Transactions Papers

Performance Assessment of DC-Free Multimode Codes

Kees A. Schouhamer Immink, *Fellow, IEEE,* and Levente Pátrovics

***Abstract*—We report on a class of high-rate dc-free codes, called multimode codes, where each source word can be represented by a codeword taken from a selection set of codeword alternatives. Conventional multimode codes will be analyzed using a simple mathematical model. The criterion used to select the "best" codeword from the selection set available has a significant bearing on the performance. Various selection criteria are introduced and their effect on the performance of multimode codes will be examined.**

I. Introduction

BINARY sequences with spectral nulls at zero frequency have found widespread application in optical and magnetic recording systems. The *dc-balanced* or *dc-free* codes, as they are often called, have a long history and their application is certainly not confined to recording practice. Since the early days of digital communication over cable, dc-balanced codes have been employed to counter the effects of low-frequency cut-off due to coupling components, isolating transformers, etc. In optical recording, dc-balanced codes are employed to circumvent or reduce interaction between the data written on the disc and the servo systems that follow the track [1]. In the literature, code implementations have been concentrated on byte-oriented dc-free codes of rate 8/10 or 8/9. (see, for example, [2]–[4]). For certain application it is desirable that the code rate is much higher than 8/9. The construction of such high-rate codes is far from obvious, as table look-up for encoding and decoding is an engineering impracticality. Two methods for high-rate code design have been described in the literature [5], [6]. Both methods utilize the idea that the correspondence between source words and the codewords is as simple as possible. A serious drawback of both methods is that the performance, in terms of suppression of low-frequency components, is far from what could be obtained according to the tenets of information theory [1], [7], but up till now attempts to improve the performance failed. Recently, however, the publications by Fair *et al.* on "guided scrambling" stimulated us to investigate the performance of their method and its varieties. In our context, guided scrambling is a member of a larger class of related coding schemes called *multimode* code. In multimode codes, each source word $\mathbf{x}$ can be represented by a member of a selection set consisting of L codewords. The encoder opts for transmitting that codeword that minimizes, according to a criterion to be defined, the low-frequency spectral contents of the encoded sequence. There are two key elements which need to be chosen judiciously: 1) the mapping between the source words and their corresponding selection sets and 2) the criterion used to select the "best" word. The spectral performance of the code greatly depends on both issues. We start with some preliminaries, followed by a section providing the state of the art. Thereafter, we will outline the new multimode schemes and analyze their spectral performance.

Paper approved by E. Ayanoglu, the editor for Communication Theory and Coding Application of the IEEE Communications Society. Manuscript received May 7, 1996; revised August 2, 1996. This paper was presented in part at the IEEE 3rd Symposium on Communications and Vehicular Technology, Eindenhoven, The Netherlands, October 25–26, 1995.

The authors are with the Philips Research Laboratories, 5656 AA Eindhoven, The Netherlands.

Publisher Item Identifier: S 0090-6778(97)01970-3.

II. Preliminaries

The running digital sum of a sequence (RDS) plays a significant role in the analysis and synthesis of codes whose spectrum vanishes at the low-frequency end. Let $\{x_i\} = \{\cdots, x_{-1}, x_0, \cdots, x_i, \cdots\}$, $x_i \in \{-1, 1\}$ be a bipolar sequence. Note that in the sequel, we will denote the value ∓ 1 of x_i by its logical equivalents '0' or "1." The (running) digital sum z_i is defined as

$$z_i = \sum_{j=-\infty}^{i} x_j = z_{i-1} + x_i.$$

It is an elementary exercise to show that if z_i is bounded, the spectral density vanishes at zero frequency [1]. The number of RDS values that the sequence assumes is often called the *digital sum variation* and denoted by N. The value of N should be as small as possible as it has a direct bearing on the amount of power at the low-frequency end. Given the parameter N it is possible to compute the maximum value of the rate, $C(N)$, of any code, irrespective of its complexity, that translates arbitrary source input into sequences obeying the given constraint. Results of computation, taken from [1], are listed in Table I. It can be seen that the sum constraint is not very expensive in terms of rate loss when N is relatively large. For instance, a sequence that takes at maximum $N = 10$ sum values has a capacity $C(10) = 0.94$, which implies a rate loss

TABLE I
CAPACITY AND SUM VARIANCE OF MAXENTROPIC SEQUENCES TAKING AT MOST N RDS VALUES VERSUS DIGITAL SUM VARIATION N

N	$C(N)$	$\sigma_z^2(N)$
3	0.5000	0.5000
4	0.6942	0.8028
5	0.7925	1.1667
6	0.8495	1.5940
7	0.8858	2.0858
8	0.9103	2.6424
9	0.9276	3.2639
10	0.9403	3.9506
11	0.9500	4.7026

of less than 6%. The quantity called *sum variance* $\sigma_z^2(N)$ plays an important role in the evaluation of the spectral properties of a code. Before explaining the relevance of the parameter $\sigma_z^2(N)$, a few words are in order regarding the low-frequency properties of dc-free codes.

