Binary Multilevel Convolutional Codes with Unequal Error Protection Capabilities

Robert H. Morelos-Zaragoza  
*LSI Logic Corporation*, robert.morelos-zaragoza@sjsu.edu

Hideki Imai  
*University of Tokyo*

Follow this and additional works at: https://scholarworks.sjsu.edu/ee_pub

Part of the Electrical and Computer Engineering Commons

**Recommended Citation**

Binary Multilevel Convolutional Codes with Unequal Error Protection Capabilities

Robert H. Morelos-Zaragoza, Senior Member, IEEE, and Hideki Imai, Fellow, IEEE

Abstract—Binary multilevel convolutional codes (CC’s) with unequal error protection (UEP) capabilities are studied. These codes belong to the class of generalized concatenated (GC) codes [1]. Binary CC’s are used as outer codes. Binary linear block codes of short length, and selected subcodes in their two-way subcode partition chain, are used as inner codes. Multistage decodings are presented that use Viterbi decoders operating on trellises with similar structure to that of the constituent binary CC’s. Simulation results of example binary two-level CC’s are also reported.

Index Terms—Multilevel codes, multistage decoding, punctured convolutional codes, unequal error protection.

I. INTRODUCTION

GENERALIZED concatenated (GC) codes [1] are a powerful family of error correcting codes based on multiple outer codes, and an inner code and its partition into subcodes. Due to their multilevel structure [2], GC codes can be designed as unequal error protection (UEP) codes. The need for UEP arises in communications systems where part of the source messages are more important, or error sensitive, than others. Specific examples include practically all digital speech and image transmission systems.

Convolutional UEP codes based on the GC code construction are generalizations of the two-level convolutional codes (CC’s) introduced independently by Pellizzoni and Spalvieri [3] and by Hattori and Saitoh [4], and then studied in [5] and [6]. Moreover, multistage decoding procedures can be devised for convolutional UEP codes as a generalization of the two-stage decoding of [2][1][1] CC’s [3]–[6].

In [3] and [4] multilevel coset (or “superimposed”) codes with outer CC’s are presented. In [3] two-level CC’s were constructed and shown to offer improved error performance, with reduced decoding complexity, compared to previously known codes with the same number of states. A construction using punctured convolutional (PC) codes as component codes in the [2][2][1] construction was proposed in [4].

In this letter it is shown that two-level CC’s based on the [2][2][1] construction are special cases of GC codes. This gives binary multilevel CC’s a rich algebraic structure that is useful in explaining their UEP capabilities and in devising low-complexity multistage decoding algorithms.

II. BINARY MULTILEVEL CONVOLUTIONAL CODES

As usual, let \((n, k, d)\) denote a binary linear block code of length \(n\), dimension \(k\), and minimum distance \(d\). The construction of a binary multilevel CC \(C\) starts with a binary \((n_1, k_1, d_1)\) code \(C_1\) and its two-way partition as a chain of \(M\) \((n_i, k_i, d_i)\) subcodes \(C_i, i = 2, 3, \ldots, M + 1\), such that \(C_1 \supseteq C_2 \supseteq \cdots \supseteq C_{M+1}\), where, for convenience, we define \(C_{M+1} := \{0\}\) and \(d_{M+1} := \infty\). Let \(C_H = [C_i/C_{i+1}]\) denote an \((n_H, k_H, \delta_i)\) binary subcode of \(C_i\), which is a set of coset representatives of \(C_{i+1}\) in \(C_i\), of dimension \(k_H = k_i - k_{i+1} = 1\) and minimum Hamming distance \(\delta_i \geq d_i, 1 \leq i \leq M\). Then \(C_1\) is the direct sum \(C_1 = C_1 + C_2 + \cdots + C_M\). For \(1 \leq i \leq M\), let \(C_{O_i}\) denote a rate-\(k_{O_i}\) memory-\(m_i\) binary CC with minimum distance \(d_{O_i}\). Then a binary multilevel CC \(C\) is defined as the direct sum

\[
C \equiv C_{O_1} \ast C_{H_1} + C_{O_2} \ast C_{H_2} + \cdots + C_{O_M} \ast C_{H_M} \tag{1}
\]

where \(C_{O_i} \ast C_{H_i}\) denotes a product (or concatenated) code with \(C_{O_i}\) as outer code and \(C_{H_i}\) as inner code, \(1 \leq i \leq M\). \(C\) is a binary \(M\)-level CC of rate \(R = k/n\) memory \(m_1 + m_2 + \cdots + m_M\), and minimum distance \(d\), where \(k\) is given by \(k = \sum_{i=1}^{M} k_{O_i}\) and \(d \geq \min\{\delta_1 d_{O_1}, \ldots, \delta_M d_{O_M}\}\).

