An Iterative Soft-Decision Decoding Algorithm

Takuya Koumoto† Toyoo Takata† Tadao Kasami† Shu Lin†
†Graduate School of Information Science,
  Nara Institute of Science and Technology
‡Department of Electrical Engineering, University of Hawaii at Manoa

Abstract

This paper presents a new minimum-weight trellis-based soft-decision iterative decoding algorithm for binary linear block codes. Simulation results for the RM(64,22), EBCH(64,24), RM(64,42) and EBCH(64,45) codes show that the proposed decoding algorithm achieves practically (or near) optimal error performance with significant reduction in decoding computational complexity. The average number of search iterations is also small even for low signal-to-noise ratio.

1 Introduction

Recently Moorthy et. al.[2] have proposed a zero-and-minimum-weight subtrellis-based iterative decoding scheme for binary linear block codes to achieve a very good trade-off between error performance and decoding complexity. In the scheme, all the candidate codewords are generated by an algebraic decoder based on a set of test error patterns proposed by Chase[1]. The zero-and-minimum-weight trellis search around the current best candidate codeword \(c\) is performed at most once, only if (i) a sufficient condition that the optimal solution is within the minimum distance from \(c\) holds or (ii) all the test error patterns have been exhausted and no candidate codeword satisfies the sufficient condition for optimality.

For the proposed decoding algorithm in this paper, preliminarily presented in [6], the initial candidate codeword is generated by a simple decoder, the zero-th order or the first order decoding proposed in [4]. The subsequent candidate codewords (if needed) are generated by a chain of minimum-weight trellis searches. This minimum-weight trellis around a candidate codeword \(c\) consists of only the codewords in code \(C\) that are at the minimum distance from \(c\), but does not include \(c\). The decoding iteration stops whenever a candidate codeword is found to satisfy a sufficient condition for optimality or the latest minimum-weight trellis search results in a repetition of a previously generated candidate codeword. Let this decoding algorithm be denoted Algorithm I-\(w_1\). The decoding process terminates faster than the Moorthy et. al. algorithm. Furthermore, the use of minimum-weight trellis search considerably reduces the possibility of being trapped into a local optimum. As a result, it achieves better error performance than the Moothy et. al. algorithm.

A necessary condition for Algorithm I-\(w_1\) to achieve good error performance is that the minimum weight codewords span the entire code. Reed-Muller(RM) codes satisfy this condition. Simulation results for the RM(64,22)(the (64,22) RM code), RM(64,42) and the EBCH(64,45)(the extended (64,45) BCH code) codes show that the proposed decoding algorithm practically achieves optimum MLD performance even in the range of relatively low SNR. The EBCH(64,24) code is an example for which the above necessary condition does not hold. For this code, the first and second minimum weight codewords span the entire code. In this case iterative decoding algorithm based on the first and second minimum-weight trellis search, denoted Algorithm I-\(w_1-w_2\), is used.

We also propose another approach to overcome the problem. Let \(C_0\) be a linear subcode of \(C\) and assume that the minimum weight codewords of \(C_0\) span \(C_0\). The decoding scheme is a combination of: (1) the iterative search using the minimum-weight trellis for \(C_0\) around the latest candidate codeword, and (2) a procedure for moving from the coset of \(C_0\) in \(C\), containing the current candidate codeword, to another coset which is likely to contain the optimal solution. Simulation results for the EBCH(64,24) code show that this scheme achieves better error performance than Algorithm I-\(w_1-w_2\).

2 Sufficient Conditions for Optimality

Suppose a binary \((N,K)\) linear block code \(C\) is used for error control over the AWGN channel using BPSK signaling. Let \(z = (z_1, z_2, \ldots, z_N)\) be the binary hard-decision sequence obtained from the received sequence \(r = (r_1, r_2, \ldots, r_N)\).