If N is finite, the spectral density is zero at zero frequency, but it is more relevant to note that there is a region of frequencies, close to the zero frequency, where the spectral density is low. The width of this region, termed the *notch width*, is of great engineering relevance. The width of the spectral notch, can be quantified by a parameter called the *cutoff frequency*, ω_0. According to the work by Justesen [8] there is a simple approximate relationship between ω_0 and the sum variance, s_z^2, of the encoded sequence, namely

$$2s_z^2\omega_0 \simeq 1. \tag{1}$$

Sequences of maximum entropy assuming at most N RDS values obey the following fundamental relationship between the sum variance $\sigma_z^2(N)$ and the redundancy $1 - C(N)$ [1]

$$0.25 \geq (1 - C(N))\sigma_z^2(N) > \frac{\pi^2/6 - 1}{4\ln 2} = 0.2326. \tag{2}$$

The above relationship can be employed to derive a simple yard stick for measuring the performance of implemented codes. The encoder efficiency is defined as

$$E = \frac{\{1 - C(n)\}\sigma_z^2(N)}{\{1 - R\}s^2(N)}. \tag{3}$$

The encoder efficiency E, as defined in (3), compares the "redundancy-sum variance products" of the implemented code and the maxentropic sequence with the same digital sum variation as the implemented code. The efficiency E will be used in the sequel to measure the performance of dc-free codes.

III. PRIOR ART

Essentially, there are three basic methods for generating dc-free sequences which are relevant to the ensuing discussion. These methods are briefly reviewed below.

A. Monomode Codes

In monomode codes, there is a one-to-one relationship between source words and codewords. By necessity, the codewords have equal numbers of 1's and -1's. There are two methods available for translating source words into codewords. The first method uses an algebraic technique, called *enumeration* [1], and in the second method, devised by Knuth [6], m-bit source words are translated into $(m+p)$-bit codewords. The translation is achieved by selecting a bit position within the m-bit word which defines two segments, each having one half of the total disparity of the m-bit word, where the *disparity* of a codeword is defined as the difference between the numbers of 1's and -1's in that codeword. A zero-disparity codeword, i.e., a codeword with an equal number of 1's and -1's, is now generated by the inversion of all the bits within one segment. The position information which defines the two segments is encoded in the p bits.

B. Bimode Codes

Bimode codes ensure balanced transmission by providing for each source word two alternative channel representations. From the alternatives available, that codeword is transmitted that minimizes the absolute value of the RDS after transmission of the new word. This selection criterion will be termed *MRDS selection criterion*. An archetypical example of a bimode code is the *polarity switch* code [5]. The encoder and decoder circuits of the polarity switch code are very simple as no look-up tables are required. Under polarity switch rules, $(n-1)$ source symbols are supplemented by one symbol called the *polarity bit.* The encoder has the option to transmit the n-bit words without modification or to invert all n symbols. The choice of a specific translation is made in such a way that the running digital sum after transmission of the new word is as close to zero as possible. The polarity bit is used at the decoder site to identify whether the transmitted codeword has been inverted or not, and can easily be reconstituted. Properties of the polarity bit code have been described in [1]. The performance of the polarity switch code can be summarized as follows. The rate of the polarity bit code is

$$R = 1 - \frac{1}{n}.$$

The sum variance of the code [1] is

$$s^2 = \frac{2n-1}{3}$$

so that the efficiency E is

$$E = 0.2326 : \frac{2}{3} \approx 0.3489. \tag{4}$$

From the above, we conclude that polarity switch codes are a far cry from the optimal situation.

It is not difficult to generalize the above principle of bimode codes to *multimode* codes, which, as the name already suggests, cater for more than two channel representations.

C. Multimode Codes

In multimode codes, each source word $\mathbf{x}$ can be represented by a member of a selection set, denoted by $\mathcal{C}_x$, consisting of L codewords. The MRDS selection criterion can be used to select the "best" codeword. More sophisticated selection criteria will be described in Section V. It should be appreciated that the usage of multimode codes is not confined to the generation of dc-free sequences. Provided that L is large enough and the selection sets $\mathcal{C}_x$ contain sufficiently different codewords, multimode codes can also be used to satisfy almost any channel constraint with a suitably chosen selection method.

A basic element of multimode codes is the one-to-L invertible mapping between the source $\mathbf{x}$ and its selection set $\mathcal{C}_x$. Examples of such mappings are the guided scrambling algorithm presented by Fair *et al.* [9], the dc-free coset codes of Deng and Herro [10], and the scrambling using a Reed–Solomon code by Kunisa *et al.* [11]. In our context, a mapping is considered to be "good" if the sets $\mathcal{C}_x$ contain sufficiently distinct codewords. The guided scrambling algorithm is briefly described below.

1) Guided Scrambling: The guided scrambling algorithm uses selection sets of size $L = 2^r$, where r is the number of redundant bits. Guided scrambling is summarized below.

