Minimum Pearson Distance Detection for Multi-Level Channels with Gain and/or Offset Mismatch

Kees A. Schouhamer Immink, *Fellow, IEEE,* and Jos H. Weber, *Senior Member, IEEE*

Abstract - *The performance of certain transmission and storage channels, such as optical data storage and Non-Volatile Memory (Flash), is seriously hampered by the phenomena of unknown offset (drift) or gain. We will show that Minimum Pearson Distance (MPD) detection, unlike conventional Minimum Euclidean Distance detection, is immune to offset and/or gain mismatch. MPD detection is used in conjunction with T-constrained codes that consist of q-ary codewords, where in each codeword T reference symbols appear at least once. We will analyze the redundancy of the new q-ary coding technique, and compute the error performance of MPD detection in the presence of additive noise. Implementation issues of MPD detection will be discussed, and results of simulations will be given.*

Key words: Constant composition code, permutation code, rank modulation, flash memory, digital optical data storage, recording, Non-Volatile Memory, NVM, mismatch, adaptive equalisation, fading, Pearson distance, Euclidean distance, fading.

I. Introduction

We consider a communication codebook, S, of chosen q-ary codewords $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ over the q-ary alphabet $\mathcal{Q} = \{0, 1, \ldots, q-1\}$, $q \geq 2$, where n, the *length* of $\boldsymbol{x}$, is a positive integer. Usually it is assumed that the primary impairment of a sent codeword $\boldsymbol{x}$ is additive Gaussian noise, but here, however, it is assumed that the received word $\boldsymbol{r} = a(\boldsymbol{x}+\boldsymbol{\nu})+b$, $r_i \in \mathbb{R}$, is scaled by an unknown factor, called *gain*, $a \neq 1$, $a > 0$, offsetted by an unknown (dc-)offset $b \neq 0$ (both quantities unknown to both sender and receiver), where a and $b \in \mathbb{R}$, and corrupted by additive noise $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$, $\nu_i \in \mathbb{R}$. We use the shorthand notation $\boldsymbol{x} + b = (x_1 + b, x_2 + b, \ldots, x_n + b)$. The unknown channel gain and offset mismatch may lead to a devastating loss in performance, as shown, for example, in [1].

There are many examples of channels with offset and gain mismatch. In optical data storage, both the gain and offset depend on the reflective index of the disc surface and the dimensions of the written features [2]. Fingerprints on optical discs may result in rapid gain and offset variations of the retrieved signal. In baseband transmission channels, offset may arise as *baseline wander*, which is caused by the attenuation of the low frequencies of the channel (a.c. coupling) [3]. In wireless communication channels, path loss will result in unknown gain, *fading*, of the channel. Reading errors in solid-state (Flash) memories may originate from cell drift in aging devices [4].

In optical disc data storage devices and non-volatile memories, constrained codes, specifically *dc-free* or *balanced* codes, have been used and/or proposed to counter the effects of offset and gain mismatch [1]. Jiang *et al.* [4] addressed a q-ary balanced coding technique, called *rank modulation*, for circumventing the difficulties with flash memories having aging offset levels. Zhou *et al.* [5], Sala *et al.* [6], and Immink [7] investigated the usage of balanced codes for enabling 'dynamic' reading thresholds in non-volatile memories.

Codewords taken from a q-ary balanced code can be retrieved by a minimum Euclidean distance detector, where the detection process of q-ary balanced codewords is *intrinsically resistant* to any offset and/or gain mismatch. Unfortunately, the redundancy price of q-ary balanced codes is unattractively high for small values of n [8].

We propose the *Pearson distance* as an alternative to the Euclidean distance, since it is, as will be shown, intrinsically resistant to offset and gain mismatch. It will be shown that the Pearson distance measure can only be applied to codebooks with special properties, and constrained coding is therefore required. To that end, we propose q-ary T-constrained codes, where T, $0 < T \leq q$, *preferred* or *reference* symbols must appear at least once in every codeword [7]. The redundancy of T-constrained codes is much lower than that of prior art q-ary balanced codes, which makes Minimum Pearson Distance detection in conjunction with T-constrained codes an attractive alternative for practical applications.

We start in Section II with a discussion of the prior art Minimum Euclidean Distance (MED) detection of q-ary balanced codewords. In Section III, we will describe Minimum Pearson Distance (MPD) detection in conjunction with T-constrained codewords, and we will show that the detection performance is independent of gain and offset.

Kees A. Schouhamer Immink is with Turing Machines Inc, Willemskade 15b-d, 3016 DK Rotterdam, The Netherlands. E-mail: immink@turing-machines.com.

Jos Weber is with Delft University of Technology, Delft, The Netherlands. E-mail: j.h.weber@tudelft.nl.

Conventional minimum Euclidean distance detection offers optimum (maximum likelihood) error performance for the 'ideal' noisy channel, so Minimum Pearson Distance detection in the presence of additive noise will, by necessity, show a somewhat lessened noise immunity. In Section IV, we will analyze the error performance of MPD detection in the presence of additive Gaussian noise. Implementation issues of MPD detection are discussed in Section V. In Section VI, we will describe our conclusions.

II. Prior Art

Before we take a look at the characteristics of the proposed technology, we will discuss prior-art detection schemes that are based on the classical Euclidean distance metric. The codeword $\boldsymbol{x} \in S$ is sent and received as $\boldsymbol{r} = \boldsymbol{x} + \boldsymbol{\nu}$, where $\boldsymbol{\nu}$ is additive Gaussian noise $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$. For each codeword $\hat{\boldsymbol{x}}$ in the codebook S the receiver computes the (squared) Euclidean distance

$$\delta'(\boldsymbol{r}, \hat{\boldsymbol{x}}) = \sum_{i=1}^{n} (r_i - \hat{x}_i)^2 \tag{1}$$

between the received signal vector $\boldsymbol{r}$ and the codeword $\hat{\boldsymbol{x}}$. After exhaustively computing all distances, the receiver decides that the codeword, $\boldsymbol{x}_o$, with minimum Euclidean distance to the received vector, $\boldsymbol{r}$, is the codeword sent, or in succinct form

$$\boldsymbol{x}_o = \arg\min_{\hat{\boldsymbol{x}} \in S} \delta'(\boldsymbol{r}, \hat{\boldsymbol{x}}). \tag{2}$$

For a mismatch channel, we simply find

$$\delta'(\boldsymbol{r}, \hat{\boldsymbol{x}}) = \sum_{i=1}^{n} (r_i - \hat{x}_i)^2 = \sum_{i=1}^{n} (a x'_i + b - \hat{x}_i)^2$$

or

$$\begin{aligned} \delta'(\boldsymbol{r}, \hat{\boldsymbol{x}}) &= -2a \sum x'_i \hat{x}_i \\ &+ \sum (a x'_i + b)^2 - 2b \sum \hat{x}_i + \sum \hat{x}_i^2, \end{aligned} \tag{3}$$

where $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\nu}$. Mismatch may lead to a warping of the Euclidean distance measure, which will lead to a loss in noise margin or even to a complete loss of the codeword. Figure 1 shows results of computations and simulations of the word error rate (WER) of a communication system using minimum Euclidean distance detection with parameters $q = 4$ and $n = 8$ a) without mismatch, and b) with a gain, $a = 1.07$, and offset, $b = 0.07$, mismatch. We may notice that the error performance is seriously affected by the mismatch. The third curve shows the error performance of a new detection scheme proposed in Section III that offers intrinsic resistance to gain and/or offset mismatch at the cost of a reduced noise margin.

