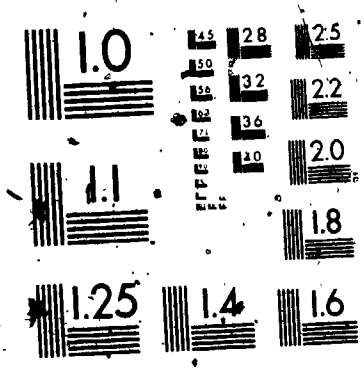

CANADIAN THESES ON MICROFICHE

I.S.B.N.

THÈSES CANADIENNES SUR MICROFICHE

National Library of Canada
Collections Development Branch

Canadian Theses on
Microfiche Service

Ottawa, Canada
K1A 0N4

Bibliothèque nationale du Canada
Direction du développement des collections

Service des thèses canadiennes
sur microfiche

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us a poor photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de mauvaise qualité.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE

NL-339 (r. 82/08)

Canada

§0.2 Hard decision & soft decision:

In data communication the received symbol is often processed by a "hard decision receiver" before decoding. If ρ is the received symbol, the hard decision receiver outputs 0 or 1 according to whether $\varphi(\rho) < 0$ or $\varphi(\rho) \geq 0$ respectively, where φ is defined above.

Likewise the analog hard decision receiver outputs 0 or 1 according to whether $k(\rho) \geq 0$ or $k(\rho) \leq 0$ respectively, where $k(\rho)$ is the quantization of ρ (e.g. if 0,1 are represented by the square waves $\_\sqcap\_$ and $\sqcap\_\sqcap$ respectively for transmission through a physical channel and $\rho(t)$ is the received wave, $k(\rho)$ may be taken as $\int \rho(t)\, dt$).

In general, use of hard decision receivers has the disadvantage of losing some information about the received symbols. On one hand it does not take into consideration the structure of the code used (e.g. the code being linear, cyclic, ... etc.). On the other hand, once a hard decision process has been used, the whole decoding process is not optimum in the sense of either minimizing the probability of symbol error or minimizing the probability of word error.

SOFT DECISION DECODING FOR BINARY LINEAR CODES

by

© Hesham El-Damhougy, B.Sc.

A thesis submitted to the Faculty of Graduate Studies in partial fulfilment of the requirements for the degree of Master of Science

Carleton University

Ottawa, Ontario

October, 1981

The undersigned hereby recommend to the Faculty of Graduate Studies acceptance of this thesis, submitted by Hesham El-Damhougy, B.Sc., in partial fulfilment of the requirements for the degree of Master of Science.

Kenneth S. Williams
.....
CHAIRMAN, DEPT. OF MATH. & STATS.

John D. Dixon
.....
SUPERVISOR

.....Irwin Reichstein.....
EXTERNAL EXAMINER

TABLE OF CONTENTS

ABSTRACT

This thesis is a survey of soft decoding methods for error correcting codes. First, two decoding rules which minimize the probability of symbol error are presented. One of them has complexity proportional to the size of the code and the other has complexity proportional to the size of the dual code. And then maximum-likelihood (optimal soft Decision) decoding is considered.

Three decoding algorithms are presented, the first two algorithms achieve maximum-likelihood decoding, but are theoretically inefficient. The third one is simple and is a good approximation for maximum-likelihood decoding.

Finally, the error-correcting capability of an algebraic analog decoding is considered when the decoding region is a hyper cone-like region which contains the transmitted codeword. In addition algebraic analog decoding via generalized minimum distance decoding is considered; both are quasi-soft decision decoding methods.

ACKNOWLEDGEMENT

I would like to extend my full appreciation to Professor John Dixon for his valuable assistance throughout this research.

Chapter 0

INTRODUCTION

§0.1 Description of the communication channel used:

By a channel we mean some medium which is available between the transmitter and the receiver over which a coded message will be sent. We assume that the input to our channel is binary i.e. the transmitter is restricted to two possible inputs, 0, 1, and that the outputs of the receiver lie in a subset $\mathcal{O} \subseteq \mathbb{R}$, where $\mathbb{R}$ is the set of real numbers (usually $\mathcal{O}$ is finite or else a finite interval).

The channel is by definition memoryless if for every input sequence $(\sigma_1, \sigma_2, \ldots, \sigma_n) \in \{1,0\}^n$ and for every output sequence $(\rho_1, \rho_2, \ldots, \rho_n) \in \mathcal{O}^n$,

$$p((\rho_1, \ldots, \rho_n)|(\sigma_1, \ldots, \sigma_n)) = \prod_{i=1}^{n} p(\rho_i|\sigma_i),$$

where $p(\rho|\sigma)$ is the probability that ρ will be received given σ was sent.

A memoryless channel is characterized by the function $p(\rho|\sigma)$, where $\rho \in \mathcal{O}$ and $\sigma \in \{0,1\}$, or equivalently is characterized by the function $\varphi(\rho) = \ell n[p(\rho|1)/p(\rho|0)]$, with the convention that $\ell n(0) = -\infty$ and $\ell n(1/0) = +\infty$.

§0.2 Hard decision & soft decision:

In data communication the received symbol is often processed by a "hard decision receiver" before decoding. If ρ is the received symbol, the hard decision receiver outputs 0 or 1 according to whether $\varphi(\rho) < 0$ or $\varphi(\rho) \geq 0$ respectively, where φ is defined above.

Likewise the analog hard decision receiver outputs 0 or 1 according to whether $k(\rho) \geq 0$ or $k(\rho) \leq 0$ respectively, where $k(\rho)$ is the quantization of ρ (e.g. if 0,1 are represented by the square waves ⎍⎍ and ⎍⎍ respectively for transmission through a physical channel and $\rho(t)$ is the received wave, $k(\rho)$ may be taken as $\int \rho(t)\, dt$).

In general, use of hard decision receivers has the disadvantage of losing some information about the received symbols. On one hand it does not take into consideration the structure of the code used (e.g. the code being linear, cyclic, ... etc.). On the other hand, once a hard decision process has been used, the whole decoding process is not optimum in the sense of either minimizing the probability of symbol error or minimizing the probability of word error.

For if $\hat{c} = (\hat{\gamma}_1, \ldots, \hat{\gamma}_n)$ is the decoded word after using a hard decision receiver and $c = (\gamma_1, \ldots, \gamma_n)$ is the transmitted codeword,

$$P(\hat{\gamma}_i \neq \gamma_1 | r) \geq \min_{\beta \in \{0,1\}} P(\beta \neq \gamma_1 | r), \quad i = 1, \ldots, n ;$$

$$P(\hat{c} \neq c | r) \geq \min_{\bar{c} \in C} P(\bar{c} \neq c | r),$$

where C is the code used. Usually we cannot expect these inequalities to be equalities.

Soft decision decoding is a general method which uses the channel measurements (i.e. $\omega(\rho)$ or $\varphi(\rho|\sigma)$) directly to decode the received word in such a way that the probability of error (symbol or word error) is minimum. In soft decision decoding the hard decision receiver is not used at all.

Soft decision decoding has been used for many years for convolutional codes [9]. But these methods seem to be less effective for long block codes. Recent work has been done on investigating soft decision decoding for block codes.

The object of this thesis is to present the most efficient and recent soft decision and analog decoding algorithms for binary block codes. The first soft decision method was proposed by Forney [2], and in chapter (3) we will give a brief description of this method.

In chapter (1) we present two soft decision decoding rules which are optimal in the sense of minimizing the probability of symbol error. In chapter (2) we present three soft decision decoding algorithms: the first two algorithms are optimal in the sense of minimizing the word error probability, the third one is an approximate algorithm for the first two algorithms. In chapter (3) we present a (quasi-) soft decision decoding rule which corrects the received word provided it lies in some specific region which contains the transmitted word.