1) In the first step, called *augmenting*, the source word $\mathbf{x}$ is preceded by all the possible binary sequences of length r to produce the set $\mathcal{B}_x = \{\mathbf{b}_1, \cdots, \mathbf{b}_L\}$. Hence

$$\mathbf{b}_1 = (0, 0, \cdots, 0, x_1, \cdots, x_m), \cdots,$$
$$\mathbf{b}_L = (1, 1, \cdots, 1, x_1, \cdots, x_m).$$

2) The selection set $\mathcal{C}_x = \{\mathbf{c}_1, \cdots, \mathbf{c}_L\}$ is obtained by scrambling all vectors in $\mathcal{B}_x$. Let the scrambler polynomial $s(x)$ be denoted by

$$s(x) = x^s + \sum_{k=1}^{s} a_k x^{s-k}$$

where s denotes the register length of the scrambler. The scrambler translates each vector $\mathbf{b} = (b_1, \cdots, b_n) \in \mathcal{B}_x$ into $\mathbf{c} = (c_1, \cdots, c_n) = f(\mathbf{b}) \in \mathcal{C}_x$ using the recursion

$$c_i = b_i + \sum_{k=1}^{s} a_k c_{i-k}. \tag{5}$$

3) The "best" codeword in $\mathcal{C}_x$ is selected for transmission.
4) At the receiver's site, the inverse operation $\mathbf{b} = f^{-1}(\mathbf{c})$ is

$$b_i = c_i + \sum_{k=1}^{s} a_k c_{i-k}.$$

The source word is found by deleting the first r bits.

In the guided scrambling algorithm described above, translation of source words into 2^r random-like channel representations is done in a fairly simple way. This basic algorithm is, however, prone to worst case situations since there is a probability that consecutive source words have representation sets whose members all have the same polarity of the disparity. In this vexatious situation, the RDS cannot be controlled, and long-term low-frequency components can build up. This flaw can be solved by a construction where each selection set consists of pairs of words of opposite disparity. As a result, there is always a codeword in the selection set that can control the RDS. A simple method embodying this idea combines the features of guided scrambling and the polarity bit code. The improved algorithm using $r \geq 2$ redundant bits is executed in six steps. In Steps 1), 2), and 5) the original guided scrambling principle is executed while Steps 3) and 4) embody the polarity bit code.

1) The source word $\mathbf{x}$ is preceded by all the possible binary sequences of length $(r-1)$ to produce the $L' = 2^{r-1}$ elements of the set $\mathcal{B}_x = \{\mathbf{b}_1, \cdots, \mathbf{b}_{L'}\}$. Hence:

$$\mathbf{b}_1 = (0, 0, \cdots, 0, x_1, \cdots, x_m), \cdots,$$
$$\mathbf{b}_{L'} = (1, 1, \cdots, 1, x_1, \cdots, x_m).$$

2) The selection set $\mathcal{B}'_x = \{\mathbf{b}'_1, \cdots, \mathbf{b}'_{L'}\}$ is obtained by scrambling all vectors in $\mathcal{B}_x$.
3) By preceding the vectors in $\mathcal{B}'_x$ with both a 'one' and a "zero," we get the set $\mathcal{B}''_x$, with $L = 2^r$ elements.
4) The selection set $\mathcal{C}_x$ is obtained by scrambling (precoding) the vectors in $\mathcal{B}''_x$ using the scrambler with polynomial $x + 1$. This embodies the polarity bit principle.
5) The "best" codeword in $\mathcal{C}_x$ is selected.
6) At the receiver end, the codeword is first descrambled using the $x+1$ polynomial, then after removing the first bit, it is descrambled. The original source word $\mathbf{x}$ is eventually reconstituted by removing the first $(r-1)$ bits.

All simulations and analyses discussed below assume the above structure where the selection set consists of pairs of words of opposite disparity.

IV. Analysis of Multi-Mode Codes

A precise mathematical analysis of the performance of multimode codes is, considering the complexity of the code, out of the question. We can either rely on computer simulation to facilitate an understanding of the operation of the coding system or try to define a simple mathematical model, which embraces the essential characteristics of the code and is also analytically tractable. We followed both approaches, and we commence by describing the mathematical model.

A. The Random Drawing Model

The key characteristic of a multimode code is that each source word can be represented by a codeword taken from a set containing L "random" alternatives. As the precise structure of the encoder is extremely difficult to analyze, we assume, in our mathematical model, that for each source block $\mathbf{x}$ the channel set $\mathcal{C}_x$ is obtained by *randomly* drawing $L/2$ n-bit words plus their $L/2$ complementary n-bit words. The precise structure of the scrambler is ignored in our model. The "best" word in the set, according to the MRDS criterion, is transmitted. The MRDS criterion ensures that the state space of the encoder, that is, the number of possible *word-end running digital sum* (WRDS) values the encoded sequence may take, is finite. However, if the codewords are relatively long, the number of states and the resulting transition matrix are still too large for a simple mathematical analysis. We therefore truncated the state space by omitting those states that do not contribute significantly to the sum variance.

B. Transition Probabilities of the Finite-State Machine

The implemented encoder schemes can be simply treated in terms of Markov models. The set of values that WRDS take prior to transmission of a codeword defines a set of states of a finite-state machine. We will use the shorthand notation $Z^{(i)}$ to denote both the WRDS at the start of the ith codeword and to refer to the encoder state itself. We commence our analysis with a computation of the state transition probabilities.