In the prior art, coding techniques and alternative detection methods have been sought to offer solace to the aforementioned loss of performance caused by channel mismatch. For example, it has been known in the art [8] that

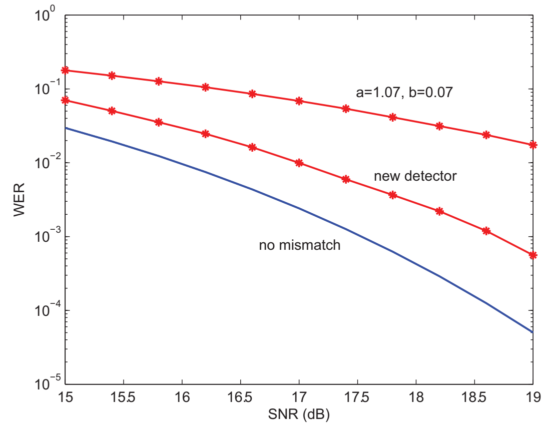

Fig. 1. Word error rate (WER) for $q = 4$, $n = 8$ a) without channel mismatch, b) with channel mismatch $a = 1.07$ and $b = 0.07$, both using minimum Euclidean distance detection, and as a comparison c) for a new detection scheme that is immune to channel mismatch using minimum Pearson distance detection (see Section III).

balanced codes in conjunction with minimum Euclidean distance detection offer intrinsic resistance to gain and/or offset mismatch. By definition, all codewords $\boldsymbol{x}$ in a balanced code have the property that the symbol sum

$$\sum_{i=1}^{n} x_i = a_1$$

and symbol energy

$$\sum_{i=1}^{n} x_i^2 = a_2,$$

are prescribed, where a_1 and a_2 are two positive integers selected by the code designer. Then, we simply find

$$\delta'(\boldsymbol{r}, \hat{\boldsymbol{x}}) = -2a \sum x'_i \hat{x}_i + \sum (a x'_i + b)^2 - 2b a_1 + a_2. \tag{4}$$

The metric consists of one (first) term, $-2a \sum x'_i \hat{x}_i$, dependent of $\hat{\boldsymbol{x}}$ and three terms independent of $\hat{\boldsymbol{x}}$. Clearly, it does not matter for the outcome of the detection process (2) or detection performance if we scale (by a positive constant) or translate the metric function with a constant independent of the variable $\hat{\boldsymbol{x}}$. A scaled or translated version of the metric is *equivalent* to the original version of the metric. Metric equivalence will be denoted by the $\equiv$ sign. From (4), it is clear that in case all codewords satisfy the symbol sum and energy constraints that (1) can be rewritten as (note we presumed $a > 0$)

$$\delta'(\boldsymbol{r}, \hat{\boldsymbol{x}}) \equiv -\sum_{i=1}^{n} x'_i \hat{x}_i. \tag{5}$$

Thus Euclidean distance detection of balanced codes is intrinsically resistant to channel mismatch as (5) is independent of the actual channel gain and offset. The sole disadvantage of the method, which makes it unattractive for practice, is the high redundancy of balanced codes [8].

In the next section, we will introduce the Minimum Pearson Distance detector, and we will show that it is an attractive alternative to the Euclidean distance for the signal detection in the presence of both channel mismatch and additive noise as it also offers intrinsic resistance to channel mismatch.

III. Minimum Pearson Distance detection of T-constrained codes

We will show that the Minimum Pearson Distance detector can only operate unambiguously in case the codebook, S, used satisfies specific constraints, and thus constrained codes are required. To that end, we will introduce a class of codes, called T-constrained codes, that satisfy the requirements of unambiguous detection using the new Pearson-distance-based detector [7]. First we will start with a presentation of the Minimum Pearson Distance detection method. Thereafter, we will discuss the various requirements to be imposed on the constrained code.

A. Minimum Pearson distance detection

It is assumed that an arbitrary codeword $\boldsymbol{x} \in S$ is sent, and that the received word, $\boldsymbol{r}$, is given by $\boldsymbol{r} = a(\boldsymbol{x}+\boldsymbol{\nu})+b$. The Pearson distance, $\delta_2(\boldsymbol{r}, \hat{\boldsymbol{x}})$, between $\boldsymbol{r}$ and $\hat{\boldsymbol{x}}$ is defined by

$$\delta_2(\boldsymbol{r}, \hat{\boldsymbol{x}}) = 1 - \rho_{\boldsymbol{r},\hat{\boldsymbol{x}}}, \tag{6}$$

where

$$\rho_{\boldsymbol{r},\hat{\boldsymbol{x}}} = \frac{\sum_{i=1}^{n}(r_i - \overline{r})(\hat{x}_i - \overline{\hat{x}})}{\sigma_r \sigma_{\hat{x}}} \tag{7}$$

is the well-known *(Pearson) correlation coefficient*, and where we define two quantities, namely the *average symbol value* of $\hat{\boldsymbol{x}}$

$$\overline{\hat{x}} = \frac{1}{n}\sum_{i=1}^{n} \hat{x}_i, \tag{8}$$

and the (unnormalized) *symbol value variance* of $\hat{\boldsymbol{x}}$

$$\sigma_{\hat{x}}^2 = \sum_{i=1}^{n}(\hat{x}_i - \overline{\hat{x}})^2, \tag{9}$$

where it is assumed that both codewords $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ are taken from a judiciously chosen codebook S, whose properties will be explained in the next subsection. The variance, σ_r^2, and average, $\overline{r}$, of the received vector are defined in the same way. The term $1-\rho_{\boldsymbol{r},\hat{\boldsymbol{x}}}$ is often named the *Pearson distance* between the vectors $\boldsymbol{r}$ and $\hat{\boldsymbol{x}}$. As the Pearson correlation coefficient falls in the interval [-1, 1], the Pearson distance lies in the interval [0, 2].

The detector operates in the same vein as the conventional Euclidean distance detector: the detector computes for all codewords $\hat{\boldsymbol{x}} \in S$ the Pearson distance between $\boldsymbol{r}$ and $\hat{\boldsymbol{x}}$. The receiver decides that the codeword, $\boldsymbol{x}_o$, with minimum Pearson distance to the received vector, $\boldsymbol{r}$, is the codeword sent, or in succinct form

$$\boldsymbol{x}_o = \arg\min_{\hat{\boldsymbol{x}} \in S} \delta_2(\boldsymbol{r}, \hat{\boldsymbol{x}}). \tag{10}$$

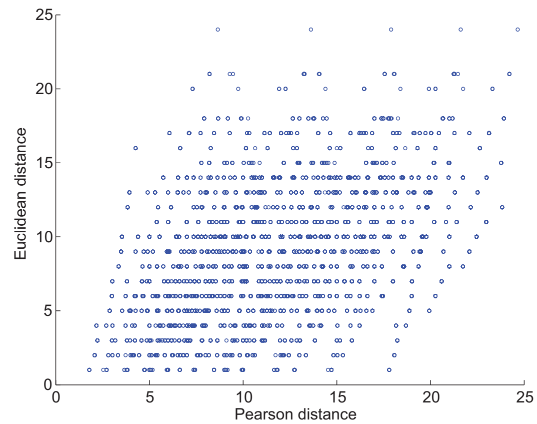

Fig. 2. Scatter diagram of Euclidean distance, $\delta'(\boldsymbol{x}, \hat{\boldsymbol{x}})$, (1) versus Pearson distance, $\delta_2(\boldsymbol{x}, \hat{\boldsymbol{x}})$, (13) for $n = 6$ and $q = 3$.

