論文 90-27-2-2

線形 2元 블럭 符號를 위한 軟判定 復號 알고리듬

(A Soft-Decision Decoding Algorithm for Linear Binary Block Codes)

沈 龍 杰,* 李 忠 雄**

(Yong Geol Shim and Choong Woong Lee)

要 約

本 論文에서는 線形 2元 블럭 符號에 대한 軟判定 復號 알고리듬을 제안하였다. 이 알고리듬은 블럭 誤謬 確率을 最小化하기 위한 것이다.

提案된 알고리듬을 既存의 다른 復號法과 비교하기 위하여 (7, 4) Hamming 符號와 (23, 12) Golay 符號에 대한 컴퓨터 시뮬레이션을 수행하였다. 그 결과, 提案된 알고리듬의 平均 硬判定 復號 回數는 항상 2以下이며 信號對 雜音比가 충분히 클 때 1에 접근하였다. 즉, 既存의 다른 復號法에 비하여 復號의 複雜度가 減少되었음을 확인하였다.

Abstracts

A soft-decision decoding algorithm for linear binary block codes is proposed, for minimizing the block error probability.

To compare the proposed algorithm with already established decoding methods, computer simulations are performed for the (7,4) Hamming code and the (23,12) Golay code. The average number of hard-decision decoding is always less than 2, and approaches to 1 when the signal to noise ratio is sufficiently large. These results show that the proposed algorithm reduces the decoding complexity.

I. 序 論

誤謬 訂正 符號는 블럭 符號와 컨벌루셔널 符號로 구분이 되며, 復號法에는 硬判定 및 軟判定 復號法이 있다. 本 論文에서는 블럭 符號에 대하여 블럭誤謬 確率을 最小로 하는 새로운 軟判定 復號 알고리듬을 提案하고자 한다.

*正會員, 檀國大學校 電子工學科

(Dept. of Elec. Eng., Dankook Univ.)

**正會員, 서울大學校 電子工學科

(Dept. of Elec. Eng., Seoul Nat'l Univ.)

接受日字：1989年 8月 30日

블럭 誤謬 確率을 最小로 하려면 受信된 信號 벡터로부터 가장 가까운 거리에 있는 符號語를 찾아내야 한다. 이에 대한 最尤復號法(maximum likelihood decoding)으로는 相關復號法(correlation decoding)이 알려져 있다.

相關復號法은 受信된 信號 벡터를 모든 符號語와 비교하여 가장 相關度가 큰 符號語를 선택하는 방법이다. 따라서 誤謬 確率의 측면에서는 가장 우수한 性能을 나타내는 반면에 符號語의 수가 대단히 많은 경우에는 사용할 수 없다.

약간의 성능 低下를 감수하면서 復號의 複雜度를 줄이는 方法으로 Forney의 generalized minimum distance 알고리듬[1]과 Chase의 알고리듬[2]을 비롯

한 여러가지 方法[3-6] 들이 硏究되어 있다. 이 方法들은 먼저 硬判定 復號를 수행하여 얻어진 符號語를 포함하면서, 적은 갯수의 符號語를 元素로 갖는 集合을 구성하고, 이 集合에 속한 符號語들 중에서 受信된 信號 벡터와의 相關度가 가장 큰 符號語를 復號 結果로 선택한다. 이 集合에 소속될 符號語들은 硬判定 復號에 의하여 얻어진다. 따라서, 復號의 複雜度는 硬判定 復號를 수행하는 回數 또는 相關度를 비교해야할 符號語의 數에 의하여 결정된다.

Chase는 Forney의 generalized minimum distance 알고리듬[1]의 개념을 더욱 발전 시켜서 세가지의 알고리듬을 제안하였다.[2] Chase의 세가지 알고리듬 중 가장 실용적이며, 본 論文의 알고리듬과의 비교대상으로 선정된 알고리듬 3을 간단히 설명하면 다음과 같다. 수신된 信號 비트들 중에서 가장 信賴度가 낮은 i개의 비트를 반전시켜서 硬判定 復號를 수행한다. 符號의 최소 거리 d가 홀수인 경우 i는 0, 2, 4, …, (d−1)의 값을 갖는다. 이렇게 硬判定 復號를 (d+1)/2번 수행하여 얻어진 符號語들을 수신 信號와 비교하여 가장 相關度가 높은 符號語를 선택한다. 후에 연구된 Hacket[4]와 Tendolka 등[6]의 방법도 동일한 개념 위에서 확립된 것이다.