Assume the ith codeword starts with RDS $Z^{(i)} = Z'$. Then the multimode code can be cast into a Markov chain model whose state transition probabilities matrix, T, is given by

$$T[Z', Z''] = \mathcal{P}(Z^{(i+1)} = Z'' \mid Z^{(i)} = Z').$$

We make the following remarks concerning the state transition matrix.

1) For the sake of simplicity, only codes using codewords of even length are considered.
2) It is assumed that at the start of the transmission WRDS is set to $+1$. As a result, since the codeword length is even, $Z^{(i)} \in \{\pm 1, \pm 3, \cdots\}$.
3) For reasons of symmetry, only the probabilities for $Z' > 0$ need to be calculated.
4) To reduce the computational load, we truncated the state space. Only those states are considered that can be reached from the $Z' = 1$, or the $Z' = -1$, state with probability greater than ϵ, where ϵ is chosen suitably small, say 10^{-6}. Other values of ϵ have been tried without, however, causing significant differences in the results obtained. The remaining states will be termed *principal states*.

We now introduce several notations. If WRDS is positive, then, according to the simple MRDS criterion, the next codeword will be of zero or negative disparity. Therefore, assuming that the encoder occupies state Z', the set of possible next states is $\mathcal{Z}_{Z'} = \{Z', Z'-2, \cdots, Z'-n\}$. Let $p(d)$ denote the probability of a codeword pair having disparity $+d$ and $-d$. The probability of the *next-state candidate* in a draw being Z^* is

$$p_{Z^*} = \begin{cases} p(|Z^* - Z'|), & \text{if } Z^* \in \mathcal{Z}_{Z'}; \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

The next-state candidate in the jth draw is denoted by $Z_j^*, j = 1, \cdots, L'$. According to the MRDS criterion, if the next state is Z'', then $|Z_j^*| \geq |Z''|$ for all j. The probability that during a draw the next-state candidate is "worse" than Z'', denoted by $q_{Z''}$, is given by

$$q_{Z''} = \sum_{|Z^*| > |Z''|, Z^* \in \mathcal{Z}_{Z'}} p_{Z^*}.$$

Now, the expression for the transition matrix T is given by

$$T[Z', Z''] = \frac{p_{Z''}}{p_{Z''} + p_{-Z''}} \left[(p_{Z''} + p_{-Z''} + q_{Z''})^{L'} - q_{Z''}^{L'} \right]. \quad (7)$$

The transition probabilities for each pair of WRDS states can be numerically determined by invoking (7). In order to make the analysis more tractable, those states are removed that can be reached from the $Z' = 1$, or the $Z' = -1$, state with probability less than ϵ. The remaining set of states, the principal states, denoted by $\mathcal{S}_K = \{-K, -K+2, \cdots, K-2, K\}$, and the truncated transition probability matrix T with elements $t_{i,j},\ i,j \in \mathcal{S}_K$ can easily be found. Thereafter, vector of the stationary probabilities, $\boldsymbol{\pi}$ with elements $\pi_i,\ i \in \mathcal{S}_K$, is found by solving $\boldsymbol{\pi} T = \boldsymbol{\pi}$. The calculation of the variance of the digital sum at the start of the codewords is now straightforward. The computation of the sum variance within the codewords is more complex, and therefore given in the Appendix.

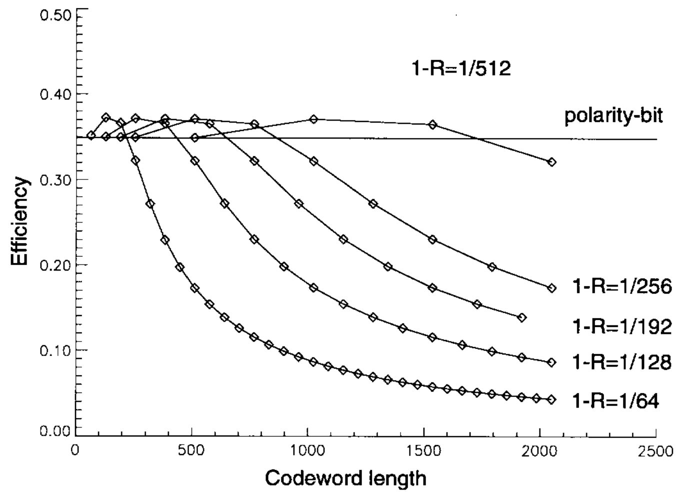

Fig. 1. Efficiency of random drawing algorithm using the MRDS selection criterion.

C. Computational Results

Using (13), we calculated the efficiency of the random drawing algorithm for selected values of the codeword length and redundancy. Fig. 1 shows the results. The connected points have the same redundancy $(1-R)$, and the ith point on a curve corresponds to a code having i redundant bits, codeword length $i/(1-R)$, and selection sets of size 2^i. For comparison purposes, we also plotted the efficiency of the polarity bit code [see (4)]. By comparing the efficiency values at the ith point on each curve, we can see that these values are approximately the same. The efficiency of the random coding algorithm is practically independent of the codeword length and is essentially determined by the *number* of redundant bits used. It can be seen that codes with two or three redundant bits are clearly more efficient than the polarity bit code. With an increasing number of redundant bits, however, the efficiency decreases. The decrease in performance, as will be explained in the next section, is due to the shortcomings of the MRDS criterion.

