Further results on binary convolutional codes with an optimum distance profile

Johannesson, Rolf; Paaske, Erik

Published in:
IEEE Transactions on Information Theory

1978

Link to publication

Citation for published version (APA):

Total number of authors: 2

General rights
Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: https://creativecommons.org/licenses/

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
as already discussed in [7], an exhaustive search becomes practical and feasible when optimizing according to several criteria. Let us start by considering the problem of finding codes that are simultaneously optimal according to several criteria. This is of fundamental importance for the error probability and for the computational performance, it seems reasonable to search for codes which are simultaneously optimal or near-optimal according to several criteria. As has been recently observed [1], [2], [3] that to obtain a good error probability when Viterbi decoding or sequential decoding is used, and further that the number of codewords with a nonzero first branch and weight \( d_w \) should be as small as possible. Also it has been observed [1], [2], [6] that, for good computational performance with sequential decoding, the column distance "grows as rapidly as possible," which is of course a lower bound on the minimum number of weight \( d_w \) paths. It is shown how the optimum distance profile criterion can be used to limit the search for codes with a large value of \( d_w \). We present extensive lists of such robustly optimal codes containing rate \( R = \frac{1}{2} \) nonsystematic codes, several with \( d_w \) superior to that of any previously known code of the same rate and memory; rate \( R = \frac{3}{4} \) systematic codes; and rate \( R = \frac{7}{8} \) nonsystematic codes. As a counterpart to quick-look-in (QLI) codes which are not "transparent," we introduce rate \( R = \frac{1}{2} \) easy-look-in-transparent (ELIT) codes with a feedforward inverse \((1 + D D)^{-1}\). In general, ELIT codes have \( d_w \) superior to that of QLI codes.

II. NOTATION AND DEFINITIONS

For rates \( R = (N - 1)/N \), we can represent an FCE by the \((N - 1) \times N\) matrix \(G(D)\), where

\[
G(D) = \begin{pmatrix}
G_1^1(D) & G_1^2(D) & \cdots & G_1^N(D) \\
G_2^1(D) & G_2^2(D) & \cdots & G_2^N(D) \\
\vdots & \vdots & \ddots & \vdots \\
G_{N-1}^1(D) & G_{N-1}^2(D) & \cdots & G_{N-1}^N(D)
\end{pmatrix},
\]

and

\[
G_i^j(D) = g_i^1 + g_i^2 D + g_i^3 D^2 + \cdots
\]

where \( G_i^j \) is the transform of the generator sequence \( g_i^1, g_i^2, g_i^3, \ldots \) \( 1 \leq i \leq N - 1, 1 \leq j \leq N \). Each \( G_i^j(D) \) is called a generator polynomial. Corresponding to the encoder, we can also write the parity matrix in polynomial form, i.e.,

\[
H(D) = [H_1^1(D), H_2^1(D), \ldots, H_1^N(D)]
\]

where each \( H_i^j(D) \) is called a parity polynomial.

We consider only noncatastrophic codes [4], and we define the overall constraint length as

\[
\nu = \max_{0 < j < N} \{\deg [H_i^j(D)]\}.
\]

Then \( \nu \) becomes also the number of memory elements in the corresponding minimal encoder.

A key parameter used in the evaluation of the encoders is the column distance, which was originally defined by Costello [5]. The order \( j \) column distance \( d_j \) is the minimum Hamming weight of all codewords having a nonzero first branch and truncated after \((j + 1)\) branches. In particular, \( d_w \) is called the free distance of the code. Also, in a previous paper [3], Johannesson introduced the distance profile as the \((\nu + 1)\)-tuple \( d = [d_0, d_1, \ldots, d_\nu] \) and defined a distance profile \( d' \) to be superior to \( d'' \) if \( d_j > d'_j \) for the smallest index \( j, 0 \leq j \leq \nu \), where \( d_j \neq d'_j \). Hence an optimum distance profile (ODP) ensures that, for the first constraint length, the column distance "grows as rapidly as possible," which is not necessarily the case for the average column distance function [2].