All codes to be discussed are linear binary block codes.

Notation:

The following notational conventions will be used throughout this thesis. Lower case Greek letters will denote scalars; lower case Latin letters will denote vectors. Wherever possible we shall attempt to represent the components of a vector by the corresponding Greek letter. For example, unless otherwise stated the scalar γ_i is to be taken as the ith component of the vector c.

Chapter 1

TWO OPTIMUM SYMBOL-BY-SYMBOL DECODING RULES FOR BINARY LINEAR CODES

§1.1 Optimal symbol-by-symbol decoding rule:

Let $c = (\gamma_1, \gamma_2, \ldots, \gamma_n) \in \{0,1\}^n$ be any codeword of an (n,k) binary linear code C. A codeword c is transmitted over a time-discrete memoryless channel with output alphabet $\mathcal{O}$. The received word is denoted by $r = (\rho_1, \rho_2, \ldots, \rho_n) \in \mathcal{O}^n$. We consider the following decoding problem: given r and the index m compute an estimate $\hat{\gamma}_m$ of the transmitted code symbol γ_m such that $P(\hat{\gamma}_m = \gamma_m | r)$ is maximum, where $p(\hat{\gamma}_m = \gamma_m | r)$ is the probability that $\hat{\gamma}_m = \gamma_m$ given r is received. Assume that all codewords are equally likely to be transmitted and that $p(\rho_i|0)$, $p(\rho_i|1)$ are available for the decoder, $i = 1,\ldots,n$. Now

$$p(\gamma_m = \sigma | r) = \sum_{\substack{c \in C \\ \gamma_m = \sigma}} P(c|r),$$

and hence

$$p(\gamma_m = \sigma | r) = \sum_{\substack{c \in C \\ \gamma_m = \sigma}} p(r|c)[p(c)/p(r)] \quad \text{(Bayes rule)}.$$

Since the codewords are sent with equal probability and the channel is discrete memoryless,

$$P(c) = \frac{1}{|C|} = \frac{1}{2^k}$$

and

$$P(r|\sigma) = \prod_{i=1}^{n} P(\rho_i|\gamma_i).$$

Hence the following decoding rule maximizes $P(\hat{\gamma}_m = \gamma_m | r)$:

Decoding Rule I :

$$\hat{\gamma}_m = \begin{cases} 0 & \text{if } \sum_{\substack{c \in C \\ \gamma_m = 0}} \prod_{i=1}^{n} P(\rho_i|\gamma_i') > \sum_{\substack{c \in C \\ \gamma_m = 1}} \prod_{i=1}^{n} P(\rho_i|\gamma_i) \\ 1 & \text{otherwise.} \end{cases}$$

We note that the above decoding rule is also valid for non-linear codes in the sense it maximizes $P(\hat{\gamma}_m = \gamma_m | r)$. The above decoding rule requires computation of 2^k products $\prod_{i=1}^{n} p(\rho_i|\sigma)$, $\sigma \in \{0,1\}$, so it has complexity proportional to $2^k n$. This decoding rule can be used in practice only with codes having a small number of code-words.

In what follows we shall present another optimal symbol-by-symbol decoding rule which is dual to the above decoding rule in the sense that every word in the dual code is used in the decoding process. (In the above decoding rule every codeword is used in the decoding process).

§1.2 Polynomial representation of codes : (See [1])

Let $c = (\gamma_1, \ldots, \gamma_n) \in \{0,1\}^n$ be a codeword. Consider the $2n$ indeterminates x_{ij}, $i = 1,\ldots,n$, $j = 0,1$.

We define the homogeneous representation $x_c(x_{10}, x_{11}, \ldots, x_{n0}, x_{n1})$ of c to be the monomial $x_c(x_{10}, x_{11}, \ldots, x_{n0}, x_{n1}) = x_{1\gamma_1} x_{2\gamma_2} \ldots x_{n\gamma_n}$. We note that $x_c(\pi_{1\gamma_1}, \pi_{2\gamma_2}, \ldots, \pi_{n\gamma_n}) = p(r|c)$, where $\pi_{ij} = p(\rho_i|j)$, $j = 0,1$; $i = 1,\ldots,n$.

We define the homogeneous polynomial representation of C to be

$$C(x_1, \ldots, x_n) = \sum_{c \in C} x_c(x_{10}, x_{11}, \ldots, x_{n0}, x_{n1}), \text{ where}$$

$x_i = (x_{i0}, x_{i1})$.

Now let $P_i = (\pi_{i0}, \pi_{i1})$. Then

$$C(p_1, p_2, \ldots, p_n) = \sum_{c \in C} \prod_{i=1}^{n} \pi_{i\gamma_i} \qquad (*)$$

where $c = (\gamma_1, \ldots, \gamma_n)$.

Let $C_{ij}(p_1, \ldots, p_n)$ be the coefficient of π_{ij} in (*). Then the decoding rule (I) can be restated as:

Decoding Rule (I') :

$$\hat{\gamma}_m = \begin{cases} 0 & \text{if } \pi_{m0}\, C_{m0}(p_1,\ldots,p_n) > \pi_{m1}\, C_{m1}(p_1,\ldots,p_n) \\ 1 & \text{otherwise.} \end{cases}$$

§1.3 Transformation of the decoding rule (I') by duality : (See [1])

Let C' be the dual of the linear code C. In this section we find a relation between $C(x_1, \ldots, x_n)$ and $C'(x_1, \ldots, x_n)$ and from that we derive a decoding rule dual to the decoding rule (I').

Recall that a character χ (of degree 1) of a group G in C (the field of complex numbers) is a group homorphism

$$\chi : G \to (C \setminus \{0\}, \cdot).$$

The character which takes the value 1 everywhere is called trivial. Clearly the group $(\mathbb{Z}_2, +)$ has a unique non-trivial character χ, namely

$$\chi(0) = 1, \quad \chi(1) = -1$$

(more generally for any character χ of a finite cyclic group $G = \langle g \rangle$ of order n, there is an n^{th} root ξ of unity such that $\chi(g^i) = \xi^i$ for each i)

We shall need the following lemmas:

Lemma 1: (see [1])

Let χ be a non-trivial character of a ring G. For any integer $m \geq 1$ let $E = \bigoplus_{i=1}^{m} G$. Define $\chi_a(t) = \chi(a.t)$, $a,t \in E$. Then χ_a is a non-trivial character of E.

Proof:

Immediate from the definition.

Lemma 2: (standard result in group characters)

If H is a finite group and χ is a non-trivial character on H, then

$$\sum_{h \in H} \chi(h) = 0 .$$

Proof:

Since χ is non-trivial, then there exists $h' \in H$ such that $\chi(h') \neq 1$.

Now $\sum_{h \in H} \chi(h) = \sum_{h \in H+h'} \chi(h) = \chi(h') \sum_{h \in H} \chi(h)$.

Since $\chi(h') \neq 1$ then $\sum_{h \in H} \chi(h) = 0$.

Q.E.D.

Lemma (3): (see [1])

Let $V_n = \bigoplus_{i=1}^{n} \mathbb{Z}_2)$ and $C, C' \subseteq V_n$ be dual binary codes of length n. Let $f: V_n \to K$ be any mapping, where K is a vector space over C. Then

$$\sum_{v \in C} f(v) = \frac{1}{|C'|} \sum_{v \in C'} \sum_{u \in V_n} f(u) \chi(u.v),$$

where χ is the non-trivial character of $(\mathbb{Z}_2, +)$.