A. UEP Capabilities

Because of their structure, multilevel CC’s can be designed as UEP CC’s. The UEP capability of a linear CC follows from a necessary and sufficient condition for a linear code to be a UEP code [7]: the set of minimum-weight codewords does not span a linear UEP code. From this condition and the definitions above, it follows that if

\[
\delta_1 d_{O_1} \geq \delta_2 d_{O_2} \geq \cdots \geq \delta_M d_{O_M} \tag{2}
\]

then those coded sequences in correspondence with \(k_{O_i}\) information bits have a Hamming distance between them of at least \(\delta_i d_{O_i}, 1 \leq i \leq M\). In the language of UEP codes we say that a binary multilevel CC is an \(M\)-level UEP code with separation vector

\[
\mathbf{\delta} = (\delta_1 d_{O_1}, \delta_2 d_{O_2}, \ldots, \delta_M d_{O_M})
\]

and message space

\[
\{0,1\}^{k_{O_1}} \times \{0,1\}^{k_{O_2}} \times \cdots \times \{0,1\}^{k_{O_M}}.
\]

For simplicity, in this letter only two-level UEP CC’s are dealt with. It is assumed that the \(M\)-component concatenated...
codes are chosen in such a way that the separation vector is \( \mathbf{s} = (s_1, s_2) \), where \( s_1 = \delta_1d_{O1} = \delta_2d_{O2} = \cdots = \delta_Ld_{O1}, s_2 = \delta_{L+4}d_{O(L+4)} = \delta_{L+2}d_{O(L+2)} = \cdots = \delta_Md_{OM} \), with \( s_1 > s_2 \) and \( 1 \leq L < M \).

### III. Two Constructions of Binary Multilevel Convolutional Codes

In this section two specific constructions of binary UEP CC’s, based on generalized concatenation, are presented. For simplicity, we only consider the cases where the inner block code is of length \( \ell_I = 2 \) or \( \ell_I = 3 \), and two or three binary CC’s are used as outer codes, although other choices of \( \ell_I \) and number of component codes are possible.

#### A. Construction I

For \( i = 1, 2 \), let \( C_{O1} \) be a rate-\( k_{O1}/n_{O1} \), memory-\( m_{i} \), and minimum distance \( d_{O1} \) CC. Let \( C_1 \) be a binary block (2, 2, 1) code with generator matrix

\[
G_1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}
\]

so that \( C_{11} = \{00, 01\} \) and \( C_{12} = \{00, 11\} \). Then it follows from the results of Section II that the code

\[
C = C_{O1} \ast C_{11} + C_{O2} \ast C_{12}
\]

is a two-level CC of rate \((k_{O1} + k_{O2})/2n_{O1}\), memory \((m_1 + m_2)\), and minimum distance

\[
d \geq \min\{d_{O1}, 2d_{O2}\}.
\]

Suppose that \( d_{O1} > 2d_{O2} \). Then it follows from (2) that \( C \) is a two-level UEP CC with separation vector \( \mathbf{s} = (d_{O1}, 2d_{O2}) \) and message space \( \{0, 1\}^{k_{O1}} \times \{0, 1\}^{k_{O2}} \).

Note that two-level CC’s constructed in this fashion are, in fact, equivalent to \( \eta[\Pi_1 + \Pi_2 + \Pi_3] \)-constructed codes.