Tanaka 등[5]은 信賴度의 threshold를 정해 놓고 이 threshold 보다 信賴度가 낮은 비트들을 erasure로 간주하여 이 비트들을 반전시켜 가면서 硬判定 復號를 수행한다. 만약 threshold 보다 信賴度가 낮은 비트 수가 Ne 이면 2^{Ne}번의 硬判定 復號 과정이 필요하다.

결국, 기존의 방법은 信賴度가 낮은 비트들을 여러가지로 반전시켜 가면서 硬判定 復號를 반복 수행하여 얻어진 符號語들 중 수신 벡터와의 相關度가 가장 큰 것을 선택하는 것이라 할 수 있다. 그런데, 최초의 硬判定 復號法으로 얻어진 符號語를 면밀히 관찰하면 차후의 復號과정에 큰 도움을 줄 수 있는 정보를 얻을 수 있는데, 기존의 방법들은 이 정보를 이용하지 않고 있다. 본 論文에서는 이 정보를 활용하여 수신 벡터와의 相關度가 커지는 방향에 존재하는 符號語를 쉽게 찾아낼 수 있는 방법을 개발함으로써 전체 復號에 소요되는 硬判定 復號 횟수를 2회 이하로 줄이고자 한다.

즉, 최초의 硬判定 復號法으로 얻어진 符號語로부터 假定된 誤謬의 갯수 및 위치, 信賴度가 작은 비트와 假定된 誤謬가 一致하는 정도에 관한 情報를 추출한다. 이 情報를 活用하여 受信된 信號벡터와의 相關度가 큰 符號語를 찾아낸다. 이 符號語와 최초의 硬判定 復號法으로 얻어진 符號語를 비교하여 受信된 信號 벡터와의 相關度가 더 큰 符號語를 軟判定 復號 結果로 선택한다.

또한, 信號對雜音比가 충분히 클 때에는 최초의硬判定 復號만으로 最尤復號가 달성되는 경우가 많다. 本 硏究에서는 최초의 硬判定 復號로 얻어진 符號語가 最尤復號 結果임을 판단할 수 있는 조건을 찾아내고, 이 조건이 만족되는 경우에는 復號를 終了함으로써 復號過程이 대폭 간소화될 수 있도록 한다.

Ⅱ. 復號 알고리듬

軟判定 復號法을 사용하는 디지탈 通信 시스템의 模型을 그림 1에 나타내었다. 最小 Hamming 거리가 d인 (n, k) 線形 2元 블럭 符號 C의 符號語를 $\mathbf{c}=(c_1, c_2, \cdots, c_n)$으로 表示한다. 여기서, $c_i \in GF(2)$이다. 이 符號語를 反極性(antipodal)變調 方式을 사용하여 送信하는 경우 符號語 C는 反極性 信號 벡터 $\mathbf{s}=(s_1, s_2, \cdots, s_n)$으로 변환되며, 여기서 s_i는 $s_i=\sqrt{E_s}(-1)^{c_i}$이고, E_s는 심볼당 平均 에너지이다. 情報비트당 에너지를 E_b라 하면 $E_b=E_s/R$이고, R은 符號率로서 k/n이 된다.

傳送路에 가해진 雜音을 片側 雜音 電力密度 N_0인 相加的 白色가우시안 雜音(AWGN)이라 假定할 때, 受信 信號 벡터 $\mathbf{r}=(r_1, r_2, \cdots, r_n)$은 $\mathbf{s}$에 雜音 $\mathbf{z}=(z_1, z_2, \cdots z_n)$를 더한 벡터이다. 즉 $r_i=s_i+z_i$이며, z_i는 平均이 0이고 分散이 $N_0/2$인 가우시안 不規則 變數이다. 또한, $z_1, z_2, \cdots, z_n$은 모두 相互 獨立이다.

受信측에서는 $\mathbf{r}$에 대한 判定을 행하여 硬判定 벡터 $\mathbf{Y}=(Y_1, Y_2, \cdots, Y_n)$과 信賴度 벡터 $\mathbf{a}=(a_1, a_2, \cdots,$

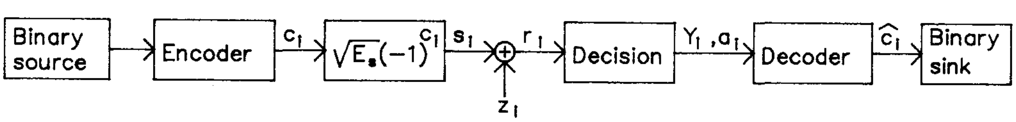


그림 1. 디지탈 通信 시스템의 模型
Fig. 1. Model of digital communication system.

a_n)을 얻어낸다. 硬判定은

$$Y_i = \begin{cases} 0, & r_i \geq 0 \\ 1, & \text{otherwise} \end{cases} \quad (i=1,2,\cdots,n) \qquad (1)$$

이고, 信賴度는

$$a_i = r_i \qquad (i=1,2,\cdots,n) \qquad (2)$$

이다.

