An Iterative Soft-Decicion 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 $\boldsymbol{c}$ is performed at most once, only if (i) a sufficient condition that the optimal solution is within the minimum distance from $\boldsymbol{c}$ 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 $\boldsymbol{c}$ consists of only the codewords in code C that are at the minimum distance from $\boldsymbol{c}$, but does not include $\boldsymbol{c}$. 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 $\boldsymbol{z} = (z_1, z_2, \ldots, z_N)$ be the binary hard-decision sequence obtained from the received sequence $\boldsymbol{r} = (r_1, r_2, \ldots, r_N)$.

Let V_N denote the vector space of all binary

N-tuples. For an N-tuple $\boldsymbol{u} = (u_1, u_2 \ldots, u_N) \in V_N$, let $L(\boldsymbol{u})$ be defined as follows:

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

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

Let $d_H(\boldsymbol{u}, \boldsymbol{v})$ denote the Hamming distance between two N-tuples, $\boldsymbol{u}$ and $\boldsymbol{v}$. For $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, ..., $\boldsymbol{u}_h \in V_N$ and positive integers d_1, d_2, ..., d_h, let $V_N(\boldsymbol{u}_1, d_1; \boldsymbol{u}_2, d_2; \ldots; \boldsymbol{u}_h, d_h)$ be defined as the set: $\{\boldsymbol{u} \in V_N : d_H(\boldsymbol{u}, \boldsymbol{u}_i) \geq d_i \text{ for } 1 \leq i \leq h\}$, and let $\underline{L}(\boldsymbol{u}_1, d_1; \boldsymbol{u}_2, d_2; \ldots; \boldsymbol{u}_h, d_h)$ be defined as the minimum of $L(\boldsymbol{u})$ over $\boldsymbol{u} \in V_N(\boldsymbol{u}_1, d_1; \boldsymbol{u}_2, d_2; \ldots; \boldsymbol{u}_h, d_h)$.

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 $\boldsymbol{u}_{\text{best}}$ denote the best of GC. Suppose that for $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, ..., $\boldsymbol{u}_h \in GC$ and positive integers d_1, d_2, ..., d_h, $\boldsymbol{u}_{\text{best}}$ is the best of $\cup_{i=0}^{h} \{\boldsymbol{u} \in B : d_H(\boldsymbol{u}, \boldsymbol{u}_i) < d_i\}$. If $\boldsymbol{u}_{\text{best}}$ satisfies the following condition,

$$L(\boldsymbol{u}_{\text{best}}) \leq \underline{L}(\boldsymbol{u}_1, d_1; \boldsymbol{u}_2, d_2; \ldots; \boldsymbol{u}_h, d_h), \quad (2)$$

then $\boldsymbol{u}_{\text{best}}$ is the optimal MLD solution in B. $\triangle\triangle$

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 $w_i \leq w_{0i}$. 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 $\boldsymbol{u} \in B$, minimum-weight search $(\boldsymbol{u}, B)$ denotes a search procedure for finding a codeword in B, denoted $\varphi_B(\boldsymbol{u})$, which has the least correlation discrepancy with respect to $\boldsymbol{z}$ among all codewords in B at the minimum Hamming distance w_{01} from $\boldsymbol{u}$. That is,

$$\varphi_B(\boldsymbol{u}) = \text{best}\{\boldsymbol{v} \in B : d_H(\boldsymbol{v}, \boldsymbol{u}) = w_{01}\}. \quad (3)$$

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

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

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

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

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

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

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

then

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

The condition (6), denoted Cond_R, is used as one of termination conditions for Iterative minimum-weight search $(\boldsymbol{u}, B)$ 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 $\boldsymbol{u}_{\text{best}}$, in the set of those candidate codewords which have been generated already, denoted GC, is optimum based on $\text{best}_h GC$ and the weight profile W of C, where h

is a specified small integer. From the Lemma, Cond_C is defined as

$$L(u_{\text{best}}) \leq \underline{L}(u_1, d_1; u_2, d_2; \ldots; u_l, d_l), \quad (8)$$

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

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

$$L(u_{\text{best}}) \leq \underline{L}(u_1, d_1; u_2, d_2; \ldots; u_{l_B}, d_{l_B}), \quad (9)$$

where $l_B = \min\{h, |GC_B|\}$ and for $1 \leq i \leq l_B$, $u_i \in \text{best}_h GC_B$ and if $\varphi_B(u_i) \in GC_B$, then $d_i = w_{02}$ and otherwise, $d_i = w_{01}$. For instance, if the minimum or the second minimum weight of C_0 is greater than that of C, then Cond_B is more effective than Cond_C only.

In the procedure, global variables GC_h and $GC_{B,h}$ are used besides GC and u_{best}. GC_h denotes the current value of $\text{best}_h GC$ and $GC_{B,h}$ denotes the current value of $\text{best}_h GC_B$. For $B \in C/C_0$, let $f(r, B)$ denote the initial candidate codeword in B for a given received sequence.

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 $GC_h \leftarrow \text{best}_h GC$. If Cond_C holds, then output u_{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 Cond_R, Cond_C or Cond_B holds. If Cond_C holds, then output u_{best} and stop. Suppose that Cond_R or Cond_B holds. If all cosets have been exhausted, then output u_{best} and stop; and otherwise, call Search-in(the coset next to B).
(D3) If either Cond_C or Cond_B 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-w_1.