**Example 1:** Let \( C_{O1} \) be a CC of rate 1/4, memory 2, and minimum distance 10 (from [9, Table A.5]). Let \( C_{O2} \) be a PC code of rate 3/4, memory 3, and minimum distance 4 (from [9, Table A.8]). Construction I results in a two-level CC \( C \) of rate 4/8, memory 5, and minimum distance 8. From condition (2), it follows that \( C_{CC} \) is a two-level UEP CC with separation vector \( \mathbf{s} = (10, 8) \) and message space \( \{0, 1\}^{k_{O1}} \times \{0, 1\}^{k_{O2}} \). Also, note that the minimum distance of the best rate-1/2 memory-3 CC is equal to 8.

**A Two-Stage Decoding Procedure:** For practical applications, in order to reduce the complexity of Viterbi decoding (at the expense of an increased error coefficient or degradation in coding gain [3], [6]), suboptimal two-stage decoding (TSD) may be adopted. Coded bits are Binary phase-shift keying (BPSK)-modulated and transmitted over an additive white Gaussian noise (AWGN) channel. The following TSD procedure is considered.

**Stage 1:** A decoder for a CC \( C_{1} \supset C_1 \) is used in this stage, where

\[
C_{1} = C_1 \oplus \{0, 1\}^{n_O} \ast \{00, 11\} = C_1 \oplus \{00, 11\}^{n_O}.
\]

The trellis structure \( T_{1} \) of code \( C_{1} \) is isomorphic (up to connections between states) to that of the trellis \( T_{1} \) of the first-level code \( C_1 = C_{O1} \ast C_{11} \). The only difference between \( T_{1} \) and \( T_{1} \) is that \( T_{1} \) contains two parallel subbranches per original two-bit symbol in a branch of \( T_{1} \). That is, \( \{00, 11\} \) (resp. \( \{01, 10\} \)) for 00 (resp. 01). Once an information sequence \( \sigma_{1} \) is decoded in this first stage, the \( k_{O1} \) most important bits (MIB’s) are recovered. Sequence \( \sigma_{1} \) is then reencoded by \( C_1 \) to obtain a coded sequence \( \sigma_{1} \). From this sequence, a modified received sequence \( \overline{\sigma}_{1} \) is obtained, where \( \overline{\sigma}_{1} = (\overline{\sigma}_{2}, \overline{\sigma}_{1}) \), and passed on to the second stage.

**Stage 2:** Using a trellis \( T_{2} \) for the second component code \( C_{2} \), the modified received sequence \( \overline{\sigma}_{1} \) is decoded and the \( k_{O2} \) less important bits (LIB’s) extracted.

**B. Construction II**

In this section a new construction method that combines three CC’s is presented. For \( i = 1, 2, 3 \), let \( C_{O1} \) be a rate-\( k_{O1}/n_{O1} \), memory-\( m_i \), and minimum distance \( d_{O1} \) CC. Let \( C_1 \) be a binary block (3, 3, 1) code with generator matrix

\[
G_1 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}
\]

and subcodes \( C_{T1} = \{000, 001\}, C_{T2} = \{000, 010\}, \) and \( C_{T3} = \{000, 011\} \). Then it follows from the results of Section II that the code

\[
C = C_{O1} \ast C_{T1} + C_{O2} \ast C_{T2} + C_{O3} \ast C_{T3}
\]

is a CC of rate \((k_{O1} + k_{O2} + k_{O3})/3n_{O} \), memory \((m_1 + m_2 + m_3)\), and minimum distance

\[
d \geq \min\{d_{O1}, d_{O2}, 3d_{O3}\}.
\]

This construction is a permuted version of a “\( \eta[\Pi_1 + \Pi_2 + \Pi_3] \)” construction.

Suppose that \( d_{O1} = d_{O2} \) and \( d_{O2} > 3d_{O3} \). Then, from condition (2), \( C \) is a two-level UEP CC with separation vector \( \mathbf{s} = (d_{O1}, 3d_{O3}) \) and message space \( \{0, 1\}^{k_{O1} + k_{O2}} \times \{0, 1\}^{k_{O3}} \).