III. HOW TO FIND GOOD ENCODERS

It is well known [6] that \( d_w \) is the principal determinant of decoding error probability when Viterbi decoding or sequential decoding is used, and further that the number of codewords with a nonzero first branch and weight \( d_w \) should be as small as possible. Also it has been observed [1], [2], [6] that, for good computational performance with sequential decoding, the column distance should "grows as rapidly as possible," i.e., an optimum distance profile is desirable. Since different criteria appear to be of fundamental importance for the error probability and for the computational performance, it seems reasonable to search for FCE's generating codes which are simultaneously optimal or near optimal according to the mentioned criteria.

At first glance, one might believe it more difficult to search for FCE's which are optimal according to several criteria, but interestingly enough, a remarkable simplification in the search procedure may be obtained if the criteria are carefully chosen. As already discussed in [7], an exhaustive search becomes practically impossible even for rather small constraint lengths, and therefore some methods are needed to limit the search for good encoders. One approach is to select a subset in which the possibility of finding good encoders is "expected to be good." Another approach is to use rules that reject a large fraction of codes from the complete ensemble either because they cannot be good encoders or because the distance properties of the codes generated equal the distance properties of some code in the remaining set.

We shall use the case of rate \( R = \frac{1}{2} \) nonsystematic codes to illustrate how a combination of the two approaches becomes very feasible when optimizing according to several criteria. Let us start with the first limitation approach and select the subset of ODP encoders. The justification to expect this subset to be good with regard to \( d_w \) lies mainly in Table V in [3], but also in the fact that, for most constraint lengths, ODP encoders have optimum \( d_w \) which is of course a lower bound on \( d_w \). The subset of ODP encoders is also relatively small. Provided that both generator polynomials are monic, a general nonsystematic rate \( R = \frac{1}{2} \) encoder of constraint length \( \nu \) is specified by \( 2^\nu \) coefficients implying an ensemble "size" of \( 2^\nu \), while our subset "size" is \( S(2^\nu) \), where \( S(\nu) \) is the number of systematic ODP encoders of constraint length.
v. From the results leading to [3], \( S(v) \) is known to be relatively small, and hence a substantial simplification in the search is obtained. Furthermore, the subset is easy to generate. Because of the restriction of the ODP to column distances up to \( v \), we can use well-known results of Bussgang [8] and Forney [9] to generate the subset of nonsystematic ODP encoders from the systematic ODP encoders in the following simple way. Let the systematic code with distance profile \( d = \{d_0,d_1,\ldots,d_v\} \) have parity matrix

\[
[H(D),H^2(D)\ldots H^{v-1}(D),1],
\]

and let \( * \) denote polynomial multiplication with truncation after degree \( v \). If \( P(D) \) runs through all monic polynomials of degree \( \leq v \), then

\[
H(D) = [P(D)] * [H^1(D),H^2(D)\ldots H^{v-1}(D),1]
= [P(D) * H^i(D),P(D) * H^2(D)\ldots P(D) * H^{v-1}(D),P(D)]
\]

runs through the corresponding set of nonsystematic encoders with distance profile \( d \).

With the properties of ODP encoders in mind, it now becomes evident that we obtain a great simplification in the search for good encoders by choosing the criteria in the following way:

1. optimum distance profile (ODP),
2. optimum free distance (OFD) conditioned on 1),
3. minimum number of weight \( d_s \) paths conditioned on 2).

In the subset of ODP encoders, we can then use the second approach, viz. rejection rules, to limit the search for encoders with optimum \( d_s \). The rules which we have used are similar to those mentioned in [7]. On the average, they discarded about 99 percent of the FCE's, and hence \( d_s \) need be calculated only for the remaining one percent of the FCE's in the subset of ODP encoders.