3.3 Choice of the Initial Canidate 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 Λ_{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

$$|M_K \cap \Lambda_{N-K_0}| = K - K_0. \quad (10)$$

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:

$$u_i = x_i, \text{ for all } i \in M_K - \Lambda_{N-K_0}. \quad (11)$$

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

Then $g(z, B)$ can be chosen as the initial candidate codeword $f(r, B)$ for $B = C/C_0$, where z is the hard-decision binary vector obtained from the received sequence r. This $g(z, B)$ 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 Cond_B 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-w_1-w_2 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-w_1-w_2 and Algorithm II. The numbers depend on the complexities of subtrellises used in the simulation. The minimum weight subtrellis of the RM(64,22) code used in Algorithm II and the first and second weight subtrellis of the EBCH(64,24) code used in Algorithm I-w_1-w_2 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 Cond_B.

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-w_1, 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 RM(64,22) and RM(64,42) codes are shown in [6].

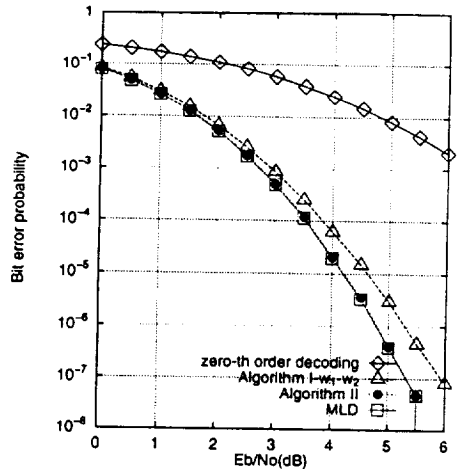

Figure 1: Bit error probabilities for the EBCH(64,24).

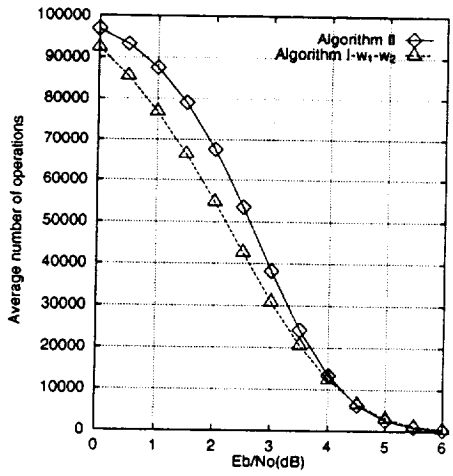

Figure 2: Average numbers of operations for EBCH(64,24).

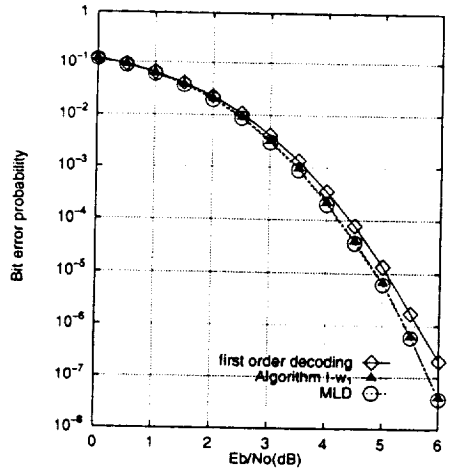

Figure 3: Bit error probabilities for the EBCH(64,45).

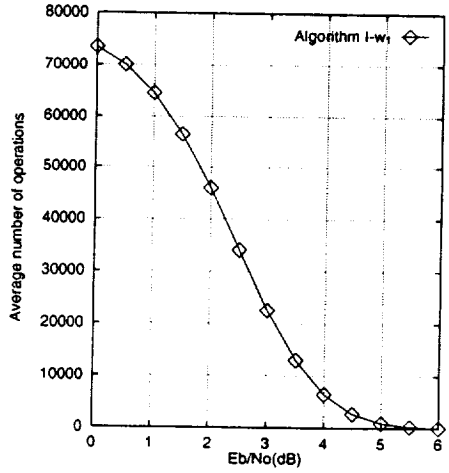

Figure 4: Average numbers of operations for EBCH(64,45).

References

[1] D. Chase, "A New Class for Decoding Block Codes with Channel Measurement Information," *IEEE Trans. Information Theory*, Vol. IT-18, pp.170–182, Jan. 1972.

[2] H. T. Moorthy, S. Lin and T. Kasami, "Soft-Decision Decoding of Binary Linear Block Codes Based on an Iterative Search Algorithm," submitted to *IEEE Trans. Information Theory*, 1995.

[3] T. Kasami, T. Takata, T. Koumoto, T. Fujiwara, H. Yamamoto and S. Lin, "The Least Stringent Sufficient Condition on Optimality of Suboptimal Decoded Codewords," *Technical Report of IEICE*, IT-94-82, The Inst. of Electronics, Information and Communication Engineers, Japan, Jan. 1995.

[4] M.P.C.Fossorier and S.Lin, "Soft Decision Decoding of Linear Block Codes Based on Ordered Statistics," *IEEE Trans. Information Theory*, Vol.IT-41, pp.1217–1234, Sept. 1995.

[5] M.P.C.Fossorier, S.Lin and J.Snyders, "On Maximum Likelihood Soft Decision Syndrome Decoding," *Proc. of IEEE International Symposium on Information Theory*, p.415, Sept. 1995.

[6] T. Koumoto, H. Nagano, T. Takata, T. Fujiwara, T. Kasami and S. Lin, "An Iterative Soft-Decision Decoding Algorithm," *Proc. of the 18th Symp. on Information Theory and Its Applications*, pp.557–560, Oct. 1995.