

符號理論의 概念

길쌈符號篇

李 晚 榮
漢陽大學校 教授

本誌 2, 4月號에서 講述한 블럭符號(線形符號 및 巡回符號)에 이어 이번 號에서는 對合符號, 即 길쌈符號(convolutional code)에 對해 記述하고자 한다. 길쌈符號는 그 復號法 中 逐次 및 最尤復號에 있어 그 構造的 表現이 代數學的(algebraic)이라기 보다는 오히려 位相學的(topological)이라는 點에서 注目할만한 特徵을 지니고 있다.

길쌈符號를 應用한 例로는 1977年 NASA에서 實施한 Voyager 宇宙探查計劃을 들 수 있다. 火星(Mars), 木星(Jupiter) 및 土星(Saturn) 등에서 一連의 科學 data 蒐集을 目的으로 했던 이 計劃은 NASA所管 California工大 JPL研究所에서 開發한, 記憶素子가 6段인 (3, 1) 길쌈符號의 CODEC을 使用하였다.

I. 序 論

길쌈符號의 符號化 및 復號法은 블럭符號의 境遇와는 顯著한 差異가 있다. 블럭符號의 境遇에는 情報系列을 k비트씩 一定한 길이의 블럭單位로 잘라 똑같은 길이의 n비트 符號語로 單純 符號化하는 것에 反해 길쌈符號의 境遇에는 블럭單位로 符號化가 實行되기는 하지만 過去의 情報비트가 現在의 情報비트에 影響을 끼침으로써 길쌈符號의 符號器는 記憶(memory)이 있다고 한다. 따라서 길쌈符號는 블럭符號에 비해 多少 複雜한 構造를 가지므로 解析 上의 難點은 있으나 誤謬訂正能力은 매우 優秀하다.

길쌈符號의 表現方法에는 遲延素子에 依한 表現과 行列에 依한 表現, 그리고 格子狀符號(trellis code)에 依한 表現法 및 樹枝狀符號(tree code)에 依한 表現法 등이 있다.

길쌈符號의 復號法은 大別해서 臨界復號法(threshold decoding), 逐次復號法(sequential decoding) 그리고 最尤復號法(maximum likelihood decoding) 등이 있다. 臨界復號法에서는 遲延素子에 依한 表現과 行列에 依한 表現 등을 使用하고, 最尤復號法과 逐次復號法에서는 各其 格子狀圖에 依한 表現法과 樹枝狀圖에 依한 表現法을 使用한다.

한편, 實用的인 側面에서 볼 때 裝置化의 複雜度가 같은 水準이라면 길쌈符號가 블럭符號보다 誤謬訂正能力이 높다. 特히 通信分野의 應用에서 길쌈符號는 블

럭符號에 비해 越等히 有望하다고 評價되고 있다.

II. 符 號 化

拘束長(constraint length) n_A 인 길쌈符號器는 m段의 遲延素子(또는 記憶素子)로 構成되며 選定된 各 遲延素子들의 出力을 2元加算器에 入力시킴으로써 그 出力이 符號系列을 形成한다. 이 符號器는 情報비트를 符號化한 符號語를 直列形的 傳送系列로 連續 處理한다. 特히 길쌈符號의 生成行列은 無限長의 行과 列로 構成되므로 情報 및 符號系列은 任意의 大端히 긴 半無限長의 系列로 看做된다.

情報系列은 k비트씩 블럭으로 分離되며 各 블럭의 k비트는 直並列變換器(S/P multiplexer)에 依해서 並列化되어 任意의 時點 t에서 m段의 遲延素子에 入力된다. 그 다음 各 遲延素子의 出力 및 直並列 變換器의 出力이 n個의 2元加算器에 入力되며 이 n個의 出力은 並直列變換器(P/S multiplexer)에 依해서 直列化되어 符號系列로서 通路로 送出되는 것이다. 이 符號系列의 總集합을 (n, k)길쌈符號라 부른다.

一般的으로 k비트의 情報블럭과 n비트의 符號語는 $k < n$ 의 關係가 있으며 k와 n은 작은 整數이다. 한 個의 情報비트가 直接 影響을 줄 수 있는 블럭數는 $m+1$ 이며, 符號器의 出力端子가 n個이므로 길쌈符號의 拘束長은 $n_A = (m+1)n$ 로 定義된다. 그리고 符號化率(code rate), 一名 情報速度는 $R = k/n$ 로 表示한다.