V. Alternative Selection Criteria

The results, plotted in Fig. 1, reveal that using more than two redundant bits does not lead to improved performance. The reason that performance decreases with an increasing number of redundant bits can easily be understood. A quick calculation will make it clear that a large selection set contains with great probability at least one zero-disparity word. On the basis of

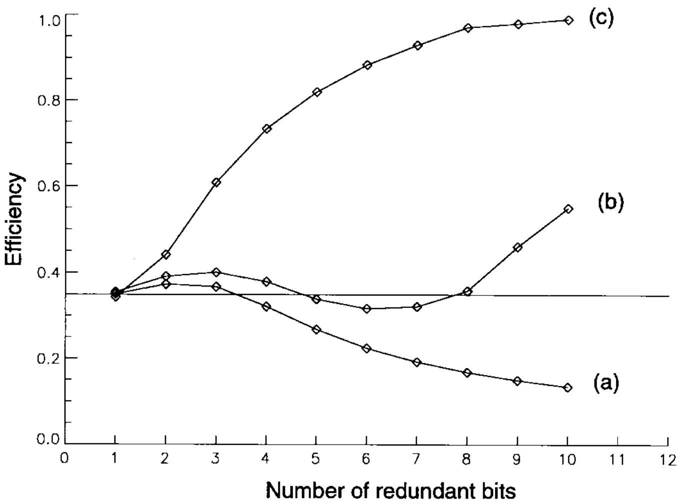

Fig. 2. Simulation results for the random drawing algorithm with *fixed* redundancy 1/128 with different selection criteria (a) MRDS, (b) MMRDS, and (c) MSW.

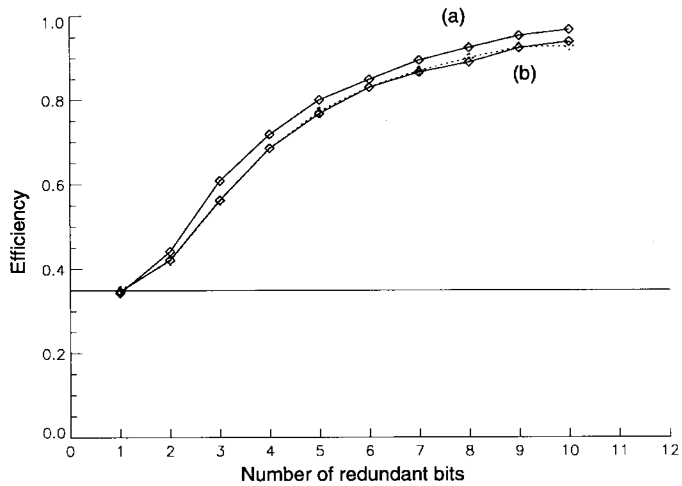

Fig. 3. Simulation results for the random drawing algorithm having *fixed* redundancy 1/128 with (a) the MSW criterion and (b) the MTO criterion. The dotted line shows the results obtained for the implemented encoding scheme using a scramblers with polynomial $x^7 + x + 1$.

the simple MRDS criterion one of the zero-disparity words is randomly chosen and transmitted. As the sum variance of zero-disparity codewords equals $(n+1)/6$, [1] irrespective of the rate of the code, we conclude that the efficiency will asymptotically approach zero. More sophisticated selection criteria, which take account of the running digital sum within the codeword, and not only at the end of the word, may result in increased performance. In order to describe these more sophisticated selection criteria, we introduce the *squared weight*, $w_{\rm sq}$, of a codeword, defined as the sum of the squared RDS values at each bit position of the codeword. The two selection criteria examined are as follows:

1) modified MRDS (MMRDS) criterion: from the codewords with minimal |WRDS|, the one with minimum $w_{\rm sq}$ is selected;
2) minimum squared weight (MSW) criterion: the codeword of minimal $w_{\rm sq}$ is selected from the selection set, irrespective of the WRDS of the codeword.

Fig. 2 shows the simulation results obtained for redundancy 1/128. Simulations of codes with other values of the redundancy produced similar results. From the curves, we infer the following.

- The MRDS method wastes the opportunity offered by the broader selection sets. By properly selecting the codeword from the ones with minimal |WRDS|, the efficiency of the MMRDS scheme tends to unity.
- As indicated by the curve of the MSW criterion, the best codewords do *not* necessarily minimize the |WRDS|. Selecting the codeword with minimal squared weight clearly results in more efficient codes.

Based on the above observations, we searched for a criterion that is simple to implement while its efficiency approaches that of the MSW criterion. The outcome is described in the next section.