A key property of the Pearson distance, $1 - \rho_{\boldsymbol{r},\hat{\boldsymbol{x}}}$, is that it is invariant to changes in translation or scale (up to a sign) in the two vectors $\boldsymbol{r}$ and $\hat{\boldsymbol{x}}$. That is,

$$\rho_{\boldsymbol{r},\hat{\boldsymbol{x}}} = \rho_{c_1+c_2\boldsymbol{r},\hat{\boldsymbol{x}}}, \tag{11}$$

where c_1 and $c_2 > 0$ are constants. Clearly, the above property ensures that the detection outcome based on the Pearson distance (10) is intrinsically resistant to offset and gain mismatch.

After some manipulation of (6), we may write down two equivalents of the Pearson distance measure, namely

$$\delta_2(\boldsymbol{r}, \hat{\boldsymbol{x}}) \equiv -\frac{1}{\sigma_{\hat{x}}}\sum_{i=1}^{n} r_i(\hat{x}_i - \overline{\hat{x}}) \tag{12}$$

and

$$\delta_2(\boldsymbol{r}, \hat{\boldsymbol{x}}) \equiv \sum_{i=1}^{n}\left(r_i - \frac{\hat{x}_i - \overline{\hat{x}}}{\sigma_{\hat{x}}}\right)^2. \tag{13}$$

The equivalence of (12) with (6) is straightforward. Eq. (13) can be rewritten as

$$\sum_{i=1}^{n} r_i^2 - \frac{2}{\sigma_{\hat{x}}}\sum_{i=1}^{n} r_i(\hat{x}_i - \overline{\hat{x}}) + \sum_{i=1}^{n}\left(\frac{\hat{x}_i - \overline{\hat{x}}}{\sigma_{\hat{x}}}\right)^2.$$

Using (9) and after deleting the first and third term, we arrive at the equivalence of (12) with (13), and thus of the equivalence of (13) with (6). Eq. (13) shows an interesting relationship with respect to the conventional Euclidean distance (1), where the vector $\hat{\boldsymbol{x}}$ is translated by $\overline{\hat{x}}$ and scaled by $\sigma_{\hat{x}}$.

In Figure 2, we have plotted an illustrative scatter diagram of the Euclidean distance $\delta'(\boldsymbol{x}, \hat{\boldsymbol{x}})$, (1), versus the Pearson distance $\delta_2(\boldsymbol{x}, \hat{\boldsymbol{x}})$, (13), for $n = 6$ and $q = 3$ for almost (see later) all possible vectors $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$. We notice quite some difference between the alternative distance measures. The Pearson distance measure is highly attractive for mismatch channels, but we must clear a few hurdles before we can fruitfully apply it in practice. It is immediate, see

(7), that the Pearson distance is undefined for codewords $\boldsymbol{x}$ with $\sigma_x = 0$. In addition, as mentioned above, we have $\rho_{\boldsymbol{r},\hat{\boldsymbol{x}}} = \rho_{c_1+c_2\boldsymbol{r},\hat{\boldsymbol{x}}}$, so that a Pearson-distance-based detector cannot distinguish between the sent codewords $\boldsymbol{x}$ and $c_1 + c_2\boldsymbol{x}$. We remedy this flaw by using constrained codes: in case $\boldsymbol{x} \in S$, all words $c_1 + c_2\boldsymbol{x}$, where c_1 is any real number and c_2 is any positive real number, are excluded from S. In addition, the q words $\boldsymbol{x} = (a, \ldots, a)$, $a \in \mathcal{Q}$, with $\sigma_x = 0$ must be barred from S. T-constrained codes, to be discussed below, satisfy all these requirements.

B. Description of T-constrained codes

T-constrained codes, presented in [7] for enabling simple dynamic threshold detection of q-ary codewords, consist of q-ary n-length codewords, where T, $0 < T \leq q$, *preferred* or *reference* symbols must appear at least once in a codeword. A set of T-constrained codewords is denoted by S_T. Of specific interest in this paper are two sets of T-constrained codes denoted by S_1 and S_2. The set S_1 has codewords where the symbol '0' appears at least once, and the set S_2 contains codewords where both the symbols '0' and '$q-1$' appear at least once. The size of $|S_1|$ and $|S_2|$, respectively, equal [7]

$$|S_1| = q^n - (q-1)^n, q > 1. \tag{14}$$

and

$$|S_2| = q^n - 2(q-1)^n + (q-2)^n, q > 1. \tag{15}$$

For the binary case, $q = 2$, we simply find that

$$|S_1| = 2^n - 1$$

(the all-'1' word is deleted), and

$$|S_2| = 2^n - 2$$

(both the all-'1' and all-'0' words are deleted).

Very efficient implementations of high-rate T-constrained codes can be constructed with the nibble replacement method described in [9]. The code is based on an algorithm that recursively removes $w > 0$ disallowed n-symbol q-ary words in a string of L n-symbol codewords while only one q-ary redundant symbol is added. The code rate of the nibble replacement equals $1 - 1/(nL)$, where

$$L \leq \left\lfloor \frac{(q-1)q^{n-1}}{w} \right\rfloor. \tag{16}$$

A conventional n-symbol block code can be constructed with a rate $1 - 1/n$, while the nibble replacement code makes it possible to construct a code with rate $1 - 1/(nL)$. For example, for a binary $T = 1$ code, $n = 10$, $w = 1$, the code rate is (at most) 5119/5120, while for a $T = 2$ code with the same parameters, $w = 2$, the code rate is (at most) 2559/2560. The rate of a conventional code would be 9/10, that is almost 10% more redundant.

The T-constraints imposed imply the following properties of the codewords.

Property A: If $\boldsymbol{x} \in S_1$ then $\boldsymbol{x} + c \notin S_1$ for all non-zero $c \in \mathbb{R}$.

Proof: By definition, $\boldsymbol{x}$ has at least one position, say k, where $x_k = 0$. Hence if $c < 0$, then $x_k + c < 0$, and thus $\boldsymbol{x} + c \notin S_1$. If $c > 0$, then $x_i + c > 0$ for all i, and thus $\boldsymbol{x} + c \notin S_1$ since $\boldsymbol{x} + c$ does not contain the symbol '0'.

Property B: If $\boldsymbol{x} \in S_2$ then $c_1 + c_2\boldsymbol{x} \notin S_2$ for all $c_1, c_2 \in \mathbb{R}$ with $(c_1, c_2) \neq (0, 1)$ and $c_2 > 0$.

Proof: Suppose $c_1 + c_2\boldsymbol{x} \in S_2$. Since $c_2 > 0$ and $x_i \geq 0$ for all i, it follows that $c_1 \leq 0$ in order to have at least one '0' in $c_1 + c_2\boldsymbol{x}$. Since $c_1 < 0$ would result in at least one negative value in $c_1 + c_2\boldsymbol{x}$ (in a position where $\boldsymbol{x}$ has a '0'), it follows that $c_1 = 0$ and thus $c_2 \neq 1$. If $0 < c_2 < 1$, then all symbols in $c_1 + c_2\boldsymbol{x} = c_2\boldsymbol{x}$ are smaller then $q-1$, while if $c_2 > 1$, then at least one value in $c_1 + c_2\boldsymbol{x}$ is larger than $q-1$ (in a position where $\boldsymbol{x}$ has a '$q-1$'). In conclusion, $c_1 + c_2\boldsymbol{x} \notin S_2$.

Property C: $\boldsymbol{x} = c \notin S_2$ for all $c \in \mathbb{R}$.

Proof: All codewords $\boldsymbol{x} \in S_2$ have at least one symbol equal to 0 and at least one symbol equal to $q-1$. Since $q > 1$, we conclude that $\boldsymbol{x} = c \notin S_2$.