復號器는 送信側에서 傳送한 符號語일 것으로 추정되는 벡터 $\hat{\mathbf{c}}=(\hat{c}_1, \hat{c}_2, \cdots, \hat{c}_n)$를 $\mathbf{Y}$와 $\mathbf{a}$로부터 찾아내야 한다. $\mathbf{c}$와 $\mathbf{Y}$의 차이 즉, 誤謬 벡터를 $\mathbf{e}=(e_1, e_2, \cdots, e_n)$이라 하면 $\mathbf{e}=\mathbf{Y}\oplus\mathbf{c}=(Y_1\oplus c_1, Y_2\oplus c_2, \cdots, Y_n\oplus c_n)$이다. 여기서 $\oplus$는 2진 덧셈을 나타낸다. $\mathbf{e}$의 Hamming 重(weight)을 $W_H(\mathbf{e})$로 表示한다. 또한, $\mathbf{e}$의 아날로그 重을 $W_a(\mathbf{e})$로 표시하고, 다음과 같이 정의한다.

$$W_a(\mathbf{e}) = \sum_{i=1}^{n} a_i e_i \qquad (3)$$

符號 C에 속한 모든 符號語 중에서 $W_a(\mathbf{e})$를 最小로 하는 符號語를 찾아내어 그것을 $\hat{\mathbf{c}}$로 하면 된다.

硬判定 벡터 $\mathbf{Y}$를 硬判定 復號法으로 復號하여 얻어진 符號語를 $\mathbf{c}_h=(c_{h1}, c_{h2}, \cdots, c_{hn})$이라 하면, $\mathbf{c}_h$에 대한 誤謬 벡터 $\mathbf{e}_h$는

$$\mathbf{e}_h = \mathbf{Y}\oplus\mathbf{c}_h \qquad (4)$$

이다.

만약 $\mathbf{e}_h=\mathbf{0}$ ($\mathbf{0}$은 영벡터)이면 $W_a(\mathbf{e}_h)=0$이다. 그런데, (2)식에서 $a_i \geq 0$이므로 (3)식으로 定義된 $W_a(\mathbf{e})$의 最小값 $W_{a,min}$은 영

$$W_{a,min} = \min_{c\in C} W_a(\mathbf{e}) \geq 0$$

가 된다. 따라서, 만약 $W_a(\mathbf{e}_h)=0$이면 $W_{a,min}=W_a(\mathbf{e}_h)=0$이므로 $\mathbf{c}_h$를 復號 結果로 하여 $\hat{\mathbf{c}}=\mathbf{c}_h$로 하고 復號를 終了한다. 이 때, $\hat{\mathbf{c}}$는 最尤復號法에 의한 復號 結果와 一致한다.

만약 $\mathbf{e}_h \neq 0$이면

$$W_{a,min} = \min\{W_a(\mathbf{e}_h),\ W_a'\} \qquad (5)$$

이다. 여기서 W_a'는 $\mathbf{c}_h$ 이외의 符號語들에 대한 誤謬의 아날로그 重의 最小값이다. 즉,

$$W_a' = \min_{\substack{c\in C \\ c\neq c_h}} W_a(\mathbf{e}) = \min_{\substack{c\in C \\ c\neq c_h}} W_a(\mathbf{c}\oplus\mathbf{Y})$$

이다. (4)식에서 $\mathbf{Y}=\mathbf{c}_h\oplus\mathbf{e}_h$ 이므로

$$W_a' = \min_{c\neq c_h} W_a(\mathbf{c}\oplus\mathbf{c}_h\oplus\mathbf{e}_h)$$

이다. 그런데 符號 C는 線形 符號이므로 $\mathbf{c}\oplus\mathbf{c}_h\in C$이고, $\mathbf{c}_h$가 아닌 모든 $\mathbf{c}$에 대하여 $\mathbf{c}\oplus\mathbf{c}_h\neq 0$이다. 따라서,

$$W_a' = \min_{\substack{c\in C \\ c\neq 0}} W_a(\mathbf{c}\oplus\mathbf{e}_h)$$

$$= \min_{\substack{c\in C \\ c\neq 0}} \sum (c_i\oplus e_{hi})a_i \quad c\ e \qquad (6)$$