Proof:

$$\sum_{v\in C'} \sum_{u\in V_n} f(u)\,\chi(u.v) = \sum_{u\in V_n} f(u) \sum_{v\in C'} \chi(u.v)$$

$$= \sum_{u\in V_n} f(u) \sum_{v\in C'} \chi(u.v).$$

Since $\sum_{v\in C'} \chi(u.v) = \begin{cases} 0 & \text{if } u \notin C \\ |C'| & \text{if } u \in C \end{cases}$, by lemma (1), (2).

So

$$\sum_{u\in C} f(u) \sum_{v\in C'} \chi(u.v) = \sum_{u\in C} f(u)\,|C'|\,.$$

Hence

$$\sum_{v\in C} f(v) = \frac{1}{|C'|} \sum_{v\in C'} \sum_{u\in V_n} f(u)\,\chi(u.v).$$

Q.E.D.

Theorem: (See [1])

$$C(x_1, x_2, \ldots, x_n) = 2^{-(n-k)}\, C'(x_1H, x_2H, \ldots, x_nH),$$

where $C(x_1, \ldots, x_n)$, $C'(x_1, \ldots, x_n)$ are the homogeneous polynomial representation of C, C' respectively and H is the Hadamard matrix $\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Proof:

Let V_n be as in lemma (3).

Consider the mapping

$$f: V_n \to C[x_{10}, x_{20}, \ldots, x_{n0}, x_{11}, x_{21}, \ldots, x_{n1}]$$

$$(\nu_1 \ \nu_2 \ \cdots \ \nu_n) \longmapsto x_{1\nu_2} \ x_{2\nu_2} \ \cdots \ x_{n\nu_n}.$$

So $$\sum_{u \in V_n} f(u)\,\chi(u.v) = \sum_{\mu_1=0}^{1} \sum_{\mu_2=0}^{1} \cdots \sum_{\mu_n=0}^{1} \prod_{i=1}^{n} x_{i\mu_i}\,\chi(\mu_i \nu_i).$$

$$= \prod_{i=1}^{n} \Big(\sum_{j=0}^{1} x_{ij}\,\chi(j\nu_i) \Big).$$

By lemma (3) we have

$$C(x_1, \ldots, x_n) = \frac{1}{|C'|} \sum_{v \in C'} \prod_{i=1}^{n} \Big(\sum_{j=0}^{1} x_{ij}\,\chi(j\nu_i) \Big)$$

$$= \frac{1}{|C'|} \sum_{v \in C'} \prod_{i=1}^{n} (x_{i0} + x_{i1}\,\chi(\nu_i))$$

$$= 2^{-(n-k)}\, C'(x_1 H, \ldots, x_n H).$$

Q.E.D.

Now $C(x_1, \ldots, x_n) = 2^{-(n-k)}\, C'(y_1, \ldots, y_n)$ where

$y_i = x_i H$,

or $C(x_{10}, x_{11}, \ldots, x_{n1}) = 2^{-(n-k)}\, C'(y_{10}, \ldots, y_{n1})$

where $y_{ij} = \sum_{k=0}^{1} x_{ik}\, h_{kj}$, $H = [h_{ij}]$.

Hence

$$C_{ij} = \frac{\partial C}{\partial x_{ij}} = 2^{-(n-k)} \sum_{k=0}^{1} \frac{\partial C'}{\partial y_{ik}} \frac{\partial y_{ik}}{\partial x_{ik}},$$

$$= 2^{-(n-k)} \sum_{k=0}^{1} C'_{ik}\, h_{kj}.$$

So the decoding rule (I') is transformed to the following decoding rule:

Decoding Rule (II):

$$\hat{y}_m = \begin{cases} 0 \text{ if } \pi_{m0} \sum_{k=0}^{l} C'_{ik} (P_1 H, \ldots, P_n H)h_{k0} > \\ \qquad \pi_{ml} \sum_{k=0}^{l} C'_{ik} (P_1 H, \ldots, P_n H)h_{k_1} \\ 1 \text{ otherwise .} \end{cases}$$

Notes:

1) The complexity of the decoding rule (II), is obviously proportional to $|C'| = 2^{n-k}n$. This decoding rule can be used in practice only with codes whose dual code has a small number of codewords, i.e. high rate codes.

2) In general, the decoded word $\hat{c}$ (using decoding rule (I) or (II)) is not a codeword; but we can assume without loss of generality that the code C is systematic and then use either Decoding Rule (I) or (II) for the first k components to make a decision.

3) Both decoding rules, in general, are not optimal in the sense of minimizing the word error probability. In the next chapter we will present two decoding rules which are optimal in that sense.

Chapter 2

Maximum-Likelihood Decoding

(Optimal Soft Decision Decoding)

§2.1 Maximum Likelihood Decoder (Optimal Soft Decision Decoder):

Let C be a binary block code of length n (not necessarily linear). A codeword c is transmitted over a time discrete memoryless channel with output alphabet $\mathcal{O}$. The received word is $r = (\rho_1, \ldots, \rho_n) \in \mathcal{O}^n$.

A maximum likelihood decoder chooses the codeword $\hat{c}$ which maximizes $p(c|r)$. This decoder is optimal in the sense of minimizing the word error probability.

Assuming that the codewords are sent with equal probability, then we shall see (corollary 2.3) that $\hat{c}$ maximizes $p(c|r)$ iff $\hat{c}$ maximizes $f(r).c^*$, where $c^* = ((-1)^{\gamma_1}, \ldots, (-1)^{\gamma_n})$ and $f(r) = (\varphi(\rho_1), \ldots, \varphi(\rho_n))$.

In general, in order to find $\hat{c}$ which maximizes $p(c|r)$, the decoder must calculate $f(r).c^*$ for all codewords; so such a decoder must be exhaustive in the sense that the received word is compared to every word in the code.

Exhaustive decoders can be used in practice only for codes having a small number of codewords.

§2.2 An Optimal Decoding Algorithm:

Let C be an (n,k) binary linear code. $(C,\oplus)$ is an Abelian group, where $\oplus$ is component by component addition mod 2. Let $C^* = \{c^* = ((-1)^{\gamma_1}, \ldots, (-1)^{\gamma_n}) \mid c = (\gamma_1, \ldots, \gamma_n) \in C\}$ and x denote component by component multiplication. Obviously (C^*,x) is an Abelian group and $C \simeq C^*$ as Abelian groups, under the isomorphism

$$c \longmapsto c^*.$$

Now let r be the received word and put $f(r) = (\varphi(\rho_1),\varphi(\rho_2),\ldots\varphi(\rho_n))$ where φ is defined in §0.1. We are assuming the following:

1) The codewords of C are sent with equal probability;

2) The codewords of C are transmitted over a time-discrete memoryless channel.

Under these assumptions we have the following result.

Theorem 2.1: (See [3])

For all $c_1,c_2 \in C$, $p(c_1|r) \geq p(c_2|r)$ iff $(c_1 \oplus c_2).(f(r) \times c_1^*) \geq 0$.

Proof:

Since $p(c|r) = p(r|c)p(c)/p(r)$ (Bayes rule), and $p(c) = \frac{1}{2^k}$ (because of condition 1) above), then

$$p(c|r) = p(r|c)/2^k\, p(r) \quad \text{for all } c \in C.$$

Hence $p(c_1|r) \geq p(c_2|r)$ iff $p(r|c_1) \geq p(r|c_2)$.