The above properties of the codebook S_2 are sufficient to solve the flaws of the Pearson distance measure discussed in the previous section. Firstly, property C guarantees that (6) cannot be singular since $\sigma_x \neq 0, \boldsymbol{x} \in S_2$. Secondly, according to Property B we have for $\boldsymbol{x}$ and $\hat{\boldsymbol{x}} \in S_2$, $\hat{\boldsymbol{x}} \neq c_1 + c_2\boldsymbol{x}$. Then in the noiseless case, $\boldsymbol{\nu} = 0$, the detector operates without errors since for any sent $\boldsymbol{x}$, the detector metric (6) is at a global minimum for $\hat{\boldsymbol{x}} = \boldsymbol{x}$ since according to a well-known property of the Pearson correlation coefficient $\rho_{\boldsymbol{x},\hat{\boldsymbol{x}}} = 1$ only for $\hat{\boldsymbol{x}} = c_1 + c_2\boldsymbol{x}$ so that $\delta_2(\boldsymbol{x}, \boldsymbol{x}) = 0$ and $\delta_2(\boldsymbol{x}, \hat{\boldsymbol{x}}) > 0$, $\hat{\boldsymbol{x}} \neq \boldsymbol{x}$.

C. Offset-mismatch channels

Evidently, Minimum Pearson Distance detection presented above can be used for channels with offset mismatch only ($a = 1$). However, by a slight modification of the Pearson distance measure we may reduce the redundancy of the constrained code, and improve the resistance against additive noise. We propose the distance measure

$$\delta_1(\boldsymbol{r}, \hat{\boldsymbol{x}}) = \sum_{i=1}^{n} (r_i - \hat{x}_i + \overline{\hat{x}})^2, \tag{17}$$

where, as we may notice, we have deleted $\sigma_{\hat{x}}$, in (13). The advantage of the modified distance measure for offset-mismatch channels is that codewords can be drawn from S_1 instead of S_2, which reduces the code redundancy.

We first show that $\delta_1(\boldsymbol{r}, \hat{\boldsymbol{x}})$ is, as claimed, independent of the channel's offset b, and thereafter we show that in the noiseless case $\boldsymbol{\nu} = 0$, $\delta_1(\boldsymbol{x}, \hat{\boldsymbol{x}})$ has a unique minimum at $\hat{\boldsymbol{x}} = \boldsymbol{x} \in S_1$. Subsequently, we will show that for application of (17) it is sufficient to draw codewords from S_1.

We find

$$\begin{aligned}\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) &= \sum_{i=1}^{n}(x'_i + b - \hat{x}_i + \overline{\hat{x}})^2 \\ &= \sum_{i=1}^{n}(x'_i - \hat{x}_i + \overline{\hat{x}})^2 + 2b\sum_{i=1}^{n}(x'_i - \hat{x}_i + \overline{\hat{x}}) + b^2,\end{aligned}$$

where $x'_i = x_i + \nu_i$. By definition (8), we have

$$\sum_{i=1}^{n}(x'_i - \hat{x}_i + \overline{\hat{x}}) = \sum_{i=1}^{n} x'_i,$$

so that

$$\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) \equiv \sum_{i=1}^{n}(x'_i - \hat{x}_i + \overline{\hat{x}})^2. \tag{18}$$

From the above it is immediate that the minimization of $\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}})$ is intrinsically resistant to the channel offset b.

In the noiseless case, $\boldsymbol{\nu} = 0$, the detector must operate error-free, so that for any $\boldsymbol{x}$ the detector metric $\delta_1(\boldsymbol{x},\hat{\boldsymbol{x}})$ must have a unique minimum at $\hat{\boldsymbol{x}} = \boldsymbol{x}$, that is, $\delta_1(\boldsymbol{x},\boldsymbol{x}) < \delta_1(\boldsymbol{x},\hat{\boldsymbol{x}})$ for $\boldsymbol{x} \neq \hat{\boldsymbol{x}}$, and $\boldsymbol{x},\hat{\boldsymbol{x}} \in S_1$. From (17), we conclude that $\delta_1(\boldsymbol{x},\hat{\boldsymbol{x}}) = \delta_1(\boldsymbol{x},\boldsymbol{x})$ is true if for all i

$$x_i = \hat{x}_i - \overline{\hat{x}} = \hat{x}_i + c, \tag{19}$$

where c is any real number. However, Property A guarantees that the above is false for codewords in S_1, and thus $\delta_1(\boldsymbol{x},\boldsymbol{x}) < \delta_1(\boldsymbol{x},\hat{\boldsymbol{x}})$, $\boldsymbol{x} \neq \hat{\boldsymbol{x}}$. Properties B and C are not required, so that it is sufficient that we draw the codewords from S_1, which, clearly, has a positive bearing on the redundancy of the system.

In the next section, we will compute the error performance of the MPD detection method in the presence of additive noise.

IV. Performance Analysis

We will now focus on the detection of codewords conveyed over channels with gain and/or offset mismatch, where we assume that a randomly chosen codeword $\boldsymbol{x} \in S_T$ is sent and received as $\boldsymbol{r} = \boldsymbol{x} + \boldsymbol{\nu}$, where $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$, and ν_i are noise samples with distribution $N(0, \sigma^2)$. Note that we dropped the scalars a and b since, as established above, the MPD detector performance is independent of those parameters.

We will start by computing the detection error performance for the simplest, 'offset only', case, $T = 1$, using (17) and codewords taken from codebook S_1.

A. $T = 1$, *offset mismatch*

We assume $\boldsymbol{x} \in S_1$ is sent, and that the receiver applies (17) to compute the distance between the received vector $\boldsymbol{r} = \boldsymbol{x} + \boldsymbol{\nu}$ for all codewords in S_1. The receiver decides that the codeword $\boldsymbol{x}_o$ was sent if (17) attains its least value for $\hat{\boldsymbol{x}} = \boldsymbol{x}_o$, that is

$$\boldsymbol{x}_o = \arg\min_{\hat{\boldsymbol{x}} \in S_1} \delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) = \arg\min_{\hat{\boldsymbol{x}} \in S_1} \sum_{i=1}^{n}(r_i - \hat{x}_i + \overline{\hat{x}})^2. \tag{20}$$

We have $r_i = x_i + \nu_i$, so that

$$\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) = \sum_{i=1}^{n}(x_i + \nu_i - \hat{x}_i + \overline{\hat{x}})^2.$$

The analysis is simplified by noticing that

$$\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) \equiv \sum_{i=1}^{n}(x_i + \nu_i - \hat{x}_i + \overline{\hat{x}} - \overline{x})^2.$$

Define $\boldsymbol{e} = \boldsymbol{x} - \hat{\boldsymbol{x}}$ and $\overline{e} = \overline{x} - \overline{\hat{x}}$, then we obtain the much simpler expression

$$\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) \equiv \sum_{i=1}^{n}(e_i - \overline{e} + \nu_i)^2.$$

The detector errs, i.e. $\boldsymbol{x}_o \neq \boldsymbol{x}$, if there is at least one codeword $\hat{\boldsymbol{x}} \in S_1, \hat{\boldsymbol{x}} \neq \boldsymbol{x}$, such that

$$\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) < \delta_1(\boldsymbol{r},\boldsymbol{x}),$$

or

$$\sum_{i=1}^{n}(e_i - \overline{e} + \nu_i)^2 < \sum_{i=1}^{n}\nu_i^2 \tag{21}$$

or

$$2\sum_{i=1}^{n}\nu_i(e_i - \overline{e}) + \sum_{i=1}^{n}(e_i - \overline{e})^2 < 0. \tag{22}$$