이다. $a_i\geq 0$ 이므로 (6)식이 最小로 되려면 $c_i\oplus e_{hi}$의 값들은 대부분 0이 되어야 하고, 1로 되는 경우에는 그에 해당하는 信賴度 a_i값이 가능한한 작아야 한다. 즉, 0이 아닌 符號語 중에서 이러한 符號語 $\mathbf{c}$를 찾아내야 한다. 그런데 이러한 符號語를 직접 찾아내는 것은 대단히 어려운 일이다. 따라서 우선 2元 벡터 $\mathbf{v}=(v_1, v_2, \cdots, v_n)$을 생각하여 $W_a(\mathbf{v}\oplus\mathbf{e}_h)$가 最小로 되도록 $\mathbf{v}$를 구성한다. $W_H(\mathbf{c})$ $(\mathbf{c}\neq\mathbf{0})$의 最小값은 符號의 最小 Hamming 거리 d이므로,

$$W_H(\mathbf{v}) = d \qquad (7)$$

인 2元 벡터 $\mathbf{v}$를 생각해야 한다. $W_a(\mathbf{v}\oplus\mathbf{e}_h)$를 最小로 하려면, $e_{hi}=1$일 때는 $v_i=1$로 놓아 $v_i\oplus e_{hi}=0$이 되도록 하고 $e_{hi}=0$ 일 때는 信賴度 a_i가 작은 경우에만 $v_i=1$로 놓아야 한다. 이 점에 착안하여 集合 $I=\{i: v_i=1\}$를 구성하는 방법을 다음과 같이 설명한다.

먼저, $J=\{j: e_{hj}=1\}$인 集合을 생각한다. 集合 J의 元素의 數는 $n(J)=W_H(\mathbf{e}_h)\leq(d-1)/2$이다. I는 J를 포함해야 한다. 즉,

$$I \supset J \qquad (8)$$

이어야 한다. 그리고, 集合 I의 元素의 數는 (7)식으로부터

$$n(I) = W_H(\mathbf{v}) = d \qquad (9)$$

이어야 한다. 集合 J^c를 $J^c=\{j: e_{hj}=0\}$이라 한다. 集合 $\{a_k: k\in J^c\}$의 모든 元素들을 크기가 작은 順序대로 排列했을 때, a_k가 제 m 번째 元素이면 $k=f(m)$으로 表示한다. 즉 $a_{f(1)}\leq a_{f(2)}\leq\cdots\leq a_{f(n(J^c))}$이다. 이때,

$$I-J=\{f(1), f(2), \cdots, f(d-n(J))\} \qquad (10)$$

이 되도록 한다. (8), (9) 및 (10) 식으로 表示되는 조건들을 만족하도록 集合 I를 구성하면 벡터$\mathbf{v}$를 얻을수 있다. 이때,

$$W_a' \geq W_a(\mathbf{v}\oplus\mathbf{e}_h) \qquad (11)$$

로 된다. $\mathbf{v}$는 符號語가 아닐 수도 있다. $\mathbf{v}$가 符號語로 될 때 (11)식의 等號가 성립한다.

$W_a(\mathbf{v}\oplus\mathbf{e}_h)$와 $W_a(\mathbf{e}_h)$를 비교하여, 만약 $W_a(\mathbf{v}\oplus\mathbf{e}_h) \geq W_a(\mathbf{e}_h)$이면, (5)식과 (11)식으로부터 $W_{a,min}=W_a(\mathbf{e}_h)$이므로 $\mathbf{c}_h$를 復號 結果로 하여 $\hat{\mathbf{c}}=\mathbf{c}_h$로 하고 復號를 終了한다. 이 때 $\hat{\mathbf{c}}$는 最尤復號法에 의한 復號結果와 一致한다.