Now since the channel is time-discrete and memoryless,

$$p(r|c) = \prod_{i=1}^{n} p(\rho_i|\gamma_i) \qquad \forall\ c \in C .$$

So $p(c_1|r) \geq p(c_2|r)$ iff $\sum_{i=1}^{n} \ell n\ p(\rho_i|\gamma_{1i}) \geq \sum_{i=1}^{n} \ell n\ p(\rho_i|\gamma_{2i})$

$$\text{iff} \quad \sum_{i=1}^{n} \ell n\ [p(\rho_i|\gamma_{1i})|p(\rho_i|\gamma_{2i})] \geq 0 . \tag{1}$$

Now

$$\ell n[p(\rho_i|\gamma_{1i})|\ p(\rho_i|\gamma_{2i})] = \begin{cases} \varphi(\rho_i)\gamma_{1i}^{*} & \text{if } \gamma_{1i} \neq \gamma_{2i} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

So (1) becomes

$$p(c_1|r) \geq p(c_2|r) \quad \text{iff} \quad (c_1 \oplus c_2).(f(r) \times c_1^{*}) \geq 0 .$$

Q.E.D.

The next five corollaries follow naturally from the above theorem.

Corollary 2.1:

For all $c_1, c_2 \in C$, $c_2.(f(r) \times c_1^{*}) \geq 0$ iff $p(c_1|r) \geq p(c_1 \oplus c_2|r)$.

Equivalently $c_2.(f(r) \times c_1^{*}) < 0$ iff $p(c_1|r) < p(c_1 \oplus c_2|r)$.

Corollary 2.2:

$\hat{c} \in C$ maximizes $p(c|r)$ iff $c.(f(r) \times \hat{c}^*) \geq 0$ for all codewords $c \in C$.

Corollary 2.3:

$\hat{c} \in C$ maximizes $p(c|r)$ iff $\hat{c}$ maximizes $f(r).c^*$.

Proof:

Let $A = \{i \mid \gamma_{1i} \neq \gamma_{2i}\}$ where $c_1 = (\gamma_{11}, \ldots, \gamma_{1n})$, $c_2 = (\gamma_{21}, \ldots, \gamma_{2n})$ are two codewords.

By equation (2) above, $p(c_1|r) \geq p(c_2|r)$ iff

$$\sum_{i \in A} \varphi(\rho_i)\gamma^*_{1i} \geq 0.$$

But $\sum_{i \in A} \varphi(\rho_i)\gamma^*_{1i} \geq 0$ iff

$$\sum_{i \in A} \varphi(\rho_i)\gamma^*_{1i} \geq \sum_{i \in A} \varphi(\rho_i)(-\gamma^*_{1i}) = \sum_{i \in A} \varphi(\rho_i)\,\gamma^*_{2i}.$$

Hence $p(c_1|r) \geq p(c_2|r)$ iff

$$f(r).c_1^* = \sum_{i \in A \cup A^c = \{1,\ldots,n\}} \varphi(\rho_i)\gamma^*_{1i} \geq \sum_{i \in A \cup A^c = \{1,\ldots,n\}} \varphi(\rho_i)\gamma^*_{2i} = f(r).c_2^*$$

because $\gamma^*_{i1} = \gamma^*_{i2}$ for all $i \in A^c$.

Q.E.D.

Corollary 2.4:

$\hat{c} \in C$ maximizes $p(c|r)$ iff $\hat{c}$ minimizes $f(r).c$.

Proof:

By corollary (2.3) and since $c^* = (11 \ldots 1) - 2c \quad \forall\, c \in C$, the proof follows immediately.

Q.E.D.

Corollary 2.5:

$\hat{c} \in C$ maximizes $p(c|r)$ iff $\hat{c}$ minimizes $\|f(r) - c^*\| = [f(r).f(r) - 2f(r).c^* + c^*.c^*]^{\frac{1}{2}}$.

Proof:

By corollary (2.3) and since $c^*.c^* = n \quad \forall\, n \in C$, the proof follows immediately.

Q.E.D.

Definition: (See [3])

The projecting subset of C, denoted by Proj(C), is defined as the smallest subset of C such that for any non-zero $c \in C$, there is a non-zero $c_p \in$ proj (C) such that

$$c \times c_p = c_p .$$

In other words, if $c \in C$ then $c \in$ Proj(C) iff no non-zero code-word can be obtained from c by setting one or more 1's equal to 0.

Lemma 2.1:

For each $c \in C$, there exists $\ell \geq 0$ and $c_1, \ldots, c_\ell \in \text{proj}(C)$ such that

$$c = c_1 \oplus \ldots \oplus c_\ell$$

and $c_1, \ldots, c_\ell$ are mutually disjoint (i.e. no two of them have 1's in the same position or equivalently $c_i . c_j = 0$, $i \neq j$).

Proof:

If $c = 0$ take $\ell = 0$, so suppose $c \neq 0$.

Since $c \in C$, then by definition there exists $c_1 \in \text{proj}(C)$ such that $c \times c_1 = c_1$. Hence one can write $c = c_1 \oplus (c \ominus c_1)$, where c_1 and $c \ominus c_1$ are mutually disjoint.

Note that $c \ominus c_1$ has fewer 1's than c. If $c \ominus c_1 = 0$ we are finished. Otherwise we can repeat the above process a finite number of times until we get

$$c = c_1 \oplus \ldots \oplus c_\ell ,$$

where $c_1, \ldots, c_\ell \in \text{proj}(C)$ are mutually disjoint.

Q.E.D.

Theorem 2.2:

Let $x = f(r) \times c_m^*$, where $c_m \in C$. Then c_m maximizes $p(c|r)$ iff $c_p . x \geq 0$ for all $c_p \in \text{proj}(C)$.

Proof:

By lemma (2.1), if $c \in C$ then there exists mutually disjoint $c_1, \ldots, c_\ell \in \text{proj}(C)$ such that $c = c_1 \oplus \ldots \oplus c_\ell$.

Since $c_1, \ldots, c_\ell$ are mutually disjoint, $c.x = c_1.x + \ldots + c_\ell.x$. Thus if $c_p.x \geq 0$ for all $c_p \in \text{proj}(C)$, then $c.x \geq 0$ for all $c \in C$. Hence by corollary (2.2) c_m maximizes $p(c|r)$.

If c_m maximizes $p(c|r)$ then by corollary (2.2) $c.x \geq 0$ for all $c \in C$. In particular $c_p.x \geq 0$ for all $c_p \in \text{proj}(C)$.

Q.E.D.

Now consider the following decoding algorithm.

Decoding Algorithm I: (See [3])

For a received word $r = (\rho_1, \ldots, \rho_n)$

1) Calculate $f(r) = (\varphi(\rho_1), \ldots, \varphi(\rho_n))$;

2) Select a codeword $c_m \in C$;

3) If $c_p.(f(r) \times c_m^*) \geq 0$ for all $c_p \in \text{prj}(C)$ then output c_m, Else;

4) When there is a $c_p \in \text{proj}(C)$ such that $c_p.(f(r) \times c_m^*) < 0$, replace c_m by $c_m \oplus c_p$ and go to 3).

Theorem 2.3:

The Decoding Algorithm I terminates after a finite number of steps and the output maximizes $p(c|r)$.

Proof: (The proof is very similar to the proof of theorem 4 of [3])

i) By theorem (2.2) if $c_p . x \geq 0$ for all $c_p \in \text{proj}(C)$ then c_m maximizes $p(c|r)$, where $x = f(r) \times c_m^*$.

ii) If $c_p . x < 0$ for some $c_p \in \text{proj}(C)$ then by corollary (2.1), $p(c_m|r) < p(c_m \oplus c_p|r)$. So we test if $c_m \oplus c_p$ will maximize $p(c|r)$ at step 3).