In Table I, we have listed the results of a computer search for nonsystematic rate \( R = \frac{1}{2} \) FCE's. In all tables, the generators are given in octal form, as introduced in [3], where the first digit denotes \([g_0,g_1,g_2]\), the second denotes \([g_3,g_4,g_5]\), etc. All the codes are ODP and, except for a few cases where the search would become unreasonably large, they are also optimum according to criteria 2) and 3). For comparison, we have plotted the \( d_s \) of our codes in Fig. 1 together with \( d_s \) for the ODF codes of Odenwalder [10], Larsen [11], and Paaske [12]; the complementary codes of Bahl and Jelinek [13]; and an upper bound on \( d_s \) calculated using the method given by Heller [14]. Four of the ODP codes, viz. those for \( v = 18, 19, 21, \) and 23 have \( d_s \) superior to that of any previously known rate \( R = \frac{1}{2} \) code with the same constraint length. Furthermore, the \( v = 23 \) code has been recommended to NASA for use with sequential decoding on the deep-space channel [15]. [16]. In general, a limitation of the search to the subset of ODP encoders seems to result in only a very small reduction in achievable \( d_s \), which is illustrated by the fact that encoders with a larger \( d_s \) are known only for \( v = 11, 12, 14, 15, \) and 16; these are listed in Table II for completeness. OFD encoders for \( v = 11 \) and 12 were previously found by Larsen [11], but the encoders listed here have a smaller number of weight \( d_s \) paths; the encoders for \( v = 14, 15, \) and 16 were hitherto unpublished, but were previously found by Paaske [12].

In Table III and Table IV, we have listed corresponding results for systematic and nonsystematic encoders of rate \( R = \frac{1}{2} \). Table V shows the parity polynomials of the codes in Table IV. For this rate, \( S(v) \) is much greater than the corresponding number for rate \( R = \frac{1}{2} \), implying that the subset of nonsystematic ODP encoders is also much greater. Therefore, even with the rejection rules in effect, an exhaustive search according to criterion 2) was not reasonable in several cases. In Fig. 2 we have compared \( d_s \) and \( d_s \) of our codes with \( d_s \) of the ODF codes by Paaske [7] and with an upper bound on \( d_s \) calculated using the method given by Heller [14].

![Fig. 1](image-url) Free distance \( d_s \) for some rate \( \frac{1}{2} \) convolutional codes.

---

**Table I**

<table>
<thead>
<tr>
<th>( v )</th>
<th>( d_1 )</th>
<th>( d_2 )</th>
<th>Notes</th>
<th>( d_s )</th>
<th>#paths</th>
<th>( d_s )</th>
<th>#paths</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>6</td>
<td>1,2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>5</td>
<td>1,2</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>74</td>
<td>54</td>
<td>1,2</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>69</td>
<td>56</td>
<td>1,2</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
<td>12</td>
<td>1,2</td>
<td>5</td>
<td>6</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>631</td>
<td>564</td>
<td>1,2</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>206</td>
<td>157</td>
<td>1,2</td>
<td>6</td>
<td>11</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>751</td>
<td>557</td>
<td>1,2</td>
<td>6</td>
<td>6</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>9</td>
<td>7666</td>
<td>6714</td>
<td>1,2</td>
<td>6</td>
<td>7</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>10</td>
<td>7512</td>
<td>5562</td>
<td>1,2</td>
<td>7</td>
<td>13</td>
<td>14</td>
<td>19</td>
</tr>
<tr>
<td>11</td>
<td>6643</td>
<td>5175</td>
<td>3</td>
<td>7</td>
<td>5</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>63371</td>
<td>14244</td>
<td>3</td>
<td>8</td>
<td>29</td>
<td>15</td>
<td>2</td>
</tr>
<tr>
<td>13</td>
<td>15332</td>
<td>77136</td>
<td>2</td>
<td>8</td>
<td>12</td>
<td>16</td>
<td>5</td>
</tr>
<tr>
<td>14</td>
<td>65231</td>
<td>43677</td>
<td>3</td>
<td>8</td>
<td>10</td>
<td>17</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>517604</td>
<td>664134</td>
<td>3,4</td>
<td>8</td>
<td>5</td>
<td>18</td>
<td>10</td>
</tr>
<tr>
<td>16</td>
<td>71706</td>
<td>52702</td>
<td>3</td>
<td>9</td>
<td>18</td>
<td>19</td>
<td>9</td>
</tr>
<tr>
<td>17</td>
<td>506477</td>
<td>673711</td>
<td>4</td>
<td>9</td>
<td>7</td>
<td>20</td>
<td>12</td>
</tr>
<tr>
<td>18</td>
<td>652604</td>
<td>7740714</td>
<td>5</td>
<td>9</td>
<td>7</td>
<td>21</td>
<td>13</td>
</tr>
<tr>
<td>19</td>
<td>512662</td>
<td>7315256</td>
<td>5</td>
<td>10</td>
<td>31</td>
<td>22</td>
<td>26</td>
</tr>
<tr>
<td>20</td>
<td>6587143</td>
<td>5399395</td>
<td>6</td>
<td>10</td>
<td>13</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>21</td>
<td>6750654</td>
<td>5037144</td>
<td>4,5</td>
<td>10</td>
<td>4</td>
<td>24</td>
<td>40</td>
</tr>
<tr>
<td>22</td>
<td>6713702</td>
<td>5051646</td>
<td>6</td>
<td>10</td>
<td>1</td>
<td>24</td>
<td>25</td>
</tr>
<tr>
<td>23</td>
<td>55346125</td>
<td>73544413</td>
<td>5,6</td>
<td>11</td>
<td>28</td>
<td>25</td>
<td>13</td>
</tr>
</tbody>
</table>

