u(v'))(1 - F_{v_1}(v'))} dv dv' du.$$

Competing interests
The authors declare that they have no competing interests.

Author details
[1] Electrical Engineering Department, Mu'tah University, P.O. BOX (7), Postal Code 61710, Mu'tah, Jordan. [2] College of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, UK.

References

1. N Chen, Z Yan, Complexity analysis of Reed-Solomon decoding over GF(2m) without using syndromes. EURASIP J. Wirel. Commun. Netw. **2008**(Article ID 843634), 1-11 (2008)
2. S Howard, C Schlegel, K Iniewski, Error control coding in low-power wireless sensor networks: when is ECC energy-efficient? EURASIP J. Wirel. Commun. Netw. **2006**(Article ID 74812), 1-14 (2006)
3. H Yagi, *A study on complexity reduction of the reliability-based maximum likelihood decoding algorithm for block codes*: (PhD thesis, Waseda University, 2005)
4. M Fossorier, S Lin, Soft-decision decoding of linear block codes based on ordered statistics. IEEE Trans. Inf. Theory. **41**, 1379–1396 (1995)
5. A Valembois, M Fossorier, Box and match techniques applied to soft decision decoding. IEEE Trans. Inf. Theory. **50**, 796–810 (2004)
6. B Dorsch, A decoding algorithm for binary block codes and J-ary output channels. IEEE Trans. Inf. Theory. **20**, 391–394 (1974)
7. J D Gazelle, Snyders, Reliability-based code-search algorithms for maximum-likelihood decoding of block codes. IEEE Trans. Inf. Theory. **43**, 239–249 (1997)
8. A Kabat, F Guilloud, R Pyndiah, New approach to order statistics decoding of long linear block codes. in *Proceedings of the GLOBECOM 2007*. (Washington, DC, USA, 26–30, November 2007, pp. 1467–1471
9. H Yagi, T Matsushima, S Hirasawa, Fast algorithm for generating candidate codewords in reliability-based maximum likelihood decoding. IEICE Trans. Fund. Electron. Commun. Comput. Sci. **E89-A**, 2676–2683 (2006)
10. M Fossorier, S Lin, Computationally efficient soft decision decoding of linear block codes based on ordered statistics. IEEE Trans. Inf. Theory. **42**, 738–750 (1996)
11. M Fossorier, Reliability-based soft-decision decoding with iterative information set reduction. IEEE Trans. Inf. Theory. **48**, 3101–3106 (2002)
12. W Jin, M Fossorier, Reliability based soft decision decoding with multiple biases. IEEE Trans. Inf. Theory. **53**, 105–119 (2007)
13. A Valembois, M Fossorier, Sort-and-match algorithm for soft-decision decoding. IEEE Trans. Inf. Theory. **45**, 2333–2338 (1999)
14. M El-Khamy, H Vialko, B Hassibi, R McEliecce, Performance of sphere decoding of block codes. IEEE Trans. Commun. **57**, 2940–2950 (2009)
15. H Vikalo, B Hassibi, On joint detection and decoding of linear block codes on Gaussian vector channels. IEEE Trans. Signal Process. **54**, 3330–3342 (2006)
16. T Hu, M Chang, List decoding of generalized Reed-Solomon codes by using a modified extended key equation algorithm. EURASIP J. Wirel. Commun. Netw. **2011**(Article ID 212136), 1-6 (2011)
17. J Ser, D Manjarres, P Crespo, S Gil-Lopez, J Garcia-Frias, Iterative fusion of distributed decisions over the Gaussian multiple-access channel using concatenated BCH-LDGM codes. EURASIP J. Wirel. Commun. Netw. **2011**(Article ID 825327) (2011)
18. S Alnawayseh, P Loskot, Low-complexity soft-decision decoding techniques for linear binary block codes. in *Proceedings of the WCSP, 2009*. Nanjing, China, 13–15 November 2009, pp. 1–5
19. S Benedetto, E Biglieri, *Principles of Digital Transmission with Wireless Applications*: (Kluwer Academic, Netherlands, 1999)
20. S Lin, D Costello, *Error Control Coding: Fundamentals and Applications*: (Prentice-Hall Englewood Cliffs, NJ, 1983)
21. A Papoulis, S Pillai, *Probability, Random Variables, and Stochastic Processes*. 4th edn (McGraw-Hill, NY, 2002)