만약 $W_a(\mathbf{v}\oplus\mathbf{e}_h) < W_a(\mathbf{e}_h)$이면 $\mathbf{v}$와 가장 가까운 符號語를 찾기 위하여 v를 硬判定 復號法으로 復號한다. 이렇게 얻어진 符號語를 $\mathbf{c}_m$이라 하고, (6)식을 最小로 하는 符號語가 $\mathbf{c}_m$일 것으로 추정한다. 이때, $\mathbf{e}_s=\mathbf{c}_m\oplus\mathbf{e}_h$라 하고 $W_a(\mathbf{e}_s)$와 $W_a(\mathbf{e}_h)$를 비교한다. 만약 $W_a(\mathbf{e}_s) \geq W_a(\mathbf{e}_h)$이면 $\mathbf{e}_h$를 아날로그 重이 가장 작은 誤謬 벡터로 간주하여 $\hat{\mathbf{c}}=\mathbf{c}_h$로 한다. 만약 $W_a(\mathbf{e}_s) < W_a(\mathbf{e}_h)$이면 $\mathbf{e}_s$를 아날로그 重이 가장 작은 誤謬 벡터로 간주하여 $\mathbf{c}_s=\mathbf{Y}+\mathbf{e}_s$로 하고, $\hat{\mathbf{c}}=\mathbf{c}_s$로 한다. 이때의 $\hat{\mathbf{c}}$는 最尤復號法에 의한 復號結果와 일치하지 않을 수도 있다.

提案된 알고리듬을 요약하면 다음과 같다.

[1단계] 硬判定 復號法으로 $\mathbf{Y}$를 復號하여 $\mathbf{c}_h$를 얻는다.
$\mathbf{e}_h$를 $\mathbf{Y}\oplus\mathbf{c}_h$로 놓는다.

[2단계] 만약 $\mathbf{e}_h=0$이면, $\mathbf{c}\hat{}$를 $\mathbf{c}_h$로 놓고 8단계로 간다.

[3단계] $W_a(\mathbf{e}_h)$를 계산한다.

[4단계] 벡터 $\mathbf{v}$를 구성하고 $W_a(\mathbf{v}\oplus\mathbf{e}_h)$를 계산한다.

[5단계] 만약 $W_a(\mathbf{v}\oplus\mathbf{e}_h) \geq W_a(\mathbf{e}_h)$이면, $\hat{\mathbf{c}}$를 $\mathbf{c}_h$로 놓고 8단계로 간다.

[6단계] 硬判定 復號法으로 $\mathbf{v}$를 復號하여 $\mathbf{c}_m$을 얻는다.
$\mathbf{e}_s$를 $\mathbf{c}_m\oplus\mathbf{e}_h$로 놓는다.
$W_a(\mathbf{e}_s)$를 계산한다.

[7단계] 만약 $W_a(\mathbf{e}_s) \geq W_a(\mathbf{e}_h)$이면, $\hat{\mathbf{c}}$를 $\mathbf{c}_h$로 놓는다.
만약 $W_a(\mathbf{e}_s) < W_a(\mathbf{e}_h)$이면,
$\mathbf{c}_s$를 $\mathbf{Y}\oplus\mathbf{e}_s$로 놓고, $\hat{\mathbf{c}}$를 $\mathbf{c}_s$로 놓는다.

[8단계] $\hat{\mathbf{c}}$를 復號 結果로 한다.

(7, 4) Hamming 符號의 경우를 예로하여 본 알고리듬의 수행 과정을 살펴보면 다음과 같다. 受信된 硬判定 벡터가 $\mathbf{Y}=(1,1,1,0,0,0,0)$이고 信賴度 벡터가 $\mathbf{a}=(0.03, 0.01, 0.02, 0.5, 0.6, 0.4, 0.5)$인 경우를 생각한다.

[1단계] 硬判定 復號法으로 $\mathbf{Y}$를 復號하면 $\mathbf{c}_h=(1,1,1,0,0,1,0)$이고,
$\mathbf{e}_h=(0,0,0,0,0,1,0)$이다.

[2단계] $\mathbf{e}_h \neq \mathbf{0}$이므로 3단계로 간다.

[3단계] $W_a(\mathbf{e}_h)=a_6=0.4$이다.

[4단계] 集合 $J=\{j : e_{hj}=1\}=\{6\}$, $J^c=\{j : e_{hj}=0\}=\{1,2,3,4,5,7\}$이다.
$\{a_k : k\in J^c\}=\{a_1, a_2, a_3, a_4, a_5, a_7\}$의 모든 元素를 크기가 작은 順序대로 排列하면 $a_2 \leq a_3 \leq a_1 \leq a_4 \leq a_7 \leq a_5$ 이므로 $f(1)=2, f(2)=3, f(3)=1, f(4)=4, f(5)=7, f(6)=5$이다.
(8)식에서 $I \supset \{6\}$이고, (9)식에서 $n(I)=d=3$ 이며, (10)식에서 $I-J=\{f(1), f(2)\}=\{2,3\}$이다. 따라서 $I=\{2,3,6\}$이고 $\mathbf{v}=(0,1,1,0,0,1,0)$이 된다. 이 때, $\mathbf{v}\oplus\mathbf{e}_h=(0,1,1,0,0,0,0)$이므로 $W_a(\mathbf{v}\oplus\mathbf{e}_h)=a_2+a_3=0.01+0.02=0.03$이다.