記憶素子 \$m\$ 段인 \$(n, k)\$ 길쌈符號의 生成系列(generator sequence)은 다음과 같이 表現된다.

$$\bar{g}^{(j)} = (g_{i,0}^{(j)}, g_{i,1}^{(j)}, \dots, g_{i,m-1}^{(j)}, g_{i,m}^{(j)}) \quad (1)$$

여기서 添字 \$i, i=1, 2, \dots, k\$는 入力 그리고 \$j, j=1, 2, \dots, n\$는 出力을 表示한다. 情報系列이 \$\bar{d} = (100\dots)\$ 일 때 \$(m+1)\$개의 비트로 構成된 符號生成系列은 符號器의 \$n\$개의 임펄스應答(impulse response)으로 解析된다. 이 符號生成系列은 다음과 같이 \$m\$次 多項式으로 表現할 수 있다.

$$g_{i,0}^{(j)}(D) = g_{i,0}^{(j)} + g_{i,1}^{(j)}D + \dots + g_{i,m-1}^{(j)}D^{m-1} + g_{i,m}^{(j)}D^m \quad (2)$$

여기서 \$D\$는 單位時間의 遲延을 表示하는 遲延演算子(delay operator)이다. 並列化된 情報系列 \$\bar{d}^{(j)} = (d_0^{(j)}, d_1^{(j)}, \dots)\$가 한 비트씩 길쌈符號器에 入力되면 情報系列 \$\bar{d}^{(j)}\$와 符號生成系列 \$\bar{g}_i^{(j)}\$와의 離散對合(discrete convolution)을 取함으로써 直列化된 符號系列 \$\bar{c}^{(j)} = (c_0^{(j)}, c_1^{(j)}, c_2^{(j)}, \dots)\$를 얻게 된다. 即

$$\bar{c}^{(j)} = \sum_{i=1}^k \bar{d}^{(i)} * \bar{g}_i^{(j)}, \quad j=1, 2, \dots, n \quad (3)$$

(3)式을 一般的으로 符號化 方程式(encoding equation)이라 부르며 다음과 같은 式으로 表現한다.

$$c_\lambda^{(j)} = \sum_{\mu=0}^m \left[\sum_{i=1}^k d_{\lambda-\mu}^{(i)} g_{i,\mu}^{(j)} \right] \quad (4)$$

여기서 \$0 \le \mu < \lambda, 1 \le j \le n\$이다.

이와 같이 符號化되는 길쌈符號의 簡單한 例로서 \$R=1/2, m=2\$段인 \$(2, 1)\$ 길쌈符號의 符號器는 그림 1과 같다.

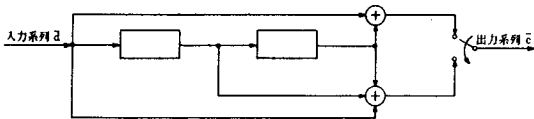


그림 1. 遲延素子 \$m=2\$段인 \$(2, 1)\$ 길쌈符號器

이 符號器는 \$m=2\$段的 記憶素子와 \$n=2\$個의 2元加算器 그리고 符號器 出力을 直列化하는 並直列變換器 등으로 構成되어 있다.

例題 1. (遲延素子에 依한 解法)

\$m=2\$段인 \$(2, 1)\$ 길쌈符號를 생각해 보자. 그림 1의 符號器를 利用하면 符號生成系列은 各各 \$\bar{g}_1^{(1)} = (g_{i,0}^{(1)}, g_{i,1}^{(1)}, g_{i,2}^{(1)}) = (101)\$과 \$\bar{g}_2^{(1)} = (g_{i,0}^{(2)}, g_{i,1}^{(2)}, g_{i,2}^{(2)}) = (111)\$ 이 된다. \$i=1, j=1, 2\$이므로 (4)式을 使用하면

$$c_\lambda^{(1)} = d_\lambda^{(1)} g_{1,0}^{(1)} + d_{\lambda-1}^{(1)} g_{1,1}^{(1)} + d_{\lambda-2}^{(1)} g_{1,2}^{(1)}$$

$$c_\lambda^{(2)} = d_\lambda^{(2)} g_{2,0}^{(2)} + d_{\lambda-1}^{(2)} g_{2,1}^{(2)} + d_{\lambda-2}^{(2)} g_{2,2}^{(2)}$$