**Notes:**

1. This code was found by Johannesson [3] and is listed here for completeness.
2. This code is OFD.
3. An OFD code with the same memory is listed in Table II.
4. The search according to criterion 3) was not exhaustive, and hence a slightly better code might exist.
5. This code has a free distance superior to that of any previously known code with the same memory.
6. The search according to criterion 2) was not exhaustive, and hence a better code might exist.
TABLE II
SOME NONSYSTEMATIC CONVOLUTIONAL CODES WHICH ARE OFD

<table>
<thead>
<tr>
<th>v</th>
<th>G_1^v</th>
<th>G_2^v</th>
<th>d_v</th>
<th>#paths</th>
<th>d_∞</th>
<th>#paths</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>7173</td>
<td>5261</td>
<td>7</td>
<td>6</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>5373</td>
<td>7230</td>
<td>7</td>
<td>3</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>6331</td>
<td>5536</td>
<td>1</td>
<td>8</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>442754</td>
<td>627324</td>
<td>7</td>
<td>2</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>715502</td>
<td>515476</td>
<td>8</td>
<td>5</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

Note:
1. The search for the code with the smallest number of weight d_∞ paths was not exhaustive, and hence a slightly better code might exist.

Massey and Costello [1] introduced the class of quick-look-in (QLI) codes, nonsystematic codes of rate $R = \frac{2}{3}$ in which the two generators differ only in the second position. The reason to prefer a QLI code rather than another nonsystematic code is the ease of extracting the information digits from the hard-decisioned received sequences, since a feedforward (FF) inverse $(P_1(D), P_2(D)) = (1,1)$ can be realized by a simple modulo 2 adder. Furthermore, since the FF-inverse has "weight" two, the "error amplification factor" $A = 2$ is the smallest possible for nonsystematic codes. However, if differential coding is used together with PSK, it is often desirable to use a "transparent" code [17], i.e., a code which has the all-one sequence as a codeword. Since the QLI codes are not transparent and since the smallest error amplification factor for a nonsystematic transparent code is $A = 3$, then, as a counterpart to the QLI codes, we are led to introduce easy-look-in-transparent (ELIT) codes such that