Since there is a finite number of codewords c and the value of the dot product with x is strictly decreasing, this decoding process will terminate with the output $c_m \oplus c_1 \oplus \ldots$ such that

$$c_p . (f(r) \times (c_m \oplus c_1 \oplus \ldots)^*) \geq 0 \quad \text{for all } c_p \in \text{proj}(C).$$

Hence by theorem (2.2), $c_m \oplus c_1 \oplus \ldots$ maximizes $p(c|r)$.

Q.E.D.

Remarks:

i) Theoretically the Decoding Algorithm I is inefficient in many cases (see table (1) below) compared to the exhaustive decoder, since the complexity of the Decoding Algorithm I is proportional to $|\text{proj}(C)|^2$.

But Hwang [3] in his computer simulation for that Algorithm found that the decoding output was always reached within two or three loops of $|\text{proj}(C)|$ checks.

ii) If all components of $f(r) \times c_m^*$ are ≥ 0 then $c.(f(r) \times c_m^*) \geq 0$ for all $c \in C$. In particular $c_p.(f(r) \times c_m^*) \geq 0$ $\forall\ c_p \in \text{proj}(C)$, so there is no need to check all $c_p.(f(r) \times c_m^*)$ to give the output c_m. So we can modify the algorithm in step 3) to check for such a case.

(n,k)	δ_H	|projC|	|C|
(15,11)	3	308	2048
(15,10)	4	385	1024
(15, 9)	4	255	512
(17, 9)	5	340	512
(17, 8)	6	221	256
(21,15)	4	1848	32768
(21,11)	6	1386	2048
(23,12)	7	3335	4096
(23,11)	8	1794	2048
(31,25)	4	23653	33554432

(Table (1), See [3])

(δ_H is the minimum (Hamming) weight of C).

§2.3 A Simple Decoding Algorithm:

In this section we give two decoding algorithms. The first one achieves maximum-likelihood decoding but is inefficient compared to the exhaustive decoder. The second one in general does not achieve maximum-likelihood decoding but it is simpler than both Decoding Algorithm I and the exhaustive decoder, and can be considered as a good approximate algorithm for maximum-likelihood decoding.

Lemma 2.2:

Let $c_1, c_2 \in C$ and $x = (\xi_1, \ldots, \xi_n) \in \mathbb{R}^n$. Then $c_1 \cdot x < c_2 \cdot x$ iff $c_1^* \cdot x > c_2^* \cdot x$.

Proof:

$$c_i^* = (11\ldots1) - 2c_i \quad \text{so} \quad c_1^* \cdot x - c_2^* \cdot x = -2(c_1 \cdot x - c_2 \cdot x)$$

Q.E.D.

By the above lemma and corollary (2.3), the output of the following decoding algorithm maximizes $p(c|r)$.

Decoding Algorithm II:

For a received word $r = (\rho_1, \ldots, \rho_n)$

1) Calculate $f(r) = (\varphi(\rho_1), \ldots, \varphi(\rho_n))$;

2) Select a codeword $c_m \in C$ and calculate $x := f(r) \times c_m^*$,

3) If $c_p.x \geq 0$ for all $c_p \in \text{proj}(C)$ then output c_m,

Else,

4) Find mutually disjoint $\hat{c}_1, \ldots, \hat{c}_\ell \in \text{proj}(C)$ which minimize $\sum_{i=1}^{t} c_i.x$ for all mutually disjoint sequences $c_1, \ldots, c_t \in \text{proj}(C)$ such that $c_i.x < 0$. Output $c_m \oplus (\hat{c}_1 \oplus \ldots \oplus \hat{c}_\ell)$.

Remarks:

i) By lemma (2.1) we note that the number of disjoint sequences in $\text{proj}(C) \geq |C|$. So the complexity of the Decoding Algorithm II is greater than or equal to $|C|$. So in general the exhaustive decoder is more efficient than the above algorithm.

ii) Now the codeword $\hat{c}$ which maximizes $p(c|r)$ is the nearest codeword to $f(r)$ (see corollary 2.5), so we can choose c_m as near as possible to $f(r)$. Since $f(r) \simeq \lambda(\pm 1, \ldots, \pm 1)$, where $\lambda \in \mathbb{R}$, one good choice for $c_m = (\gamma_{m1}, \ldots, \gamma_{mn})$ is such that $\omega(\rho_i)\gamma^*_{mi} > 0$, $i = 1, \ldots, k$ (assuming without loss of generality that C is systematic).

Hence the following decoding algorithm may be considered as a good approximation for the Decoding Algorithm II.

Decoding Algorithm III: (for a systematic code C)

For a received word $r = (\rho_1, \ldots, \rho_n)$

1) Calculate $f(r) = (\varphi(\rho_1), \ldots, \varphi(\rho_n))$, and find the unique codeword c_m such that $\varphi(\rho_i)\gamma^*_{mi} \geq 0$, $i = 1,\ldots,k$. Calculate $x := f(r) \times c^*_m$.

2) Set $c_1 = 0$.

3) For all $c \in \text{proj}(C)$ Do

If $c.x < 0$ then

If $c.c_1 = 0$ then $c_1 := c_1 \oplus c$

Else If $c.x < c_1.x$ then $c_1 := c$.

4) Output $c_m \oplus c_1$.

Note:

The complexity of the above algorithm is proportional to $|\text{proj}(C)|$.

§2.4 Appendix:

Hwang [5] gave an optimal decoding algorithm. But unfortunately this decoding algorithm, as we will see now, is incorrect.

We follow Hwang [5] in presenting his decoding algorithm.

Let $r = (\rho_1, \ldots, \rho_n)$ be the received word, consider $f(r) = (\varphi(\rho_1), \ldots, \varphi(\rho_n))$. Assume without loss of generality that

$$\min_{1\le i\le k} |\varphi(\rho_i)| \ge \max_{k+1\le j\le n} |\varphi(\rho_j)| .$$

Since k positions will determine at least one codeword, a codeword c_1 can be chosen such that

$$x := f(r) \times c_1^* \text{ has its first k components} \ge 0$$

i.e. $\xi_i := \varphi(\rho_i) \times \gamma_1^* \ge 0, \quad i = 1, \ldots, k.$

By corollary (2.1) if c_1 does not maximize $p(c|r)$, then there exists a codeword $c = (\gamma_1, \ldots, \gamma_n)$ such that

$$c \cdot (f(r) \times c_1^*) < 0 .$$

Hence

$$\min_{1\le i\le k} |\xi_i| \left(\sum_{i=1}^{k} \gamma_i\right) \le \sum_{i=1}^{k} \gamma_i \xi_i < -\sum_{j=k+1}^{n} \gamma_j \xi_j \le \left(\sum_{j=k+1}^{n} \gamma_j\right) \max_{k+1\le j\le n} |\xi_j|$$

which implies

$$\sum_{i=1}^{k} \gamma_i < \sum_{j=k+1}^{n} \gamma_j .$$

More generally if we define

$$C_r = \{c \in C \mid \sum_{i=1}^{k} \gamma_i < \sum_{j=k+1}^{n} \gamma_j\},$$

then each $c \in C$ satisfying

$$c.(f(r) \times c_1^*) < 0$$

lies in C_r.

So consider the following algorithm.

Decoding Algorithm: (see [5])

For a received word $r = (\rho_1, \ldots, \rho_n)$

1) Calculate $f(r)$ and find some $c_1 \in C$ such that $x := f(r) \times c_1^*$ has its first k components ≥ 0.

2) If $c.x \geq 0$ for all $c \in C_r$ then Output c_1, otherwise

3) Output $\hat{c} := c_1 \oplus c_t$ where c_t minimizes $c.x$.