[5단계] $W_a(\mathbf{v}\oplus\mathbf{e}_h) < W_a(\mathbf{e}_h)$이므로 6단계로 간다.

[6단계] 硬判定 復號法으로 $\mathbf{v}$를 復號하면 $\mathbf{c}_m=(1,1,0,0,1,0)$이고, $\mathbf{e}_s=(1,1,1,0,0,0,0)$, $W_a(\mathbf{e}_s)=a_1+a_2+a_3=0.03+0.01+0.02=0.06$이 된다.

[7단계] $W_a(\mathbf{e}_s) < W_a(\mathbf{e}_h)$ 이므로 $\hat{\mathbf{c}}=\mathbf{c}_s=\mathbf{Y}\oplus\mathbf{e}_s=(0,0,0,0,0,0,0)$이다.

[8단계] 復號 結果는 $\hat{\mathbf{c}}$이다.

Ⅲ. 시뮬레이션

Hamming 符號와 Golay 符號는 2원 符號이면서 완전 符號(perfect code)이다. 즉 동일한 갯수의 誤謬를 정정하는 다른 符號에 비하여 符號率을 높일 수 있기 때문에 많이 사용되고 있다. 이러한 이유로 본 연구에서는 이 두 符號를 시뮬레이션 대상으로 하였으며, 시뮬레이션 실행 시간을 고려하여 (7, 4) Hamming 符號와 (23, 12) Golay 符號를 선택하였다.

사용한 컴퓨터는 삼성 SSM 286(IBM PC AT 호환 기종)이며, MS FORTRAN으로 프로그램 하였다. 시뮬레이션 환경은 相加的 白色 가우시안 雜音이 있는 傳送路를 통하여 反極性 信號를 사용하는 것으로 假定하였다. 信號 에너지 E_s는 1로 정규화하였다. 雜音으로는 平均이 0이고, 分散이 $N_0/2$인 가우시안 不規則 變數를 사용하였다. 그리고 送信측에서 전송하는 데이타는 모두 "0"으로 假定하였다. (線形 符號를 사용하고 있으므로 이렇게 假定하여도 일반성을 잃지 않는다.)

本 論文에서 提案한 알고리듬과 함께 通常的인 硬判定 復號法, Chase의 알고리듬(알고리듬 3)[2], 相關復號法에 대한 시뮬레이션을 행하여 그 結果를 비교하였다. 특히 Chase의 알고리듬은 현재까지 알려진 方法들 중 가장 대표적이고, 本 論文의 알고리듬과 밀접한 관계가 있다는 점에서 비교의 대상으로 선정되었다.

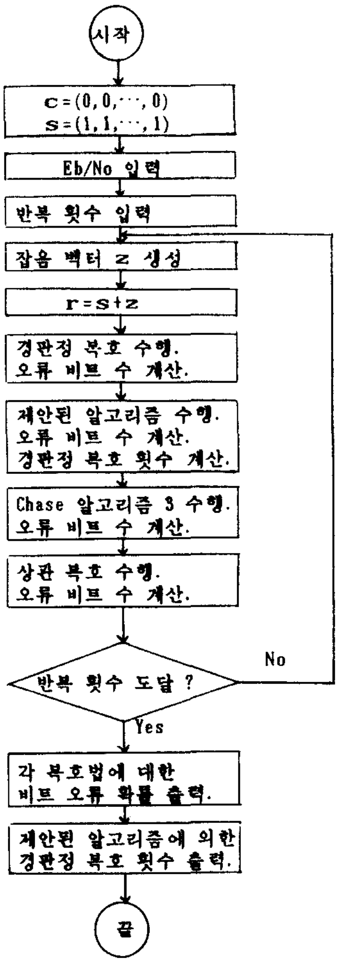


그림 2. 시뮬레이션의 흐름도
Fig. 2. Flow chart of simulation.