The left-hand side of (22) is a stochastic variable with distribution $N(\alpha_1, \beta_1\sigma^2)$, where

$$\alpha_1 = \sum_{i=1}^{n}(e_i - \overline{e})^2$$

and

$$\beta_1 = 4\sum_{i=1}^{n}(e_i - \overline{e})^2.$$

We define the square of the distance between the vectors $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ by

$$d_1^2(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{4\alpha_1^2}{\beta_1} = \sum_{i=1}^{n}(e_i - \overline{e})^2.$$

The probability that $\delta_1(\boldsymbol{r},\hat{\boldsymbol{x}}) < \delta_1(\boldsymbol{r},\boldsymbol{x})$ equals

$$Q\left(\frac{d_1(\boldsymbol{x},\hat{\boldsymbol{x}})}{2\sigma}\right),$$

where the Q-function is defined by

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{u^2}{2}}\,du.$$

The word error rate (WER) over all coded sequences $\boldsymbol{x}$ is now upperbounded by

$$\text{WER} < \frac{1}{|S_1|}\sum_{\boldsymbol{x} \in S_1}\sum_{\hat{\boldsymbol{x}} \neq \boldsymbol{x}} Q\left(\frac{d_1(\boldsymbol{x},\hat{\boldsymbol{x}})}{2\sigma}\right). \tag{23}$$

Define the square of the minimum distance between any possible pair of codewords in S_1 by

$$d_{\min,1}^2 = \min_{\substack{\boldsymbol{x},\hat{\boldsymbol{x}} \in S_1 \\ \boldsymbol{x} \neq \hat{\boldsymbol{x}}}} d_1^2(\boldsymbol{x},\hat{\boldsymbol{x}}) = \min_{\boldsymbol{e} \neq 0}\sum_{i=1}^{n}(e_i - \overline{e})^2. \tag{24}$$

Then, for asymptotically large signal-to-noise-ratio's, i.e. for $\sigma << 1$, the word error rate is overbounded by [10]

$$\text{WER} < N_1 Q(\frac{d_{\min,1}}{2\sigma}), \tag{25}$$

where N_1 is the average number of pairs of codewords (neighbors) at minimum distance, $d_{\min,1}$. The minimum distance, $d_{\min,1}$, is called (asymptotic) *coding loss*. We simply find

$$d_{\min,1} = \sqrt{1 - \frac{1}{n}}. \tag{26}$$

The computation of the average number neighboring pairs of codewords $\hat{\boldsymbol{x}}$ of $\boldsymbol{x}$ in S_1 at minimum distance $d_{\min,1}$ is a formidable combinatorics exercise. We may, however, compute an approximation to N_1 by using the following observation. Clearly, a pair of codewords $\hat{\boldsymbol{x}}$ of $\boldsymbol{x}$ at unity Euclidean distance $\delta'(\hat{\boldsymbol{x}}, \boldsymbol{x})$ is also a pair of codewords at minimum distance $d_{\min,1}$. For a full set of q^n codewords, a neighbor at unity Euclidean distance from $\boldsymbol{x}$ is obtained by adding or subtracting a 'one' from each of the n symbols in $\boldsymbol{x}$, unless the symbol equals 0 or $(q-1)$, then we can only add a one (if $x_1 = 0$) or subtract a one (if $x_i = q-1$). Then the average number of pairs of codewords at unity Euclidean distance equals $2n(q-1)/q$. Since the code set S_1 is obtained by deleting a relatively small number of codewords from the full set, we will approximate the average number of neighbors, N_1, at minimum distance $d_{\min,1}$ in S_1 by

$$N_1 \approx \frac{2n(q-1)}{q}. \tag{27}$$

For small values of n and $q > 2$ the computation of N_1 is amenable by exhaustively computing the distance between each pair of codewords. We found that in the range of (small) values of n and q investigated that the approximation is of sufficient accuracy for engineering applications.

By combining (25), (26), and (27), we obtain an upperbound to the word error rate (WER)

$$\text{WER} < \frac{2n(q-1)}{q} Q\left(\frac{1}{2\sigma}\sqrt{1-\frac{1}{n}}\right). \tag{28}$$

Figure 3 shows the WER, computed using the above upperbound (28), as a function of the signal-to-noise-ratio (SNR), where the quantity SNR is defined by

$$\text{SNR(dB)} = -20\log_{10}\sigma.$$

Results are presented in Figure 3 for two binary cases, $n = 4$ and $n = 12$. Computer simulations of the detection process were conducted for assessing the accuracy of (28). The dotted lines show the results obtained by computer simulations, which compare fairly well with (28).

B. Case $T = 2$, offset and gain mismatch

We will now examine the error performance of the Pearson-distance-based detection scheme. The receiver uses the Pearson distance (6) for the evaluation of the received word, where we assume that $\boldsymbol{x} \in S_2$ is sent, and received as $\boldsymbol{r} = \boldsymbol{x} + \boldsymbol{\nu}$. The receiver errs if there is at least one codeword $\hat{\boldsymbol{x}} \neq \boldsymbol{x}, \hat{\boldsymbol{x}} \in S_2$, such that

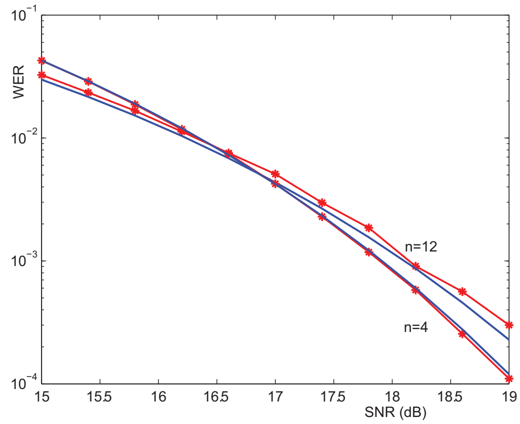

Fig. 3. Word error rate (WER) of offset-resistant detection as a function of the signal-to-noise ratio (SNR) for the binary case, $q = 2$, and $n = 4$ and $n = 12$ computed using upperbound (28). The dotted lines show results of computer simulations.

$$\delta_2(\boldsymbol{r}, \hat{\boldsymbol{x}}) < \delta_2(\boldsymbol{r}, \boldsymbol{x}).$$

After working out using (12), we obtain

$$-\sum_{i=1}^{n} r_i(\frac{\hat{x}_i - \overline{\hat{x}}}{\sigma_{\hat{x}}}) < -\sum_{i=1}^{n} r_i(\frac{x_i - \overline{x}}{\sigma_x}).$$

We substitute $r_i = x_i + \nu_i$, and obtain

$$\sum_{i=1}^{n}(x_i + \nu_i)(a_i - \hat{a}_i) < 0, \tag{29}$$

where

$$a_i = \frac{x_i - \overline{x}}{\sigma_x}$$

and

$$\hat{a}_i = \frac{\hat{x}_i - \overline{\hat{x}}}{\sigma_{\hat{x}}}.$$

The left-hand side of inequality (29) is a stochastic variable with distribution $N(\alpha_2, \beta_2\sigma^2)$, where

$$\alpha_2 = \sum_{i=1}^{n} x_i(a_i - \hat{a}_i)$$

and

$$\beta_2 = \sum_{i=1}^{n}(a_i - \hat{a}_i)^2.$$

Since

$$\sum_{i=1}^{n} a_i = \sum_{i=1}^{n} \hat{a}_i = 0,$$

we have

$$\begin{aligned}\alpha_2 &= \sum_{i=1}^{n} x_i(a_i - \hat{a}_i)\\ &= \sum_{i=1}^{n}(x_i - \overline{x})(a_i - \hat{a}_i)\\ &= \sum_{i=1}^{n}(x_i - \overline{x})\left(\frac{x_i - \overline{x}}{\sigma_x} - \frac{\hat{x}_i - \overline{\hat{x}}}{\sigma_{\hat{x}}}\right)\\ &= \sigma_x(1 - \rho_{\boldsymbol{x},\hat{\boldsymbol{x}}}). \qquad (30)\end{aligned}$$