Remarks:

i) The claim that "k positions will determine at least one codeword" is in general false. So the above decoding algorithm will fail to find c_1 in step 1) for some received words r.

ii) Even if there exist some codes for which the above claim is true, the decoding algorithm seems to be complicated compared to the exhaustive decoder. It requires computing C_r for each received word r because the claim "without loss of generality" requires the fact that C_r will vary with different received words r.

Chapter 3

ALGEBRAIC ANALOG DECODING FOR BINARY LINEAR CODES

In this chapter we consider the error correcting capability of an algebraic analog decoder for binary linear codes; by analog we mean that the output alphabet of the communication channel is the set $\mathbb{R}$ of all real numbers. The received word $r = (\rho_1, \rho_2, \ldots, \rho_n) \in \mathbb{R}^n$, is processed by a "demodulation function" $f: \mathbb{R} \to [-1,1]$ which satisfies certain prescribed conditions.

The first decoding method to be discussed is a symbol-by-symbol decoding method. Practically this method can be used efficiently for binary cyclic codes.

§3.1 A symbol-by-symbol analog decoding rule: (See [4],[8])

We consider the class of demodulation functions $f: \mathbb{R} \to [-1,1]$ where $\mathbb{R}$ is the set of real numbers which satisfy the following conditions:

D1) f is a continuous non-increasing function.

D2) For all $\xi \in \mathbb{R}$, $f(\xi) = -f(1-\xi)$.

D3) For any integer $m \geq 1$, $\prod_{i=1}^{m} f(\xi_i) \geq f\left(\sqrt{\sum_{i=1}^{m} \xi_i^2}\right)$.

D4) For any two integers m,k with $m \geq 1$, $0 \leq k \leq m$ we have

$$\sum_{i=1}^{m} \xi_i^2 \leq k/4 \Rightarrow \sum_{i=1}^{m} F(\xi_i) > m-k .$$

For example, one function which satisfies the above conditions is

$$f(\xi) = \begin{cases} 1 & \xi \leq 0 \\ \cos(\pi\xi) & 0 < \xi < 1 \\ -1 & \xi \geq 1 \end{cases}$$

(The proof can be found in [4], [8]).

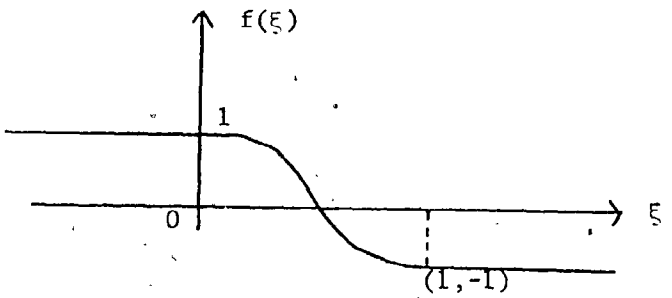

Fig. (1)

Now suppose that the codeword $c = (\gamma_1, \gamma_2, \ldots, \gamma_n) \in \{0,1\}^n$ is transmitted and $r = (\rho_1, \rho_2, \ldots, \rho_n) \in \mathbb{R}^n$ is received.

Denote by $S_\alpha(c)$ the n-dimensional ball of radius $\alpha/2$ and center c i.e. $S_\alpha(c) = \{x \in \mathbb{R}^n \mid \delta_E(x,c) < \alpha/2\}$, where $\delta_E(x,c)$ is the Euclidean distance between x and c.

Let $V_\alpha(c) = \{y \in \mathbb{R}^n \mid y = x - z_c$ for some $x \in S_\alpha(c)$ and $z_c = ((-1)^{\gamma_1}\xi_1, (-1)^{\gamma_2}\xi_2, \ldots, (-1)^{\gamma_n}\xi_n)$ where $\xi_i \geq 0, i = 1,\ldots,n\}$. Then $V_\alpha(c)$ is a hyper cone-like region which contains $S_\alpha(c)$ in the sense that for all $u = x_1 - z_{1c}$, $v = x_2 - z_{2c} \in V_\alpha(c)$,

$\lambda u + \lambda v \in V_\alpha(c)$ whenever $\lambda, \mu \geq 0$ and $\mu u + \lambda v \in S_\alpha(c)$

Let C' be the dual code of C, and choose $c'_1, c'_2, \ldots, c'_J \in C'$, $c'_i = (\gamma'_{i1}, \ldots, \gamma'_{in})$, such that $\gamma'_{i1} = 1, \quad 1 \leq i \leq J$. Set $\lambda_i = \sum_{j=1}^{J} \gamma'_{ji}$ and $\lambda = \max_{1 \leq i \leq J} \lambda_i$.

Let

$$F_1(r) = \lambda f(\rho_1) + \sum_{j=1}^{J} \prod_{i=2}^{n} (f(\rho_i))^{\gamma'_{ji}} \tag{3.1}$$

where f satisfies conditions D1-D4 above, with the convention that $0^0 = 1$.

Denote by $\hat{c}$ an estimate of c. Consider the following decoding rule.

Decoding Rule (III): (See [4],[8])

$$\hat{\gamma}_1 = \begin{cases} 1 & \text{if } F_1(r) \leq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Then we have:

Theorem (3.1):

If $r \in V_\sigma(c)$, where $\sigma = [(J+\lambda)/\lambda]^{\frac{1}{2}}$, then the decoding rule (III) gives $\hat{\gamma}_1 = \gamma_1$.

Proof:

We shall show that $(-1)^{\gamma_1} F_1(r) > 0$ whenever $r \in V_\sigma(c)$.

From equation (3.1) above we have

$$(-1)^{\gamma_1} F_1(r) = \lambda (-1)^{\gamma_1} f(\rho_1) + \sum_{j=1}^{J} (-1)^{\gamma_1} \prod_{i=2}^{n} (f(\rho_i))^{\gamma'_{ji}}. \tag{3.2}$$

Let $1 \le j \le J$. Then since $c.c' \equiv 0 \pmod 2$ for all $c' \in C'$, and $\gamma'_{ji} = 1$ we have $(-1)^{\gamma_1} = \prod_{i=2}^{n} (-1)^{\gamma_1 \gamma'_{ji}}$.

So we may rewrite (3.2) as

$$(-1)^{\gamma_1} F_1(r) = \lambda (-1)^{\gamma_1} f(\rho_1) + \sum_{j=1}^{J} \prod_{i=2}^{n} ((-1)^{\gamma_i} f(\rho_i))^{\gamma'_{ji}}.$$

Since $r \in V_\sigma(c)$, we have $\rho_i = \xi_i - (-1)^{\gamma_i} \xi_i$ for some $z = (\xi_1, \ldots, \xi_n)$, $x = (\xi_1, \xi_2, \ldots, \xi_n) \in S_\sigma(c)$, $\xi_i \ge 0$.

Now if $\gamma_i = 1$ then condition D2 shows $(-1)^{\gamma_i} f(\rho_i) = f((-1)^{\gamma_i}(\rho_i - \gamma_{ij}))$ and if $\gamma_i = 0$ then the same identity is trivial.