그림2는 시뮬레이션의 흐름도(flow chart)이다. 送信 符號語 **c**의 성분은 모두 0으로 假定하였고, 反極性 信號의 심볼당 平均에너지 E_s는 1로 정규화 하였으므로 信號 벡터**s**의 성분은 모두 1이다. 가우시안 不規則 變數 발생 프로그램에 의하여 雜音 벡터 **z**를 생성한다. 受信 信號 벡터**r**을 각각의 信號法으로 復號했을 때, 데이타 1이 나타나면 送信 데이타는 모두 1이었으므로 誤謬 비트로 계산한다. 비트 誤謬確率은 情報 비트 중에서 발생한 誤謬 비트의 수를 총 정보 비트로 나누어 계산한다. 제안된 알고리듬에서의 硬判定 復號 횟수는 하나의 符號語를 復號할 때, 6단계를 수행하지 않고 復號를 완료하면 硬判定 復號를 1번 수행하는 것이고, 6단계를 수행하면 硬判定 復號를 2번 수행하는 것으로 계산할 수 있다.

復號法의 複雜度는 硬判定 復號 回數와 相關度를 비교해야 할 符號語의 數에 의하여 결정된다. 표 1과 표 2에 각각(7, 4) Hamming 符號와 (23, 12) Golay符號에 대하여 여러가지 E_b/N_0 값에 따른 시뮬레이션 결과를 나타내었다. 硬判定 復號法의 경우는 항상 1회의 硬判定 復號 과정이 필요하다. 提案된 알고리듬의 경우는 平均 硬判定 復號 回數를 나타내었으며, 이것은 相關度를 비교해야 할 符號語 수의 平均과同一하다. 提案된 알고리듬에서 平均 硬判定 復號 回數는 항상 2보다 작고, E_b/N_0가 충분히 크면 1에 접근하는 것을 알 수 있다. Chase의 알고리듬 3의 경우는 硬判定 復號 回數를 나타내었다.(7, 4) Hamming 符號의 경우에는 최소 거리가 3이므로 그대로 硬判定 復號한 결과와 신뢰도가 낮은 2개의 비트를 반전시켜서 復號한 결과를 비교해야 하므로 硬判定 復號 횟수는 항상 2회이다. (23, 12) Golay 符號의 경우에는 최소 거리가 7이므로 반전시키는 비트 수가 0, 2, 4, 6으로 되어 항상 4 회의 硬判定 復號과정이 필요하다. 相關復號法의 경우는 相關度를 비교해야 할 符號語의 수를 나타내었으며, 이것은 符號語 전체의 수와 同一하다. 즉(7, 4) Hamming 符號의 경우 $2^4=16$개이며, (23, 12) Golay 符號의 경우 $2^{12}=4,096$ 개

표 1. (7, 4) Hamming 符號에 대한 硬判定 復號 및 相關의 平均 回數
Table 1. Average number of hard-decision decoding and correlation for the (7, 4) Hamming code.

E_b/N_0[dB] / 復號法	0.0	1.0	2.0	3.0	4.0	5.0	6.0
Hard-decision Decoding	1	1	1	1	1	1	1
Proposed Algorithm	1.258	1.194	1.132	1.081	1.043	1.021	1.008
Chase Algorithm 3	2	2	2	2	2	2	2
Correlation Decoding	16	16	16	16	16	16	16

표 2. (23, 12) Golay 符號에 대한 硬判定 復號 및 相關의 平均 回數
Table 2. Average number of hard-decision decoding and correlation for the (23, 12) Golay code.

E_d/N_0[dB] / 復號法	0.0	1.0	2.0	3.0	4.0	5.0	6.0
Hard-decision Decoding	1	1	1	1	1	1	1
Proposed Algorithm	1.633	1.465	1.293	1.143	1.049	1.013	1.001
Chase Algorithm 3	4	4	4	4	4	4	4
Correlation Decoding	4096	4096	4096	4096	4096	4096	4096

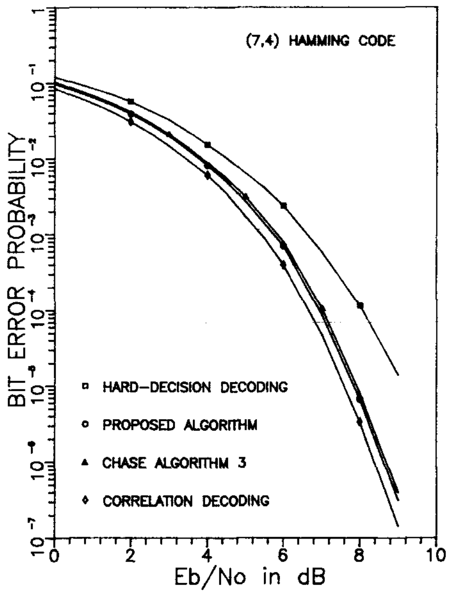


그림 3. (7, 4) Hamming 符號에 대한 E_b/N_0 對 비트 誤謬 確率圖

Fig. 3. Bit error probability versus E_b/N_0 for the (7, 4) Hamming code.