A. The Minimum Threshold Overrun Criterion

Our objective, in this section, is to construct a selection criterion which takes into account the RDS values within the codeword while having a structure that is also easy to implement. The proposed selection scheme, termed *minimum threshold overrun* (MTO) criterion, utilizes the parameter "RDS threshold," denoted by M, $M > 0$. The MTO penalty is simply the number of times the absolute value of the running digital sum within a word is larger than M. As the squaring operation needed for the MSW criterion is avoided, the implementation of the MTO criterion is not more complex than the MRDS method. The codeword with minimum penalty is transmitted. If two or more codewords have the same penalty, one of them is chosen randomly and transmitted. This procedure does not seriously deteriorate the performance as it is fairly improbable that two or more codewords in the selection set have the same penalty value. Fig. 3, curve (b), shows simulation results obtained with the MTO criterion. Optimal values of the threshold M were found by trial and error. We can see that the MTO criterion is only slightly less efficient than the MSW criterion. All results shown so far have been obtained by a simulation program of the random drawing algorithm. As a final check we also conducted simulations with a full-fledged implementation using a scramblers with polynomial $x^7 + x + 1$. Experiments with other scrambler polynomials did not reveal significant differences. The dotted curve, Fig. 3, gives results on the basis of the MTO criterion. The curve shows a nice agreement with results obtained with the random drawing algorithm. As the proof of the pudding we have computed the power spectral density (PSD) of two typical examples. The results are displayed in Fig. 4.

VI. Conclusion

Multimode codes have been mathematically analyzed by introducing a simple random drawing model. We have presented alternative selection criteria and examined their effect on the spectral efficiency. Multimode codes are excellent candidate dc-free codes when both low-spectral content at the low-frequency end and high rate are at a premium. For given rate and proper selection criteria, the spectral content of multimode codes is very close to the minimal content promised

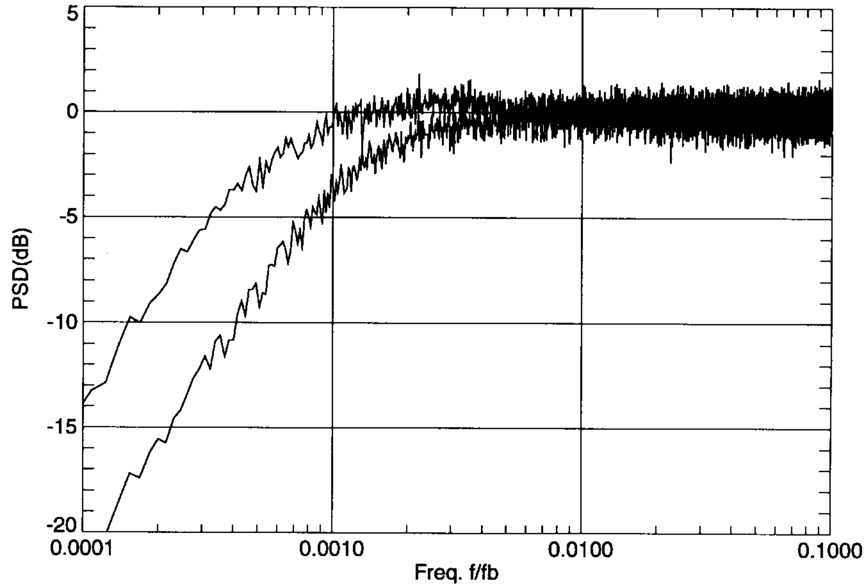

Fig. 4. Spectra of encoded sequences generated by the polarity switch code (upper curve) and multimode code (lower curve). The redundancy is in both cases 1/128. The multimode code has six redundant bits (codeword length is $6 \times 128 = 768$) and it uses the MTO selection criterion.

by information theory.

Appendix

In this Appendix, we will compute the sum variance of sequences encoded with the random drawing model. A codeword $\mathbf{c}$ with binary elements $\{0,1\}$ is translated into the n-tuple $\hat{\mathbf{c}} = (\hat{c}_1, \cdots, \hat{c}_n)$ where $\hat{c}_i = (-1)^{c_i+1}, i = 1, \cdots, n$.

Suppose the ith codeword in the sequence, $\mathbf{c}^{(i)} = (c_1^{(i)}, \cdots, c_n^{(i)})$, starts with initial RDS $Z^{(i)} = z_0$. The RDS at the jth symbol position of $\mathbf{c}^{(i)}$, denoted by z_j, equals

$$z_j = z_0 + \sum_{m=1}^{j} \hat{c}_m^{(i)}, \quad 1 \le j \le n.$$

The running sum variance at the jth position, given z_0, equals

$$\begin{aligned} E\{z_j^2 \mid z_0\} &= E\left\{\left(z_0 + \sum_{m=1}^{j} \hat{c}_m^{(i)}\right)^2\right\} \\ &= E\left\{z_0^2 + \sum_{m=1}^{j} \left(\hat{c}_m^{(i)}\right)^2 + 2z_0 \sum_{m=1}^{j} \hat{c}_m^{(i)} \right. \\ &\quad \left. + 2 \sum_{m_1=1}^{j-1} \sum_{m_2=m_1+1}^{j} \hat{c}_{m_1}^{(i)} \hat{c}_{m_2}^{(i)} \right\} \end{aligned}$$