Let \(V_N\) denote the vector space of all binary
be defined as the minimum of 

$$L(u) = \sum_{\{i: u_i \neq z_i \text{ and } 1 \leq i \leq N\}} |r_i|.$$  

(1)

$L(u)$ is called the correlation discrepancy of $u$ with respect to $z$, and the smaller $L(u)$ is, the larger the correlation between $u$ and $r$ is. For $u$ and $v \in V_N$, $u$ is said to be better than $v$ if $L(u) \leq L(v)$. For a nonempty subset $X$ of $V_N$ and a positive integer $h$, let $h^*$ denote $\min\{|X|, h\}$ and let $best_hX$ denote the set of the $h^*$ best $n$-tuples in $X$, that is, for any $u \in best_hX$ and $v \in X - best_hX$, $L(u) \leq L(v)$. The best $n$-tuple in $X$ will be denoted $best X$.

Let $d_H(u, v)$ denote the Hamming distance between two $N$-tuples, $u$ and $v$. For $u_1, u_2, \ldots, u_h \in V_N$ and positive integers $d_1, d_2, \ldots, d_h$, let $V_N(u_1, d_1; u_2, d_2; \ldots; u_h, d_h)$ be defined as the set: \{ $u \in V_N : d_H(u, u_i) \geq d_i$ for $1 \leq i \leq h$\}, and let $L(u_1, d_1; u_2, d_2; \ldots; u_h, d_h)$ be defined as the minimum of $L(u)$ over $u \in V_N(u_1, d_1; u_2, d_2; \ldots; u_h, d_h)$.

Then we have the following early termination condition of an iterative decoding algorithm without degrading the error performance.

**Lemma:** At a stage of an iterative decoding algorithm for a block code $B(C$ itself or a coset of a linear subcode of $C$), let $GC$ denote the set of those candidate codewords which have been generated already, and let $u_{best}$ denote the best of $GC$. Suppose that for $u_1, u_2, \ldots, u_h \in GC$ and positive integers $d_1, d_2, \ldots, d_h$, $u_{best}$ is the best of $\cup_{i=1}^h \{u \in B : d_H(u, u_i) < d_i\}$. If $u_{best}$ satisfies the following condition,

$$L(u_{best}) \leq L(u_1, d_1; u_2, d_2; \ldots; u_h, d_h),$$  

(2)

then $u_{best}$ is the optimal MLD solution in $B$. \(\Delta\Delta\)

Expressions for evaluating the right-hand side of (2) for $h = 1, 2$ and $3$ have been derived [3]. These bounds are used in our proposed iterative decoding algorithm for early termination conditions.

### 3 Decoding Procedure

Let $C$ be a binary linear $(N, K)$ code with weight profile $W = \{0, w_1, w_2, \ldots\}$, and let $C_0$ be a binary linear $(N, K_0)$ subcode of $C$ with weight profile $W_0 = \{0, w_01, w_02, \ldots\}$, where $w_1$ is the minimum weight of $C$, $w_01$ is that of $C_0$ and $w_1 \leq w_01$. As a special case, $C_0$ may be $C$ itself.

#### 3.1 Minimum-weight Subtrellis Search for a Coset

For $B \in C/C_0$ and $u \in B$, minimum-weight search $(u, B)$ denotes a search procedure for finding a codeword in $B$, denoted $\varphi_B(u)$, which has the least correlation discrepancy with respect to $z$ among all codewords in $B$ at the minimum Hamming distance $w_01$ from $u$. That is,

$$\varphi_B(u) = \{v \in B : d_H(v, u) = w_01\}. \quad (3)$$

If $C_0$ is spanned by the set of minimum weight codewords of $C$, that is, $C_0 = C$ or $C$ is an UEP (unequal error protection) code, then $w_01 = w_1$ and

$$\varphi_B(u) = \varphi_C(u) \equiv \{v \in C : d_H(v, u) = w_1\}. \quad (4)$$

This search procedure is implemented by using the minimum-weight subtrellis of $C_0$ around $u$. This minimum-weight subtrellis is sparsely connected and much simpler than the full trellis of the code[2, 6].