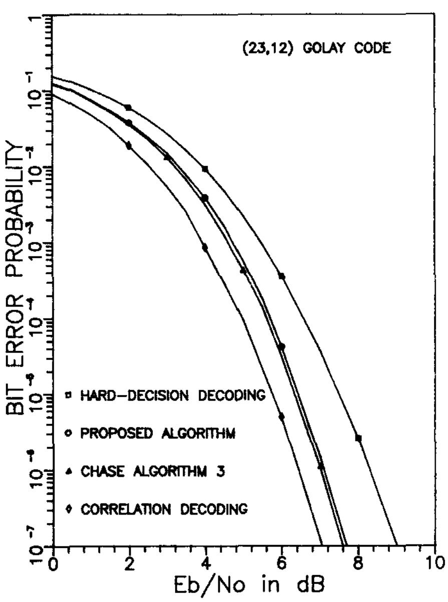


그림 4. (23, 12) Golay 符號에 대한 E_b/N_0 對 비트 誤謬 確率圖

Fig. 4. Bit error probability versus E_b/N_0 for the (23, 12) Golay code.

이다.

그림 3과 그림 4에 각각 (7, 4) Hamming 符號와 (23, 12) Golay 符號에 대하여 E_b/N_0 값의 變化에 따른 비트 誤謬 確率의 시뮬레이션 결과를 나타내었다 (7, 4) Hamming 符號의 경우 提案된 알고리듬의 비트 誤謬 確率은 Chase 알고리듬보다 낮다. (23, 12) Golay 符號의 경우 提案된 알고리듬의 비트 誤謬 確率은 Chase 알고리듬보다 높지만, 表 2에서 復號의 複雜度는 提案된 알고리듬이 가장 작음을 알 수 있다

Ⅳ. 結 論

本 論文에서는 새로운 軟判定 復號 알고리듬을 提案하여 復號의 複雜度를 減小 시켰으며, 시뮬레이션을 통하여 이를 확인하였다.

提案된 알고리듬은 모든 線形 2元 블럭 符號에 적용될 수 있다. 그리고 본 알고리듬의 수행에 필요한 硬判定 復號의 數는 항상 2回 以下이다. 따라서, 硬判定 復號 과정의 複雜度가 큰 符號일수록, 본 알고리듬을 적용함으로써 軟判定 復號의 複雜度 減少 효과를 더욱 크게 기대할 수 있다.

參 考 文 獻

[1] G.D. Forney, Jr., *Concatenated codes.* Cambridge, Mass.: M.I.T. Press, 1966, Ch.3.

[2] D. Chase, "A class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory,* vol. IT-18, pp. 170-182, Jan. 1972.

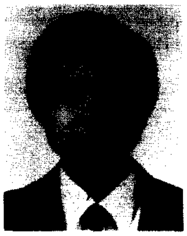
[3] B.G. Dorsch, "A decoding algorithm for binary block codes and J-ary output channels," *IEEE Trans. Inform. Theory,* vol. IT-20, pp. 391-394, May 1974.

[4] C.M. Hackett, "An efficient algorithm for soft-decision decoding of the (24,12) extended Golay code," *IEEE Trans. Commun.,* vol. COM-29, pp. 909-911, June 1981.

[5] H. Tanaka and K. Kakigahara, "Simplified correlation decoding by selecting possible codewords using erasure information," *IEEE Trans. Inform. Theory,* vol. IT-29, pp. 743-748, Sept. 1983.

[6] N.N. Tendolkar and C.R. P. Hartmann, "Generalization of Chase algorithms for soft-decision decoding of binary linear codes," *IEEE Trans. Inform. Theory,* vol. IT-30, pp. 714-721, Sept. 1984.

著 者 紹 介



沈 龍 杰(正會員)
1960年 1月 7日生. 1982年 2月 서울대학교 전자공학과 졸업. 1984年 2月 서울대학교 대학원 전자공학과 졸업(공학석사). 1984年～1986年 금성사 중앙연구소 연구원. 1984年 3月～현재 서울대학교 대학원 전자공학과 박사과정. 1988年 3月 ～ 현재 단국대학교 전자공학과 전임강사. 주관심분야는 符號理論, 情報理論, 디지탈通信 시스템 등임.

李 忠 雄(正會員)　第26卷 第5號 參照
현재 서울대학교 전자공학과 교수