By D1, f is non-increasing so $f((-1)^{\gamma_i}(\rho_i - \gamma_i)) \ge f((-1)^{\gamma_i}(\xi_i - \gamma_i))$. Hence $(-1)^{\gamma_i} f(\rho_i) \ge f((-1)^{\gamma_i}(\xi_i - \gamma_i))$, and so putting $\eta_i = (-1)^{\gamma_i}(\xi_i - \gamma_i)$ we get

$$(-1)^{\gamma_i} F_1(r) \ge \lambda f(\eta_1) + \sum_{j=1}^{J} \prod_{i=2}^{n} (f(\eta_i))^{\gamma'_{ji}}.$$

By condition D3, since all $\gamma'_{ji} = 0$ or 1 we get

$$(-1)^{\gamma_1} F_1(r) \ge \lambda f(\eta_1) + \sum_{j=1}^{J} f\left[\sqrt{\sum_{i=2}^{n} \eta_i^2 \gamma'_{ji}}\right]. \tag{3.3}$$

Since $\sum_{j=1}^{J} \gamma'_{ji} = \lambda_i$,

$$\sum_{i=1}^{n} \eta_i^2 = \frac{1}{\lambda} [\lambda \eta_1^2 + \sum_{j=1}^{J} \sum_{i=2}^{n} \eta_i^2 \gamma'_{ji} + \sum_{i=2}^{n} (\lambda - \lambda_i) \eta_i^2]$$

where $\lambda = \max_{1 \leq i \leq n} \lambda_i$.

Then since $\sum_{i=1}^{n} \eta_i^2 = \sum_{i=1}^{n} (\xi_i - \gamma_i)^2 < \frac{1}{4} (J+\lambda) / \lambda$ and $\lambda - \lambda_i \geq 0$, we have

$$\lambda \eta_1^2 + \sum_{j=1}^{J} \sum_{i=2}^{n} \eta_i^2 \gamma'_{ji} < (J+\lambda) / 4 .$$

Then since $\lambda f(\eta_1) = \sum_{k=1}^{\lambda} f(\eta_1)$ and condition D4 we have

$$\lambda f(\eta_1) + \sum_{j=1}^{J} f[\sqrt{\sum_{i=2}^{n} \eta_i^2 \gamma'_{ji}}] > (J+\lambda) - (J+\lambda) = 0 .$$

Hence

$$(-1)^{\gamma_1} F_1(r) > 0 .$$

Q.E.D.

Corollary (3.1)

If $\sigma = (J+\lambda) / \lambda$, then the sets $V_\sigma(c)$ $(c \in C)$ are pairwise disjoint.

Corollary (3.2)

$\sigma = (J+\lambda) / \lambda \leq \delta_H$; where δ_H is the minimum Haming distance (weight) of C .

Proof:

Let δ_E be the minimum Euclidean distance between pairs of elements of C. Clearly $\delta_E = \sqrt{\delta_H}$.

Q.E.D.

Now let H be the parity check matrix of C. Then the rows of H are parity checks i.e. define equations

$$\sum_{i=1}^{n} h_i \gamma_i = 0$$

with each $h_i = 0$ or 1, which every codeword c must satisfy.

Definition (3.1) (See [6])

A set of parity check equations is called orthogonal on the ith coordinate if the coefficient of γ_i is 1 in each equation, but no other γ_j has a coefficient 1 in more than one equation in the set.

Definition (3.2) (See [7])

A linear code C is called 1-step orthogonalizable, if for each i, $i = 1,\ldots,n$, there are at least $\delta_H - 1$ parity checks orthogonal on the ith coordinate where δ_H is the minimum Hamming distance of C.

We have immediately the following.

Theorem 3.2:

If C is 1-step orthogonalizable, then we can find J parity checks such that $\lambda = 1$ and $(J+\lambda)/\lambda = \delta_H$ (so the bound of corollary 3.2 is possible in this case).

Proof:

By the above definition, we can find $J = \delta_H - 1$ parity checks orthogonal on the first coordinate. Since these parity checks are orthogonal, $\lambda = 1$.

Q.E.D.

Notes:

1) The complexity of the Decoding Rule (III) is proportional to J.

2) The Decoding Rule (III) can be implemented efficiently for cyclic codes by storing $f(r) = (f(\rho_1), \ldots, f(\rho_n))$ in a cyclic analog shift register. (See Fig. (2)).

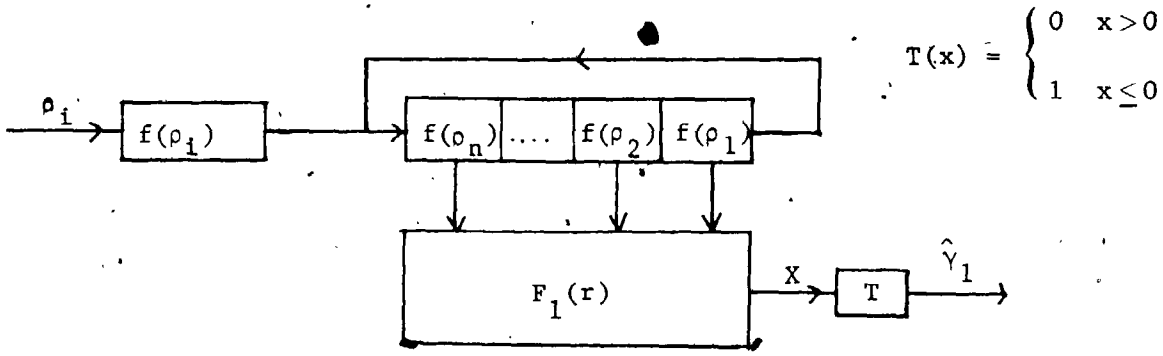

Fig. (2)

§3.2 Generalized Minimum Distance Decoding (GMDD)

Generalized minimum distance decoding was proposed by Forney [2, Chapter 3] to link the gap between probabilistic (maximum likelihood) decoding and Hamming distance (algebraic) decoding.

In this section we will give a brief description of this method and then use it in the next section in relation to algebraic analog decoding.

We shall use the following two theorems whose proofs can be found in [2].

Notation:

For each $c = (\gamma_1, \ldots, \gamma_n) \in \{0,1\}^n$ we define $c^* = ((-1)^{\gamma_1}, \ldots, (-1)^{\gamma_n}) \in \{-1,1\}^n$.

Theorem 3.3:

For any $a = (\alpha_1, \ldots, \alpha_n) \in [-1,1]^n$, there exists at most one codeword $c \in C$, such that

$$a.c^* > n - \delta_H. \tag{3.4}$$

Remark:

We note that if $a \in \{-1,1\}^n$ then $a.c^* = n - 2\delta_H(a,c^*)$, where $\delta_H(a,c^*)$ is the Hamming distance between a and c^*, so in this special case if condition (3.4) is satisfied and a is the

received word, a can be decoded correctly using an algebraic decoding method.

We shall only discuss decoding when condition (3.4) is satisfied. This is called generalized minimum distance decoding.

Let $a = (\alpha_1, \ldots, \alpha_n) \in [-1,1]^n$. Order the indices $i_1, \ldots, i_n$ so that

$$|\alpha_{i_1}| \leq |\alpha_{i_2}| \leq \ldots \leq |\alpha_{i_n}| .$$

For each ℓ, $1 \leq \ell \leq n$, define

$$\beta_\ell(\alpha_{ij}) = \begin{cases} 0 & \text{if } 1 \leq j \leq \ell \\ -1 & \text{if } \alpha_{ij} \geq 0 \text{ and } j > \ell \\ 1 & \text{if } \alpha_{ij} < 0 \text{ and } j > \ell \end{cases}$$

(The coordinate 0 is usually called an erasure).

So the ℓ smallest coordinates in absolute value are erased and the others replaced by 1 or -1.

Consider the vectors $b_\ell = (\beta_\ell(\alpha_1), \ldots, \beta_\ell(\alpha_n))$, $\ell = 1, \ldots, n$. Then

Theorem (3.4):

If for some codeword c we have

$$a.c^* > n - \delta_H, \tag{3.5}$$

then for some ℓ, $1 \leq \ell \leq n$, $b_\ell.c^* > n - \delta_H$.

If condition (3.4) is satisfied for some $c \in C$, the decoding problem is to find the unique c which satisfies (3.4).