### TABLE III
ODP SYSTEMATIC CONVOLUTIONAL CODES WITH RATE $\frac{2}{3}$

<table>
<thead>
<tr>
<th>v</th>
<th>G_1^v</th>
<th>G_2^v</th>
<th>d_v</th>
<th>#paths</th>
<th>d_∞</th>
<th>#paths</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>58</td>
<td>64</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>64</td>
<td>62</td>
<td>4</td>
<td>17</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>67</td>
<td>63</td>
<td>4</td>
<td>7</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>554</td>
<td>704</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>664</td>
<td>742</td>
<td>5</td>
<td>30</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>665</td>
<td>743</td>
<td>5</td>
<td>15</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5734</td>
<td>6370</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>5736</td>
<td>6326</td>
<td>6</td>
<td>54</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>5736</td>
<td>6323</td>
<td>6</td>
<td>26</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>6414</td>
<td>74334</td>
<td>6</td>
<td>17</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>57372</td>
<td>6326</td>
<td>6</td>
<td>6</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>57371</td>
<td>63225</td>
<td>7</td>
<td>72</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>664150</td>
<td>743314</td>
<td>7</td>
<td>31</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>664072</td>
<td>743346</td>
<td>7</td>
<td>21</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>573713</td>
<td>632252</td>
<td>7</td>
<td>10</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>6640344</td>
<td>7431004</td>
<td>8</td>
<td>102</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>551432</td>
<td>7023726</td>
<td>8</td>
<td>39</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>5514633</td>
<td>7023725</td>
<td>8</td>
<td>25</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>5736142</td>
<td>63235074</td>
<td>8</td>
<td>18</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>6641546</td>
<td>74311464</td>
<td>9</td>
<td>135</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>6641547</td>
<td>74311465</td>
<td>9</td>
<td>68</td>
<td>12</td>
<td></td>
</tr>
</tbody>
</table>

Our first optimality criterion selects the subset of codes with an optimum $d_\infty$, but we can of course enlarge the subset by requiring only an optimum $d_v$, where $l < \nu$. For the case of ELIT codes we have chosen $l = \nu - 2$ for two reasons.

IV. TRANSPARENT CODES WITH A SIMPLE ENCODER

Massey and Costello [1] introduced the class of quick-look-in (QLI) codes, nonsystematic codes of rate $R = \frac{2}{3}$ in which the two generators differ only in the second position. The reason to prefer a QLI code rather than another nonsystematic code is the ease of extracting the information digits from the hard-decisioned received sequences, since a feedforward (FF) inverse $(P_1(D), P_2(D)) = (1,1)$ can be realized by a simple modulo 2 adder. Furthermore, since the FF-inverse has "weight" two, the "error amplification factor" $A = 2$ is the smallest possible for nonsystematic codes. However, if differential coding is used together with PSK, it is often desirable to use a "transparent" code [17], i.e., a code which has the all-one sequence as a codeword. Since the QLI codes are not transparent and since the smallest error amplification factor for a nonsystematic transparent code is $A = 3$, then, as a counterpart to the QLI codes, we are led to introduce easy-look-in-transparent (ELIT) codes such that

1E) $G_1(D)$ and $G_2(D)$ both have odd weight, and
2E) $(1 + D)G_1(D) + DG_2(D) = 1$.

Property 2E) specifies that ELIT codes have the FF-inverse $(P_1(D), P_2(D)) = (1 + D, D)$ with error amplification factor $A = 3$.

Although the subset of ODP QLI codes [3] is small, it contains FCE's with relatively large values of $d_\infty$, and therefore one could expect the subset of ODP ELIT codes to have the same property. However, this turned out not to be the case. For several constraint lengths, the latter subset contains only bad codes, and in some cases, it is even empty. To overcome this problem, it is necessary to enlarge the subset which can be done systematically using a generalized distance profile:

$$d_l = [d_0, d_1, d_2, \ldots, d_l].$$

Our first optimality criterion selects the subset of codes with an optimum $d_\infty$, but we can of course enlarge the subset by requiring only an optimum $d_v$, where $l < \nu$. For the case of ELIT codes we have chosen $l = \nu - 2$ for two reasons.