**A Three-Stage Decoding Procedure:** A decoding procedure for codes obtained from Construction II is similar to the two-stage decoder in the previous section—decoding proceeds in three stages, passing decoded information from one stage to the next, with Viterbi decoders for codes \( C_{1} \), \( C_{2} \), and \( C_{3} \), where \( C_{1} = C_{1} \oplus \{0, 1\}^{n_{O}} \ast \{00, 010, 011, 111\} = C_{1} \oplus \{000, 010, 101, 111\}^{n_{O}} \) with \( C_{1} \supset C_{1} \), and \( C_{2} = C_{2} \oplus \{0, 1\}^{n_{O}} \ast \{000, 011\} = C_{2} \oplus \{000, 111\}^{n_{O}} \) with \( C_{2} \supset C_{2} \).

### IV. Tables of Codes

By means of a computer search, good binary two-level CC’s were found based on the construction methods presented in the previous section. The goal of the search was to find codes with minimum multistage decoding complexity (number
TABLE I
PARAMETERS OF THE COMPONENT CODES OF BEST TWO-LEVEL CC’S OBTAINED IN THE COMPUTER SEARCH. EACH ROW LISTS THE PARAMETERS OF A MEMORY-$m$, RATE-$k/n$, AND FREE DISTANCE $d_f$ CC. LABELS G1, j REFER TO PC CODES. $C_1$ AND $D_1$ REFER TO RATE-1/3 AND RATE-1/4 CODES, RESPECTIVELY.

<table>
<thead>
<tr>
<th>Label</th>
<th>$m$</th>
<th>$k$</th>
<th>$n$</th>
<th>$d_f$</th>
<th>$c_f$</th>
<th>$G$</th>
<th>$P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1.1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>(5,7)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>G1.2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>(15,17)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>G2.1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>(5,7)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>G2.2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>124</td>
<td>(15,17)</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Code</th>
<th>$C_{O1}$</th>
<th>$C_{O2}$</th>
<th>$m$</th>
<th>$k$</th>
<th>$n$</th>
<th>$d_{c1}$</th>
<th>$a_{c1}$</th>
<th>$b_{c1}$</th>
<th>$c_{c1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2</td>
<td>C3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>10</td>
<td>6</td>
<td>(13,15,17)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td></td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>12</td>
<td>2</td>
<td>(5,7,7,7)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td></td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>15</td>
<td>11</td>
<td>(133,145,175)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td></td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>2</td>
<td>(5,7,7,7)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td></td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>13</td>
<td>4</td>
<td>(135,15,17)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D3</td>
<td></td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>16</td>
<td>8</td>
<td>(45,27,33,37)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D4</td>
<td></td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>18</td>
<td>6</td>
<td>(53,67,71,75)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D5</td>
<td></td>
<td>6</td>
<td>1</td>
<td>4</td>
<td>20</td>
<td>37</td>
<td>(135,135,147,163)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D6</td>
<td></td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>22</td>
<td>2</td>
<td>(255,275,313,337)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D7</td>
<td></td>
<td>8</td>
<td>1</td>
<td>4</td>
<td>24</td>
<td>4</td>
<td>(463,535,733,745)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D8</td>
<td></td>
<td>9</td>
<td>1</td>
<td>4</td>
<td>27</td>
<td>10</td>
<td>(1117,1365,1633,1653)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

V. COMPUTER SIMULATION RESULTS

To illustrate the error performance of binary multilevel CC’s, two codes from Construction I were simulated. In Figs. 1 and 2, MIB-MLD and LIB-MLD are used to denote the bit-error rates of the MIB’s and LIB’s, with maximum-likelihood decoding (MLD), respectively. Similarly, MIB-TSD and LIB-TSD refer to the bit-error rates with TSD.

Fig. 1. Error performance of code S1: MLD and TSD of the MIB’s and LIB’s.

Fig. 2. Error performance of code S3: MLD and TSD of the MIB’s and LIB’s.
replacing the rate-1/4 component code D1 of Table I by a more powerful rate-1/4 code D3. It can be seen from Figs. 1 and 2 that the performance of the MIB with MLD improves drastically when increasing the number of states from code S1 to code S3 (component code D3 needs 16 states as opposed to four states of D1). The improvement in performance for the MIB is gained by, in addition to increasing the number of states, reducing the coding gain for the LIB. A discussion of the effect of error coefficients in two-stage decoding can be found in [6] and is not addressed here.

REFERENCES