In a similar fashion, we find

$$\beta_2 = 2(1 - \rho_{\boldsymbol{x},\hat{\boldsymbol{x}}}). \qquad (31)$$

Define the square of the distance, $d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}})$, between the codewords $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ by

$$d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{4\alpha_2^2}{\beta_2} = 2\sigma_x^2(1 - \rho_{\boldsymbol{x},\hat{\boldsymbol{x}}}). \qquad (32)$$

The distance, $d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}})$, is not symmetric in the vectors $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$, that is, in general, $d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}}) \neq d_2^2(\hat{\boldsymbol{x}}, \boldsymbol{x})$. The quantity, $d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}})$, is the product of the Pearson distance between the vectors $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$, which is symmetric in $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$, and the word variance of the sent word $\boldsymbol{x}$. Clearly, sent words $\boldsymbol{x}$ that have a small symbol value variance are more prone to error than words with a large symbol value variance. The word error rate (WER) over all coded sequences $\boldsymbol{x}$ is upperbounded by

$$\text{WER} < \frac{1}{|S_2|} \sum_{\boldsymbol{x} \in S_2} \sum_{\hat{\boldsymbol{x}} \neq \boldsymbol{x}} Q\left(\frac{d_2(\boldsymbol{x}, \hat{\boldsymbol{x}})}{2\sigma}\right). \qquad (33)$$

We define the minimum squared distance between any pair of codewords in S_2, $d^2_{\min,2}$, by

$$\begin{aligned}d^2_{\min,2} &= \min_{\substack{\boldsymbol{x},\hat{\boldsymbol{x}} \in S_2\\ \boldsymbol{x} \neq \hat{\boldsymbol{x}}}} d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}})\\ &= \min_{\substack{\boldsymbol{x},\hat{\boldsymbol{x}} \in S_2\\ \boldsymbol{x} \neq \hat{\boldsymbol{x}}}} 2\sigma_x^2(1 - \rho_{\boldsymbol{x},\hat{\boldsymbol{x}}}). \qquad (34)\end{aligned}$$

At high signal-to-noise ratios, the word error rate (WER) is overbounded by [10]

$$\begin{aligned}\text{WER} &< N_2 Q(\frac{\alpha_2}{\sqrt{\beta_2}\sigma})\\ &= N_2 Q(\frac{d_{\min,2}}{2\sigma}), \qquad (35)\end{aligned}$$

where N_2 is the (average) number of nearest neighbors. We will now compute the minimum distance, $d^2_{\min,2}$, and the number of nearest neighbors, N_2.

In order to find the minimum distance, $d_{\min,2}$, we must compute the distance between any pair of codewords $\boldsymbol{x}, \hat{\boldsymbol{x}} \in S_2$, which implies an exhaustive domain search of size around $|S_2|^2$. We may significantly reduce the number of evaluations by the following observations.

It is not difficult to see that any permutation of the symbols of a T-constrained n-length q-ary codeword yields again a T-constrained codeword. A constant composition code is a set of n-length q-ary codewords where the numbers of occurrences of the symbols within a codeword is the same for each codeword [11]. These specialize to constant weight codes in the binary case, and permutation codes in the case that each symbol occurs exactly once.

Define the *composition vector* $\boldsymbol{w}(\boldsymbol{x}) = (w_0, \ldots, w_{q-1})$ of $\boldsymbol{x}$, where the q entries w_j, $j \in \mathcal{Q}$, of $\boldsymbol{w}$ indicate the number of occurrences of the symbol $j \in \mathcal{Q}$ in $\boldsymbol{x}$. Thus for a sequence $\boldsymbol{x}$, we denote the number of appearances of the symbol j by

$$w_j(\boldsymbol{x}) = |\{i : x_i = j\}| \text{ for } j = 0, 1, \ldots, q-1. \qquad (36)$$

It is immediate that $\sum w_j = n$ and $w_j \in \{0, \ldots, n\}$. A constant composition code will be denoted by $S_{\boldsymbol{w}}$, where $\boldsymbol{w}$ is the composition vector that characterizes the code.

Clearly, the codebook S_T is the union of K constant composition codes, denoted by $S_{\boldsymbol{w}_i}$, $1 \leq i \leq K$, whose composition vector

$$\boldsymbol{w}_i = (w_0, \ldots, w_j, \ldots, w_{q-1})_i, \quad 1 \leq i \leq K,$$

has $w_0 > 0$ for $T = 1$, and both $w_0 > 0$ and $w_{q-1} > 0$ for $T = 2$. The number, K, of constant composition codes equals

$$K = \binom{n + q - 1 - T}{q - 1}. \qquad (37)$$

For the binary case, $q = 2$, we simply find

$$K = n + 1 - T.$$

Each constant composition code is characterized by one codeword of the code, called *pivot word*, denoted by $\boldsymbol{x}_{p_i}$, $1 \leq i \leq K$. The pivot word $\boldsymbol{x}_{p_i}$ is, by definition, the largest word in the lexicographical ordering (the largest symbols first) of the codewords in $S_{\boldsymbol{w}_i}$. Thus, for a pivot word we have $\boldsymbol{x}_{p_i} \in S_{\boldsymbol{w}_i}$ and $\boldsymbol{x}_{p_i} = (x_{p_i,1}, x_{p_i,2}, \ldots, x_{p_i,n})$, where $x_{p_i,1} \geq x_{p_i,2} \geq x_{p_i,3} \ldots \geq x_{p_i,n-1} \geq x_{p_i,n}$, $1 \leq i \leq K$. By definition we have for $T = 2$, $x_{p_i,1} = q - 1$ and $x_{p_i,n} = 0$, $1 \leq i \leq K$. The remaining codewords in S_T consist of all distinct sequences that can be formed by permuting the order of the n symbols that form the K pivot words $\boldsymbol{x}_{p_i}$, $1 \leq i \leq K$.

Let the codewords $\boldsymbol{x} \in S_{\boldsymbol{w}_i}$ and $\hat{\boldsymbol{x}} \in S_{\boldsymbol{w}_j}$, $i \neq j$, then we have

$$d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}}) = 2\sigma_x^2(1 - \rho_{\boldsymbol{x},\hat{\boldsymbol{x}}}) = A - B\sum_{i=1}^{n} x_i \hat{x}_i,$$

where

$$A = 2\sigma_x^2(1 + \frac{n\overline{\boldsymbol{x}}\overline{\hat{\boldsymbol{x}}}}{\sigma_x \sigma_{\hat{x}}})$$

and

$$B = \frac{2\sigma_x}{\sigma_{\hat{x}}}$$

are positive constants. Since the average symbol value and symbol value variance of codewords from the same constant composition code are all equal, the only degree of freedom we have for minimizing $d_2^2(\boldsymbol{x}, \hat{\boldsymbol{x}})$ is permuting the symbols in $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ in such a way that the inner product $\sum_{i=1}^{n} x_i \hat{x}_i$ is maximized. It has been shown by Slepian [12] that $\sum_{i=1}^{n} x_i \hat{x}_i$ is maximized by pairing the largest symbol of $\boldsymbol{x}$ with the largest symbol of $\hat{\boldsymbol{x}}$, the second largest symbol of $\boldsymbol{x}$ with the second largest symbol of $\hat{\boldsymbol{x}}$, etc. Since, by definition, the pivot words are sorted by descending order of the symbol values, we conclude that the minimum inter-subset distance is given by

$$\min_{\boldsymbol{x} \in S_{\boldsymbol{w}_i}, \hat{\boldsymbol{x}} \in S_{\boldsymbol{w}_j}} d_2(\boldsymbol{x}, \hat{\boldsymbol{x}}) = d_2(\boldsymbol{x}_{p_i}, \boldsymbol{x}_{p_j}), \;\; i \neq j.$$