- $d_\infty$ for nonsystematic ODP codes
- $d_\infty$ for ELIT codes with optimum $d_v$, $l < \nu$
- Upper bound on $d_\infty$
- $d_v$ for systematic ODP codes
- $d_v$ for ODP codes

Fig. 2. Free distance $d_\infty$ and column distance for some rate $\frac{2}{3}$ convolutional codes.

Fig. 3. Free distance $d_\infty$ for some rate $\frac{2}{3}$ convolutional codes with a simple encoder inverse.
TABLE IV
ODP Nonsystematic Convolutional Codes with Rate 2/3

<table>
<thead>
<tr>
<th>v</th>
<th>G1</th>
<th>G2</th>
<th>G3</th>
<th>G4</th>
<th>G5</th>
<th>Notes</th>
<th>d_v</th>
<th>#paths</th>
<th>d_m</th>
<th>#paths</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>15</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>60</td>
<td>30</td>
<td>70</td>
<td>34</td>
<td>74</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>50</td>
<td>24</td>
<td>54</td>
<td>28</td>
<td>70</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>54</td>
<td>30</td>
<td>64</td>
<td>00</td>
<td>46</td>
<td>6</td>
<td>5</td>
<td>30</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>64</td>
<td>12</td>
<td>52</td>
<td>26</td>
<td>66</td>
<td>44</td>
<td>1</td>
<td>5</td>
<td>15</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>54</td>
<td>16</td>
<td>66</td>
<td>25</td>
<td>71</td>
<td>60</td>
<td>5</td>
<td>9</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>53</td>
<td>23</td>
<td>51</td>
<td>36</td>
<td>53</td>
<td>67</td>
<td>6</td>
<td>54</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>11</td>
<td>710</td>
<td>260</td>
<td>670</td>
<td>320</td>
<td>404</td>
<td>714</td>
<td>6</td>
<td>29</td>
<td>10</td>
<td>29</td>
</tr>
<tr>
<td>12</td>
<td>740</td>
<td>260</td>
<td>520</td>
<td>367</td>
<td>414</td>
<td>515</td>
<td>2</td>
<td>6</td>
<td>27</td>
<td>10</td>
</tr>
<tr>
<td>13</td>
<td>710</td>
<td>260</td>
<td>670</td>
<td>440</td>
<td>593</td>
<td>732</td>
<td>6</td>
<td>5</td>
<td>11</td>
<td>9</td>
</tr>
<tr>
<td>14</td>
<td>676</td>
<td>846</td>
<td>206</td>
<td>256</td>
<td>470</td>
<td>142</td>
<td>7</td>
<td>65</td>
<td>12</td>
<td>58</td>
</tr>
<tr>
<td>15</td>
<td>722</td>
<td>856</td>
<td>420</td>
<td>457</td>
<td>735</td>
<td>27</td>
<td>7</td>
<td>38</td>
<td>12</td>
<td>25</td>
</tr>
<tr>
<td>16</td>
<td>1700</td>
<td>840</td>
<td>7560</td>
<td>7024</td>
<td>5164</td>
<td>4260</td>
<td>3</td>
<td>7</td>
<td>14</td>
<td>12</td>
</tr>
<tr>
<td>17</td>
<td>5330</td>
<td>3290</td>
<td>5390</td>
<td>5690</td>
<td>6530</td>
<td>5434</td>
<td>3</td>
<td>7</td>
<td>13</td>
<td>18</td>
</tr>
<tr>
<td>18</td>
<td>6734</td>
<td>1734</td>
<td>4330</td>
<td>1574</td>
<td>5140</td>
<td>7014</td>
<td>3</td>
<td>8</td>
<td>106</td>
<td>14</td>
</tr>
<tr>
<td>19</td>
<td>5044</td>
<td>3570</td>
<td>4734</td>
<td>1024</td>
<td>5712</td>
<td>5622</td>
<td>3</td>
<td>8</td>
<td>43</td>
<td>14</td>
</tr>
<tr>
<td>20</td>
<td>7030</td>
<td>3852</td>
<td>7566</td>
<td>7012</td>
<td>6756</td>
<td>5100</td>
<td>3</td>
<td>8</td>
<td>23</td>
<td>14</td>
</tr>
<tr>
<td>21</td>
<td>6562</td>
<td>2116</td>
<td>4160</td>
<td>6431</td>
<td>4434</td>
<td>7225</td>
<td>3</td>
<td>8</td>
<td>11</td>
<td>15</td>
</tr>
<tr>
<td>22</td>
<td>57700</td>
<td>12140</td>
<td>63060</td>
<td>15244</td>
<td>70444</td>
<td>47730</td>
<td>3</td>
<td>9</td>
<td>144</td>
<td>16</td>
</tr>
<tr>
<td>23</td>
<td>51630</td>
<td>25240</td>
<td>42050</td>
<td>61960</td>
<td>61234</td>
<td>44334</td>
<td>3</td>
<td>9</td>
<td>60</td>
<td>16</td>
</tr>
</tbody>
</table>