Iterative minimum-weight search $(u, B)$ is to generate a sequence of candidate codewords, $\varphi_B(u), \varphi_B(\varphi_B(u)), \ldots$, until a certain termination condition holds. It is shown in [6] that

$$L(\varphi_B^{i+2}(u)) \leq L(\varphi_B^i(u)), \text{ for } i \geq 0, \quad (5)$$

where $\varphi_B^0(u) \equiv u$ and $\varphi_B^{i+1}(u) \equiv \varphi_B(\varphi_B^i(u))$ for $i \geq 0$, and that if $L(\varphi_B^{i+2}(u)) < L(\varphi_B^i(u))$ for $0 \leq i \leq I$, then $\varphi_B^i(u)$ with $1 \leq i \leq I$ are all different. If $j$ is the smallest index such that

$$L(\varphi_B^j(u)) = L(\varphi_B^{j-2}(u)), \quad (6)$$

then

$$\min\{L(\varphi_B^j(u)) : 0 \leq i \leq j\} = \min\{L(\varphi_B^{j-2}(u)), L(\varphi_B^{j-1}(u))\}. \quad (7)$$

The condition (6), denoted Cond$_R$, is used as one of termination conditions for iterative minimum-weight search $(u, B)$ to avoid repetition.

#### 3.2 Decoding Procedure for $C$

Suppose the set of codewords of weight $w_01$ in $C_0$ spans $C_0$ and $K - K_0$ is not large. We propose a new decoding procedure for $C$ which consists of iterated minimum-weight searches in a coset and coset shiftings. Two early termination conditions, Cond$_C$ for the entire procedure and Cond$_B$ in the subprocedure for a coset $B \in C/C_0$, are used besides the termination condition Cond$_R$ in the subprocedure for a coset.

Cond$_C$ is a sufficient condition [3] that the best candidate codeword, denoted $u_{best}$, in the set of those candidate codewords which have been generated already, denoted $GC$, is optimum based on $best_hGC$ and the weight profile $W$ of $C$, where $h$
is a specified small integer. From the Lemma, CondC is defined as

\[ L(u_{\text{best}}) \leq L(u_1, d_1; u_2, d_2; \ldots; u_l, d_l), \]

where \( l = \min\{h, |GC|\} \) and for \( 1 \leq i \leq l, u_i \in \text{best}_iGC \) and if \( \varphi_C(u_i) \in GC \), then \( d_i = w_2 \) and otherwise, \( d_i = w_1 \).

CondB is a sufficient condition that there remain no codewords in \( B \) better than \( u_{\text{best}} \). This condition is also based on \( \text{best}_iGC \) where \( GC_B = GC \cap B \) and the weight profile \( W_0 \) of \( C \) which is the same as the distance profile of any coset of \( C/C_0 \). From the Lemma, CondB is defined as

\[ L(u_{\text{best}}) \leq L(u_1, d_1; u_2, d_2; \ldots; u_l, d_l), \]

where \( l_B = \min\{h, |GC_B|\} \) and for \( 1 \leq i \leq l_B, u_i \in \text{best}_iGC_B \) and if \( \varphi_B(u_i) \in GC_B \), then \( d_i = w_2 \) and otherwise, \( d_i = w_1 \).

3.2.1 Decoding Algorithm II

We assume that \( z \notin C \).

(D1) (i) Generate \( f(r, B) \) for all \( B \in C/C_0 \), and number the \( 2^{K-K_0} \) cosets of \( C/C_0 \) in the increasing order of correlation discrepancy, \( L(f(r, B)) \).

(ii) Initialize \( GC \leftarrow \{f(r, B) : B \in C/C_0\} \), \( u_{\text{best}} \leftarrow \text{best}GC \) and \( GC_{B,h} \leftarrow \text{best}GC_B \). If CondC holds, then output \( u_{\text{best}} \) and stop. Otherwise, initialize \( GC_B, GC_{B,h} \leftarrow \{f(r, B)\} \) for \( B \in C/C_0 \), and perform Search-in (the first coset).