Define the minimum intra-subset distance by

$$d_2(\boldsymbol{x}_{p_i}, \boldsymbol{x}_{p_i}) = \min_{\substack{\boldsymbol{x}, \hat{\boldsymbol{x}} \in S_{\boldsymbol{w}_i} \\ \boldsymbol{x} \neq \hat{\boldsymbol{x}}}} d_2(\boldsymbol{x}, \hat{\boldsymbol{x}}),$$

then it is immediate that

$$d_{\min,2} = \min_{1 \leq i,j \leq K} d_2(\boldsymbol{x}_{p_i}, \boldsymbol{x}_{p_j}). \tag{38}$$

A second reduction in the number of distance evaluations can be made by observing that

$$d_2(\boldsymbol{x}, \hat{\boldsymbol{x}}) = d_2(q - 1 - \boldsymbol{x}, q - 1 - \hat{\boldsymbol{x}}). \tag{39}$$

The above observation implies that we may limit the distance evaluations to (sent) words $\boldsymbol{x}$, whose number of '$q-1$' symbols is less than the number of '0' symbols. For other values of q, $q > 2$, the pivot words start with a series of '$q-1$' symbols, followed by a series of '$q-2$' symbols etc, and ended by a series of '0' symbols. The series of symbols may be of zero length, except, however, for the symbol values '$q-1$' and '0', where the series must have a length of at least unity.

Figure 4 shows results of computations, where we plotted $-20\log(d_{\min,2})$ (dB) as a function of n with $q = 2$, 3, 4 as a parameter. After perusing the diagram, we notice that the variation of $d_{\min,2}$ as a function of n diminishes with larger values of q. We further notice that with increasing n the minimum distance $d_{\min,2}$ converges to a limiting value. In our computations, we found that (but we could not prove it for all q and n) that the codewords $\boldsymbol{x} = (q-1, q-2, 0, \ldots, 0)$ and $\hat{\boldsymbol{x}} = (q-1, q-1, 0, \ldots, 0)$, and their inverses $(q-1, q-1, \ldots, q-1, 1, 0)$ and $(q-1, q-1, \ldots, q-1, 0, 0)$ are at minimum squared distance $d_{\min,2}^2 = \min 2\sigma_x^2(1 - \rho_{\boldsymbol{x},\hat{\boldsymbol{x}}})$, $\boldsymbol{x}, \hat{\boldsymbol{x}} \in S_2$. For $q = 2$, the codewords (1, 0, 0, ...) and its inverse $(1, \ldots, 1, 0)$ each have $n-1$ nearest neighbors. This also holds for the n permutations of $(1, 0, 0, \ldots)$ plus its inverse $(0, 1, 1, \ldots)$, so that, for $q = 2$, the total number of pairs of codewords at minimum distance equals $2n(n-1)$. For $q > 2$, the codeword $(q-1, q-2, 0, \ldots)$ and its $n(n-1)$ permutations each have only one codeword at minimum distance. The same holds for its inverse (0, 1, $q-1$, $q-1$,...), so that for $q > 2$ there are $2n(n-1)$ pairs of codewords at minimum distance. We conclude with (35) that

$$\text{WER} < \frac{2n(n-1)}{|S_2|} Q(\frac{d_{\min,2}}{2\sigma}), \;\; \sigma << 1. \tag{40}$$

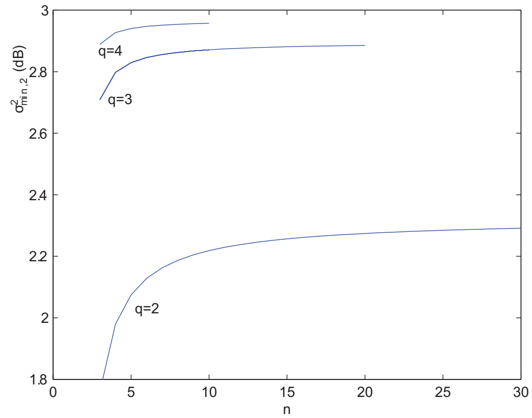

Fig. 4. Minimum squared distance, $-20\log(d_{\min,2})$ (dB), as a function of the codeword length n with $q = 2, 3, 4$ as a parameter. Although lines have been drawn, the curves consist of discrete points.

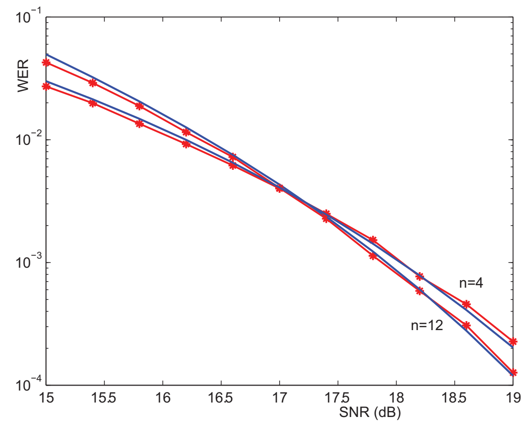

Fig. 5. Word error rate (WER) of offset&gain-resistant detection as a function of the signal-to-noise ratio (SNR) for $q = 2$, $n = 4 and 12$. The diagram shows the error performance computed by upperbound (33) and computer simulations (dotted lines).

Upperbound (40), where it is assumed that minimum distance error events dominate the performance, is far from accurate in the range WER $> 10^{-8}$. A better approximation to the word error rate is given by upperbound (33), where the full distance profile is used. Figure 5 shows the word error rate computed using upperbound (33) for $q = 2$ and $n = 4$, 12, as a function of the signal-to-noise-ratio. The computer simulations (dotted lines) compare fairly well with upperbound (33).

V. Implementation issues

In the above we did not yet address the implementation of the Minimum Pearson Distance detector, and we tacitly assumed that the receiver computes the Pearson distance for all codewords in the codebook before it may give its verdict on the word sent. Below we will show that we

TABLE I
PIVOT WORDS, AVERAGE SYMBOL VALUE, THE SYMBOL VALUE VARIANCE AND THE DISTANCE OF CASE DESCRIBED IN EXAMPLE 1.

i	$\boldsymbol{x}_{p_i}$	$\overline{x}_{p_i}$	$\sigma^2_{x_{p_i}}$	$\delta_2(\boldsymbol{r}, \boldsymbol{x}_{p_i})$
1	(1,0,0,0)	1/4	3/4	5.93
2	(1,1,0,0)	2/4	1	6.23
3	(1,1,1,0)	3/4	3/4	6.74

may reduce the number of distance computations to the number of constant composition subsets, K, that constitute the full set S_T. This means for $q = 2$ and $T = 2$, for example, that instead of $2^n - 2$ for a full set, only $n - 1$ computations of the Pearson distance are required. As discussed earlier, Slepian [12] showed that a *single* permutation modulation code may be very efficiently decoded by applying a sorting algorithm to the entries of the received signal vector. Slepian's method can be extended to our case incorporating a plurality of K constant composition codes. The receiver has tabulated the K pivot words, where the symbols of the pivot words are, as described above, sorted in descending order, i.e., the symbols with the largest values first. Detection of the received vector, $\boldsymbol{r}$, is accomplished by invoking the following two-step procedure.