Notes:
1. This code is OFD.
2. The search according to criterion 3) was not exhaustive, and hence a slightly better code might exist.
3. The search according to criterion 2) was not exhaustive, and hence a better code might exist.

1) For most values of v, this subset contains codes with reasonably large values of d_m.
2) The subset is easy to generate. We denote codes with a FF-inverse \( (1 + D, D) \), i.e., codes satisfying condition 2E), as easy-look-in (ELI) codes, and we denote by C(V) the subset of ELI codes of constraint length v with an optimum generalized distance profile \( d_m \). Then C(v) can be easily generated from C(v - 1) since, for each generator polynomial \( G(D) \) in C(v), there are only two candidate codes for C(v - 1), namely \( G(D) = G(D) \cdot D \) and \( G(D) = G(D) \cdot D^{-1} \cdot D^r \), each of which belongs to C(v) if it also has optimum column distance \( d_m \). Finally, the codes are also transparent if they satisfy condition 1E).

TABLE V
Parity Polynomials of the Codes in Table IV

<table>
<thead>
<tr>
<th>v</th>
<th>H1</th>
<th>H2</th>
<th>H3</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>74</td>
<td>54</td>
<td>64</td>
</tr>
<tr>
<td>4</td>
<td>50</td>
<td>62</td>
<td>72</td>
</tr>
<tr>
<td>5</td>
<td>65</td>
<td>65</td>
<td>53</td>
</tr>
<tr>
<td>6</td>
<td>45</td>
<td>45</td>
<td>70</td>
</tr>
<tr>
<td>7</td>
<td>72</td>
<td>72</td>
<td>53</td>
</tr>
<tr>
<td>8</td>
<td>65</td>
<td>40</td>
<td>57</td>
</tr>
<tr>
<td>9</td>
<td>50</td>
<td>45</td>
<td>65</td>
</tr>
<tr>
<td>10</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>11</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>12</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>13</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>14</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>15</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>16</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>17</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>18</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>19</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>20</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>21</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>22</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
<tr>
<td>23</td>
<td>710</td>
<td>710</td>
<td>65</td>
</tr>
</tbody>
</table>

Notes:
1. This code is also OFD.
2. This code is also ODP.
3. The subset of ELIT codes with optimum \( d_m \) is empty for \( v = 21 \).
TABLE VII
SOME ELI CODES WITH FREE DISTANCE SUPERIOR TO THAT OF ELIT AND QLI CODES WITH THE SAME MEMORY