Now clearly $b_\ell . c^* = n - 2\delta_\ell - \theta$, where δ_ℓ is the Hamming distance between c^* and b_ℓ in the unerased coordinates and θ is the total number of erasures in b_ℓ.

So $b_\ell . c > n - \delta_H$ implies $2\delta_\ell + \theta < \delta_H$.

Set $\hat{b}_\ell = (\hat{\beta}_{\ell_1}, \ldots, \hat{\beta}_{\ell_n})$, where

$$\beta_{\ell_i} = \begin{cases} 0 & \text{if } \beta_\ell(\alpha_i) = 1 \\ 1 & \text{if } \beta_\ell(\alpha_i) = -1 \\ ? & \text{if } \beta_\ell(\alpha_i) = 0 \end{cases}$$

and ? is an erasure symbol.

Under the hypothesis of Theorem (3.3) we can find c by applying any algebraic decoding method to the vector $\hat{b}_\ell$ as the received word (e.g. See [7], pp. 305-307).

Note that there is no point trying the $\hat{b}_\ell$'s which have more than $\delta_H - 1$ erasures, namely all $\hat{b}_\ell$'s, $\ell \geq \delta_H$, for in this case $b_\ell . c^* \leq n - \delta_H \ \forall\ c \in C$. Also it is unnecessary to try the $\hat{b}_\ell$'s for which $\delta_H - \ell$ is even, for if such $\hat{b}_\ell$ gives c then $\delta_H(\hat{b}_\ell, c^*) < (\delta_H - \ell)/2 - 1$ (since ℓ is the number of erasures, we must have $2\delta_\ell + \ell \leq \delta_H - 1$), but then $\hat{b}_{\ell+1}$ must also give c. So only those $\hat{b}_\ell$, for which $\ell < \delta_H$ and $\delta_H - \ell$ is odd need to be

tried in the decoding process for finding the unique codeword c, i.e. at most $\lfloor (\delta_H + 1)/2 \rfloor$ trials are needed in the decoding process.

Forney [2, Chapter 3] considered GMDD when

$$a = (\alpha_1(\rho_1), \ldots, \alpha_n(\rho_n)); \quad \alpha_i(\rho_i) = \begin{cases} -1 & \varphi_i(\rho_i) \geq T \\ 0 & -T < \varphi_i(\rho_i) < T \\ 1 & \varphi_i(\rho_i) \leq -T \end{cases}$$

where $r = (\rho_1, \ldots, \rho_n)$ is the received word, $\varphi_i(\rho_i) = \ell n[p(\rho_i|1)/p(\rho_i|0)]$ and T is some threshold. He showed that the decoding method is optimum (in the sense of minimizing a bound on the probability of not decoding correctly or failing to decode at all) if $T = 2\theta$ where θ is the total number of erasures in r. For errors- only (no erasures) this decoding method is optimal if

$$\alpha_i(\rho_i) = \begin{cases} -1 & \text{if } \varphi_i(\rho_i) \geq 0 \\ 1 & \text{if } \varphi_i(\rho_i) < 0 \end{cases}$$

§3.3 Algebraic Analog Decoding via GMDD: (See [4])

Suppose we are not able to find a set J of parity checks such that $(J+\lambda)/\lambda = \delta_H$ i.e. we are unable to maximize σ in Theorem (3.1). In this section we will use the GMDD to decode the received word r, which will correct r whenever $r \in V_{\frac{\delta_E}{2}}(c)$ (δ_E is the minimum Euclidean distance between pairs of distinct codewords in $\mathcal{C}$).

Theorem 3.5:

Let $f(r) := (f(\rho_1), \ldots, f(\rho_n))$ where f satisfies conditions D1-D4. Then there is at most one codeword c such that

$$f(r).c^* > n - \delta_H .$$

Proof:

By letting $a = f(r)$, the result follows from Theorem (3.3).

Q.E.D.

Theorem 3.6:

If $r \in V_{\frac{\delta_E}{2}}(c)$, then $f(r).c^* > n - \delta_H$.

Proof: (the proof is very similar to the proof of Theorem 4 in [4]).

Let $r \in V_{\frac{\delta_E}{2}}(c)$, then $r = x - z$ for some $x \in S_{\frac{\delta_E}{2}}(c)$ and some $z = ((-1)^{\gamma_1}\xi_1, \ldots, (-1)^{\gamma_n}\xi_n)$, $\xi_i \geq 0$.

Since f is non-increasing,

$$f((-1)^{\gamma_i}(\rho_i - \gamma_i)) \geq f((-1)^{\gamma_i}(\xi_i - \gamma_i)).$$

But by condition D2,

$$f((-1)^{\gamma_i}(\rho_i - \gamma_i)) = (-1)^{\gamma_i} f(\rho_i)$$

and

$$f((-1)^{\gamma_i}(\xi_i - \gamma_i)) = (-1)^{\gamma_i} f(\xi_i),$$

and so

$$\sum_{i=1}^{n} (-1)^{\gamma_i} f(\rho_i) = f(r).c^* \geq \sum_{i=1}^{n} (-1)^{\gamma_i} f(\xi_i) = f(x).c^* .$$

Since $x \in S_{\delta_E}(c)$,

$$\sum_{i=1}^{n} (\xi_i - \gamma_i)^2 = \sum_{i=1}^{n} [(-1)^{\gamma_i}(\xi_i - \gamma_i)]^2 < \delta_E^2/4 = \delta_H/4 ,$$

so by condition D4 - we have

$$f(x).c^* = \sum_{i=1}^{n} (-1)^{\gamma_i} f(\xi_i) = \sum_{i=1}^{n} f((-1)^{\gamma_i}(\xi_i - \gamma_i)) > n - \delta_H .$$

Hence

$$f(r).c^* \geq f(x).c^* > n - \delta_H .$$

Q.E.D.

Corollary:

The $V_{\delta_E}(c)$ (for $c \in C$) are pairwise disjoint.

By the above theorem we can use the GMDD to decode r correctly if $r \in V_{\partial_E}(c)$. One disadvantage of this method is that if $r \notin V_{\partial_E}(c)$ for all codewords $c \in C$ then the GMDD will fail to decode r at all.

REFERENCES

[1] G. Battail, M.C. Decouvelaere, and P. Godlewski, "Replication decoding", IEEE Trans. Inform. Theory, 25 (1979), 332-345.

[2] G.D. Forney, Jr., "Concatenated Codes", Cambridge, MA: M.I.T., 1966.

[3] T.Y. Hwang, "Decoding linear block codes for minimizing word error rate", IEEE Trans. Inform. Theory, 25 (1979), 733-737.

[4] T.Y. Hwang, "On the error-correcting capability of algebraic analog decoding", IEEE Trans. Inform. Theory, 26 (1980), 107-109.

[5] T.Y. Hwang, "Efficient optimal decoding of linear block codes", IEEE Trans. Inform. Theory, 26 (1980), 603-606.

[6] F.J. MacWilliams and N.J.A. Sloane, "The theory of error correcting codes", North-Holland Pub. Co., 1978.

[7] W.W. Peterson and E.J. Weldon, Jr., "Error-correcting codes", 2nd ed., Cambridge, MA: M.I.T., 1972.

[8] L.D. Rudolph, G.R.P. Hartmann, T.Y. Hwang, and N.Q. Duc, "Algebraic analog decoding of linear binary codes", IEEE. Trans. Inform. Theory, 25 (1979), 430-440.

[9] Viterbi, Andrew and Omura, Jim, "Digital Communication and Coding", New York: McGraw-Hill Book Co., 1978.

END

06-06-83

FIN