1) The n symbols of the received word, $\boldsymbol{r}$, are sorted, largest to smallest, in the same way as taught in Slepian's prior art.
2) Compute $\delta_2(\boldsymbol{r}, \boldsymbol{x}_{p_i})$ or $\delta_1(\boldsymbol{r}, \boldsymbol{x}_{p_i})$, $1 \leq i \leq K$, using (12) or (17), for all K pivot words. The receiver decides that a word taken from $S_{\boldsymbol{w}_k}$, $1 \leq k \leq K$, was sent in case the pivot word $\boldsymbol{x}_{p_k}$ is at minimum distance to the received vector $\boldsymbol{r}$. Now that we have ascertained that the sent word is a member of $S_{\boldsymbol{w}_k}$, we can decode the received word by Slepian's procedure for single permutation codes.

Below we will describe a simple example for the gain-and-offset mismatch channel to illustrate the detection routine.

Example *1:* Let $n = 4$ and $q = 2$, then there are 14 codewords in S_2 and 3 subsets of constant composition. Assume the received vector is $\boldsymbol{r}$ = (1.2, 2.3, 0.4, 0.8). First we sort the symbols of $\boldsymbol{r}$ largest to smallest, and obtain (2.3, 1.2, 0.8, 0.4), and compute $\delta_2(\boldsymbol{r}, \boldsymbol{x}_{p_i})$ for each pivot word $\boldsymbol{x}_{p_i}$. Table I lists $\boldsymbol{x}_{p_i}$, $i = 1, 2, 3$, the average symbol value of the pivot words, the symbol value variance of the pivot words, and the distance $\delta_2(\boldsymbol{r}, \boldsymbol{x}_{p_i})$ computed using (13). We conclude that the pivot word (1,0,0,0) has the smallest distance, 5.93, to the received vector $\boldsymbol{r}$, so that the detector, following Slepian's sorting algorithm assigns the largest symbol in $\boldsymbol{r}$ to a '1' and the three other symbols to '0s'. Thus, the receiver decides that the word (0,1,0,0) was sent.

VI. CONCLUSIONS

We have studied coding and detection techniques for q-level channels with gain and/or offset mismatch. We have analyzed a new technique, where codeword detection is based on the Pearson distance. The proposed detection technique is used in conjunction with T-constrained codes consisting of q-ary codewords, where in each codeword T reference symbols appear at least once. We have shown that the new detection technique is intrinsically resistant to offset and/or gain mismatch. The redundancy of T-constrained codes is much lower than that of prior art balanced codes that offer offset and gain mismatch resistant detection, which makes the new codes more attractive for practical applications. We have analyzed the error performance of the Pearson Distance detection technique in the presence of additive noise. Results of computer simulations of the noisy channel have been compared with analytical expressions of the word error rates.

REFERENCES

[1] K.A.S. Immink, 'Coding Methods for High-Density Optical Recording', *Philips J. Res.,* vol. 41, pp. 410-430, 1986.
[2] G. Bouwhuis, J. Braat, A. Huijser, J. Pasman, G. van Rosmalen, and K.A.S. Immink, *Principles of Optical Disc Systems,* Adam Hilger Ltd, Bristol and Boston, 1985.
[3] K.W. Cattermole, 'Principles of Digital Line Coding', *Int. Journal of Electronics,* vol. 55, pp. 3-33, July 1983.
[4] A. Jiang, R. Mateescu, M. Schwartz, and J. Bruck, 'Rank Modulation for Flash Memories', *IEEE Trans. Inform. Theory,* vol. IT-55, no. 6, pp. 2659-2673, June 2006.
[5] H. Zhou, A. Jiang, and J. Bruck, 'Error-correcting schemes with dynamic thresholds in nonvolatile memories', Int. Symposium in Inform. Theory (ISIT), St Petersburg, July 2011.
[6] F. Sala, R. Gabrys, and L. Dolecek, 'Dynamic Threshold Schemes for Multi-Level Non-Volatile Memories'. *IEEE Trans. on Commun.,* pp. 2624-2634, Vol. 61, July 2013.
[7] K.A.S. Immink, 'Coding Schemes for Multi-Level Flash Memories that are Intrinsically Resistant Against Unknown Gain and/or Offset Using Reference Symbols', *Electronics Letters,* vol. 50, pp. 20-22, 2014.
[8] K.A.S. Immink, 'Coding Schemes for Multi-Level Channels with Unknown Gain and/or Offset Using Balance and Energy constraints', IEEE International Symposium on Information Theory, (ISIT), Istanbul, July 2013.
[9] K.A.S. Immink, 'High-Rate Maximum Runlength Constrained Coding Schemes Using Nibble Replacement', *IEEE Trans. Inform. Theory,* pp. 6572-6580, vol. IT-58, No. 10, Oct. 2012.
[10] G.D. Forney Jr., 'Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference', *IEEE Trans. Inform. Theory,* vol. IT-18, pp. 363-378, May. 1972.
[11] W. Chu, C.J. Colbourn, and P. Dukes, 'On Constant Composition Codes', *Discrete Applied Mathematics,* Volume 154, Issue 6, 15 pp. 912-929, April 2006.
[12] D. Slepian, 'Permutation Modulation', *Proc IEEE,* vol. 53, pp. 228-236, March 1965.

Kees Schouhamer Immink (M81-SM86-F90) received his PhD degree from the Eindhoven University of Technology. He was with Philips Research Labs in Eindhoven from 1968 till 1998. In 1998, he founded Turing Machines Inc, a company that has been successful in applying the tenets of information theory to digital data storage and transmission. He is, since 1994, an adjunct professor at the Institute

for Experimental Mathematics, Duisburg-Essen University, Germany.

Immink designed coding techniques of virtually all consumer-type digital audio and video recording products, such as Compact Disc, CD-ROM, CD-Video, Digital Audio Tape recorder, Digital Compact Cassette system, DCC, Digital Versatile Disc, DVD, Video Disc Recorder, and Blu-ray Disc. He received widespread recognition for his many contributions to the technologies of video, audio, and data recording. He received a Knighthood in 2000, a personal Emmy award in 2004, the 1996 IEEE Masaru Ibuka Consumer Electronics Award, the 1998 IEEE Edison Medal, the 1999 AES Gold Medal, the 2004 SMPTE Progress Medal, and the Eduard Rhein Prize for Technology in 2014. He received the Golden Jubilee Award for Technological Innovation by the IEEE Information Theory Society in 1998. He was named a fellow of the IEEE, AES, and SMPTE, and was inducted into the Consumer Electronics Hall of Fame, and elected into the Royal Netherlands Academy of Sciences and the US National Academy of Engineering. He received an honorary doctorate from the University of Johannesburg in 2014. He served the profession as President of the Audio Engineering Society inc., New York, in 2003.

Jos H. Weber (S'87-M'90-SM'00) was born in Schiedam, The Netherlands, in 1961. He received the M.Sc. (in mathematics, with honors), Ph.D., and MBT (Master of Business Telecommunications) degrees from Delft University of Technology, Delft, The Netherlands, in 1985, 1989, and 1996, respectively.

Since 1985 he has been with the Faculty of Electrical Engineering, Mathematics, and Computer Science of Delft University of Technology. Currently, he is an associate professor in the Department of Intelligent Systems. He is the chairman of the WIC (Werkgemeenschap voor Informatie- en Communicatietheorie in the Benelux) and the secretary of the IEEE Benelux Chapter on Information Theory. He was a Visiting Researcher at the University of California at Davis, USA, the University of Johannesburg, South Africa, the Tokyo Institute of Technology, Japan, and EPFL, Switzerland. His main research interests are in the areas of channel and network coding.