<table>
<thead>
<tr>
<th>v</th>
<th>c₀</th>
<th>c₁</th>
<th>Notes</th>
<th>d₉</th>
<th>d₈</th>
<th>d₇</th>
<th>d₆</th>
<th>d₅</th>
<th>d₄</th>
<th>d₃</th>
<th>d₂</th>
<th>d₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>56</td>
<td>62</td>
<td>1,2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>311</td>
<td>723</td>
<td>2,6</td>
<td>11</td>
<td>6</td>
<td>6</td>
<td>10</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6466</td>
<td>63521</td>
<td>2</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>1</td>
<td>15</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>611012</td>
<td>233036</td>
<td>8</td>
<td>1</td>
<td>8</td>
<td>17</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>646501</td>
<td>622003</td>
<td>8</td>
<td>2</td>
<td>17</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>646604</td>
<td>635214</td>
<td>2</td>
<td>10</td>
<td>18</td>
<td>10</td>
<td>8</td>
<td>20</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>646601</td>
<td>635203</td>
<td>8</td>
<td>2</td>
<td>17</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes:
1. This code is also OFD.
2. This code is also ODP.

V. CONCLUSION

The ODP criterion seems important for two reasons. One is the improvement observed in the computational performance for sequential decoding of ODP codes. The other is that, in a search for "good" encoders, the ODP criterion can be used to limit the ensemble size without serious degradation in the attainable values of $d_a$.

REFERENCES


Addition to "A Method for Decoding of Generalized Goppa Codes"

DAVID M. MANDELBAM

Abstract—In a previous correspondence, a decoding procedure which uses continued fractions and which is applicable to a wide class of algebraic codes including Goppa codes was presented. The efficiency of this method is significantly increased.

The efficiency of the decoding method in the above correspondence can be significantly increased. In the following, all references will be to the equations and bibliography of the original paper.

In the decoding process, the test (18) with the associated multiplications is not required; instead, the test for the correct convergent can be taken as

$$\text{deg}(r_{i+1}(x)) < k + \text{deg}(q_i(x)),$$

(18')

and therefore no multiplication is needed.

This is shown as follows. From [6], we have

$$q_{i-1}(x)r_{i+1}(x) + q_i(x)r_i(x) = m(x)$$

and

$$p_{i-1}(x)r_{i+1}(x) + p_i(x)r_i(x) = u(x).$$

Multiplying the above two equations by $p_i(x)$ and $q_i(x)$, respectively, and subtracting the first from the second yields

$$q_i(x)p_{i-1}(x) - p_i(x)m(x) = q_i(x)p_{i-1}(x) + q_i(x)p_i(x)r_i(x) - p_i(x)q_i(x)r_i(x) - p_i(x)q_{i-1}(x)r_{i+1}(x).$$

Since for the correct convergent

$$u(x) = u(x) - m(x)p_i(x)/q_i(x)$$

the rule (18') follows. Also, $u(x)$ can be obtained by means of $u(x) = (-1)r_{i+1}(x)/q_i(x)$ when (18') is satisfied. This is a particularly attractive method of obtaining $u(x)$ for low-rate codes.

As a result of the new test (18'), it is seen that this procedure is of equivalent complexity to that given in [4].

It can also be easily seen that the method presented in the above correspondence can be used for decoding Goppa and Bose-Chandhuri-Hocquenghem (BCH) codes using the standard syndrome $S(x)$ as developed in [9]. That is, if $S(x) = \eta(x)/\alpha(x)$ and $g(x)$, then, if we set $S'(x) = \alpha(x)/\eta(x)$ where $r = \text{deg}(S(x))$, we can use the syndrome $S'(x)$ in the decoding method in the above correspondence. It will be noticed that this method, like the Berlekamp-Massey algorithm, starts with the "end" of the syndrome while the method of [4] starts in the "middle" of the syndrome.

Two corrections should be made in the example in the above correspondence; namely, the remainders $r_2(x)$ and $r_6(x)$ should read $r_2(x) = \beta^8x^6 + \beta^3x^4 + x^3 + x^2 + \beta^3x^2 + \beta^3x + \beta$ and $r_6(x) = \beta^8x^4 + \beta^3x^2 + \beta^3x + \beta^3 + 1$, respectively.

Manuscript received February 28, 1977; revised June 1, 1977.

The author is at P.O. Box 845, Eatontown, NJ 07724.