where the operator $E\{.\}$ averages over all codewords $\mathbf{c}^{(i)}$ that start with an initial RDS z_0. As the source population of codewords is the full set of vectors of nonpositive (nonnegative) disparity, the expectations $E\{\hat{c}_{m_1}^{(i)} \hat{c}_{m_2}^{(i)}\}$ and $E\{\hat{c}_{m_1}^{(i)}\}$, $m_1 \neq m_2$, are independent of the symbol positions m_1 and m_2. For the sake of convenience, we use the shorthand notation $E\{\hat{c}_{m_1}^{(i)}\} = \mu$ and $E\{\hat{c}_{m_1}^{(i)} \hat{c}_{m_2}^{(i)}\} = r_0, 1 \le m_1, m_2 \le n$, $m_1 \neq m_2$. Substitution yields the running sum variance at the jth symbol position

$$E\{z_j^2 \mid z_0\} = z_0^2 + j + 2j\mu z_0 + j(j-1)r_0. \tag{8}$$

The sum variance of a codeword starting with initial RDS z_0, designated by $s^2 \mid z_0$, is found by averaging the running digital sum variance over all n symbol positions of the codeword or

$$\begin{aligned} s^2 \mid z_0 &= \frac{1}{n} \sum_{j=1}^{n} E\{z_j^2 \mid z_0\} \\ &= z_0^2 + \frac{n+1}{2} + \mu(n+1)z_0 + \frac{1}{3}(n^2-1)r_0. \end{aligned}$$

The probability that a codeword starts with an RDS $z_0 = i, i \in \mathcal{S}_K$ equals the stationary probability π_i, so that by taking the probability into account that a codeword starts with RDS z_0 and averaging over all initial states in $\mathcal{S}_K$, the following expression is found for the sum variance s^2:

$$s^2 = E\{z_0^2\} + \frac{n+1}{2} + \frac{1}{3}(n^2-1)r_0 + (n+1)\mu \sum_{i \in \mathcal{S}_K} i\pi_i. \tag{9}$$

The variance of the initial sum values, $E\{z_0^2\}$, equals

$$E\{z_0^2\} = \sum_{i \in \mathcal{S}_K} i^2 \pi_i.$$

The quantity μ can be estimated by noting the periodicity, i.e., $E\{z_0^2\} = E\{z_n^2\}$. Evaluating (8) yields

$$E\{z_n^2 \mid z_0\} = z_0^2 + n + 2n\mu z_0 + n(n-1)r_0$$

and after averaging, where the the probability of starting with an initial RDS z_0 is taken into account, we obtain

$$E\{z_n^2\} = E\{z_0^2\} + n + n(n-1)r_0 + 2n\mu \sum_{i \in \mathcal{S}_K} i\pi_i$$

so that with $E\{z_n^2\} = E\{z_0^2\}$ we find

$$\mu \sum_{i \in \mathcal{S}_K} i\pi_i = -\frac{1}{2}[1 + (n-1)r_0].$$

Substitution in (9) yields

$$s^2 = E\{z_0^2\} - \frac{1}{6}(n^2-1)r_0. \tag{10}$$

1) Computation of the Correlation: We next calculate the correlation $r_0 = E\{\hat{c}_{m_1} \hat{c}_{m_2}\}$ of the symbols at the m_1th and the m_2th symbol position within the same codeword. It is obvious that $E\{\hat{c}_{m_1}\hat{c}_{m_1}\} = 1$. If $m_1 \neq m_2$, some more work is needed. In that case,

$$\begin{aligned} E\{\hat{c}_{m_1}\hat{c}_{m_2}\} &= \mathcal{P}(\hat{c}_{m_1} = \hat{c}_{m_2}) - \mathcal{P}(\hat{c}_{m_1} \neq \hat{c}_{m_2}) \\ &= 1 - 2\mathcal{P}(\hat{c}_{m_1} \neq \hat{c}_{m_2}),\ m_1 \neq m_2. \end{aligned} \tag{11}$$

Assume a codeword $\mathbf{c}$ to be of disparity $2k$. Then the probability that a symbol at position m_1 in the codeword equals 1 is

$$\mathcal{P}\big(\hat{c}_{m_1} = 1 \mid Z^{(i+1)} = Z^{(i)} + 2k\big) = \frac{1}{n}\Big(\frac{n}{2} + k\Big).$$

The probability that another symbol at position $m_2 \neq m_1$ within the same codeword equals -1 is

$$\begin{aligned} &\mathcal{P}\big(\hat{c}_{m_2} = -1 \mid c_{m_1} = 1, Z^{(i+1)} = Z^{(i)} + 2k\big) \\ &\quad = \frac{1}{n-1}\Big(\frac{n}{2} - k\Big). \end{aligned}$$

Hence

$$\mathcal{P}\big(\hat{c}_{m_1} \neq \hat{c}_{m_2} \mid Z^{(i+1)} = Z^{(i)} + 2k\big) = \frac{n^2 - 4k^2}{2n(n-1)}$$

and (11) yields the correlation for codewords of disparity $2k$

$$\begin{aligned} r(k) &= E\big\{\hat{c}_{m_1}\hat{c}_{m_2} \mid Z^{(i+1)} = Z^{(i)} + 2k\big\} \\ &= \frac{1}{n-1}\left(\frac{4}{n}k^2 - 1\right). \end{aligned}$$