(D2) Search-in (B): Execute Iterative minimum-weight search \((f(r, B), B)\) together with updating the global variables each time a new candidate codeword is generated until either CondR, CondC or CondB holds. If CondC holds, then output \( u_{\text{best}} \) and stop. Suppose that CondR or CondB holds. If all cosets have been exhausted, then output \( u_{\text{best}} \) and stop; otherwise, call Search-in (the coset next to B).

(D3) If either CondC or CondB holds for every coset in \( C/C_0 \), the output is optimum.

For the special case where \( C = C_0 \), this decoding algorithm becomes Algorithm I-w1.

3.3 Choice of the Initial Candidate Codeword \( f(r, B) \) for \( B \in C/C_0 \)

For a given received sequence \( r = (r_1, r_2, \ldots, r_N) \), let \( M_K \) be the location set of the most reliable basis of the column space of a generator matrix of \( C \), and let \( \Lambda_{N-K_0} \) be the location set of the least reliable basis for the column space of a parity-check matrix of \( C_0 \). Then it follows from Theorem 1 in [5] that

\[ |M_K \cap \Lambda_{N-K_0}| = K - K_0. \]

For \( x = (x_1, x_2, \ldots, x_N) \in V_N \) and a coset \( B \in C/C_0 \), a codeword \( u = (u_1, u_2, \ldots, u_N) \in B \) satisfying the following condition is uniquely determined:

\[ u_i = x_i \text{ for all } i \in M_K - \Lambda_{N-K_0}. \]

Let \( g(x, B) \) denote the above codeword \( u \) in \( B \).

Then \( g(x, B) \) can be chosen as the initial candidate codeword \( f(r, B) \) for \( B \in C/C_0 \), where \( z \) is the hard-decision binary vector obtained from the received sequence \( r \). This \( g(x, B) \) for \( B = C/C_0 \) is a simple generalization of the zero-th order decoding proposed by Fossorier and Lin[4] to a coset of \( C_0 \) in \( C \). Similarly, a generalization of their first order decoding can be used.

4 Examples

Example 1: Let \( C = \text{EBCH}(64,24) \) and \( C_0 = \text{RM}(64,22) \). Then \( w_1 = 16, w_2 = 18, w_{01} = 16 \) and \( w_{02} = 24 \). Decoding Algorithm II, where early termination condition CondB is not used, has been simulated for this code. The simulation results are shown in Figures 1 and 2. For comparison, the simulation results of Algorithm I-w1-w2 for this code are also shown. Figure 1 shows the bit error probabilities. We see that Algorithm II practically achieves optimal error performance. Figure 2 shows the average numbers of operations (addition and comparison) of Algorithm I-w1-w2 and Algorithm II. The numbers depend on the complexities of subtrellises used in the simulation. The minimum weight subtrellis of the \( \text{RM}(64,22) \) code used in Algorithm II and the first and second weight subtrellis of the \( \text{EBCH}(64,24) \) code used in Algorithm I-w1-w2 are obtained simply by purging the 4-section full trellis diagrams of the codes. The construction of better subtrellis is under study.

We see that the average number of operations of Algorithm II can be reduced by using CondB.

Example 2: Let \( C = C_0 = \text{EBCH}(64,45) \) with \( w_1 = 8 \) and \( w_2 = 10 \). For this case, the simulation results of Algorithm I-w1, where the initial candidate codeword is provided by the first order decoding in [5], are shown in Figures 3 and 4.

Simulation results of the \( \text{RM}(64,22) \) and \( \text{RM}(64,42) \) codes are shown in [6].
Figure 1: Bit error probabilities for the EBCH(64,24).

Figure 2: Average numbers of operations for EBCH(64,24).

Figure 3: Bit error probabilities for the EBCH(64,45).

Figure 4: Average numbers of operations for EBCH(64,45).

References