Using the above, we find that

$$\begin{aligned} r_0 &= E\{r(k)\} \\ &= \sum_{j \in \mathcal{S}_K} \pi_j \sum_{l \in \mathcal{S}_K} \mathcal{P}(Z^{(i+1)} = l \mid Z^{(i)} = j)\, r\left(\frac{l-j}{2}\right) \\ r_0 &= \sum_{j \in \mathcal{S}_K} \pi_j \sum_{l \in \mathcal{S}_K} t_{j,l} \frac{1}{n-1}\left[\frac{4}{n}\left(\frac{l-j}{2}\right)^2 - 1\right] \\ &= \frac{1}{n-1}\left\{\frac{1}{n}\left[\sum_{j \in \mathcal{S}_K} \pi_j \sum_{l \in \mathcal{S}_K} t_{j,l}(l-j)^2\right] - 1\right\}. \end{aligned} \tag{12}$$

The sum variance can be determined using (10) and (12)

$$\begin{aligned} s^2 = \sum_{j \in \mathcal{S}_K} \pi_j j^2 - \frac{1}{6}(n^2 - 1) r_0 &= \sum_{j \in \mathcal{S}_K} \pi_j j^2 \\ - \frac{n+1}{6n}\left[\sum_{j \in \mathcal{S}_K} \pi_j \sum_{l \in \mathcal{S}_K} t_{j,l}(l-j)^2\right] &+ \frac{n+1}{6}. \end{aligned} \tag{13}$$

Acknowledgment

The authors wish to thank I. Fair for his valuable remarks that helped improve the contents of this paper.

References

[1] K. A. S. Immink, *Coding Techniques for Digital Recorders*. Englewood Cliffs, NJ: Prentice-Hall International, 1991.
[2] S. Fukuda, Y. Kojima, Y. Shimpuku, and K. Odaka, "8/10 modulation codes for digital magnetic recording," *IEEE Trans. Magn.*, vol. MAG-22, pp. 1194–1196, Sept. 1986.
[3] A. X. Widmer and P. A. Franaszek, "A dc-balanced, partitioned-block, 8b/10b transmission code," *IBM J. Res. Develop.*, vol. 27, no. 5, pp. 440–451, Sept. 1983.
[4] H. Yoshida, T. Shimada, and Y. Hashimoto, "8-9 block code: A dc-free channel code for digital magnetic recording," *SMPTE J.*, vol. 92, pp. 918–922, Sept. 1983.
[5] F. K. Bowers, U.S. Patent 2 957 947, 1960.
[6] D. E. Knuth, "Efficient balanced codes," *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 51–53, Jan. 1986. See also P. S. Henry, "Zero disparity coding system," U.S. Patent 4 309 694, Jan. 1982.
[7] H. Hollmann and K. A. S. Immink, "Performance of efficient balanced codes," *IEEE Trans. Inform. Theory*, vol. 37, pp. 913–918, May 1991.
[8] J. Justesen, "Information rates and power spectra of digital codes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 457–472, May 1982.
[9] I. J. Fair, W. D. Gover, 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, pp. 289–297, Feb. 1991.
[10] R. H. Deng and M. A. Herro, "DC-free coset codes," *IEEE Trans. Inform. Theory*, vol. 34, pp. 786–792, July 1988.
[11] A. Kunisa, S. Takahashi, and N. Itoh, "Digital modulation method for recordable digital video disc," in *Proc. 1996 IEEE Int. Conf. Consumer Electron.*, June 1996, pp. 418–419.

Kees A. Schouhamer Immink (M'81–SM'86–F'90) received the M.S. and Ph.D. degrees from the Eindhoven University of Technology, Eindenhoven, The Netherlands.

He joined the Philips Research Laboratories, Eindhoven, in 1968, where he currently holds the position of Research Fellow. He has contributed to the design and development of a wide variety of digital consumer-type audio and video-recorders such as the compact disc, compact disc video, R-DAT, DCC, and DVD. He holds 32 U.S. patents and has written numerous papers in the field of coding techniques for optical and magnetic recorders.

Dr. Immink is the Chairman of the IEEE Benelux Chapter on Consumer Electronics, a Governor of the Audio Engineering Society (AES), and a Governor of the IEEE Information Theory Society. He was named a Fellow of the AES, SMPTE, and IEE. Furthermore, he was recognized and awarded with the AES Silver Medal in 1992, the IEE Sir Thomson Medal in 1993, the SMPTE Poniatoff Gold Medal in 1994, and the IEEE Ibuka Consumer Electronics Award in 1996. He is a member of the Royal Netherlands Academy of Arts and Sciences.

Levente Pátrovics was born in Budapest, Hungary. He received the M.Sc. degree in electrical engineering and computer science in 1994 from the Technical University of Budapest, Hungary. From May to December 1995, he was working towards the Ph.D. degree at Philips Research Laboratories, Eindhoven, The Netherlands.

His research interests are in information theory, particular construction, and analysis of constrained codes. Currently, he is with Lufthansa Systems, Hungaria Kft.