등을 얻는데 \$\bar{g}_1^{(1)} = (101)\$과 \$\bar{g}_2^{(1)} = (111)\$을 위 式에 代入하면

$$c_\lambda^{(1)} = d_\lambda^{(1)} + d_{\lambda-1}^{(1)}$$

$$c_\lambda^{(2)} = d_\lambda^{(2)} + d_{\lambda-1}^{(2)} + d_{\lambda-2}^{(2)}$$

이 된다. 有限長 \$L=5\$인 情報系列을 \$\bar{d}^{(1)} = (10101)\$ 이라 할 때 그에 對應하는 符號系列은 \$0 \le \lambda \le 6\$에 對해서 다음과 같이 얻어진다.

\$\lambda\$	\$c_\lambda^{(1)}, j=1, 2\$	\$(c_\lambda^{(1)}, c_\lambda^{(2)})\$
0	\$c_0^{(1)} = d_0^{(1)} = 1, c_0^{(2)} = d_0^{(2)} = 1\$	(1, 1)
1	\$c_1^{(1)} = d_1^{(1)} = 0, c_1^{(2)} = d_1^{(1)} + d_0^{(2)} = 0 + 1 = 1\$	(0, 1)
2	\$c_2^{(1)} = d_2^{(1)} + d_1^{(1)} = 1 + 1 = 0, c_2^{(2)} = d_2^{(1)} + d_1^{(2)} + d_0^{(2)} = 1 + 0 + 1 = 0\$	(0, 0)
3	\$c_3^{(1)} = d_3^{(1)} + d_2^{(1)} = 0 + 0 = 0, c_3^{(2)} = d_3^{(1)} + d_2^{(2)} + d_1^{(2)} = 0 + 1 + 0 = 1\$	(0, 1)
4	\$c_4^{(1)} = d_4^{(1)} + d_3^{(1)} = 1 + 1 = 0, c_4^{(2)} = d_4^{(1)} + d_3^{(2)} + d_2^{(2)} = 1 + 0 + 1 = 0\$	(0, 0)
5	\$c_5^{(1)} = d_5^{(1)} = 0, c_5^{(2)} = d_5^{(1)} + d_4^{(2)} = 1 + 0 = 1\$	(0, 1)
6	\$c_6^{(1)} = d_6^{(1)} = 1, c_6^{(2)} = d_6^{(1)} = 1\$	(1, 1)

이와 같이 해서 並列化된 2個의 出力디지트는 並直列變換器에 依해 \$(c_\lambda^{(1)}, c_\lambda^{(2)})\$로 直列化되어 다음과 같은 單一系列(single sequence)의 符號語가 形成된다. 即

$$\bar{c} = (c_0^{(1)} c_0^{(2)}, c_1^{(1)} c_1^{(2)}, \dots, c_6^{(1)} c_6^{(2)}) = (11, 01, 00, 01, 00, 01, 11)$$

이것이 通信路에 送出되는 傳送系列이 된다.

例題 2. (離散對合에 依한 解法)

本 例에서도 그림 1의 길쌈符號器를 利用하여 똑같이 주어진 符號生成系列 \$\bar{g}_1^{(1)} = (101)\$과 \$\bar{g}_2^{(1)} = (111)\$ 그리고 情報入力系列 \$\bar{d}^{(1)} = (10101)\$을 使用함으로써 出力系列을 다음의 離散對合方法에 依해 求해 보자.

$$\bar{c}^{(1)} = \bar{d}^{(1)} * \bar{g}_1^{(1)} = (10101) * (101) = (1000001)$$

$$\bar{c}^{(2)} = \bar{d}^{(1)} * \bar{g}_2^{(1)} = (10101) * (111) = (1101011)$$

여기서 \$*\$는 對合(convolution)의 標記이다. \$\bar{c}^{(1)}, \bar{c}^{(2)}\$를 並直列變換器에 依해 直列系列化하면 다음의 符號語를 얻는다.

$$\bar{c} = (11, 01, 00, 01, 00, 01, 11)$$

이 符號系列은 例題 1에서 얻은 結果와 同一하다.

記憶素子가 \$m\$段인 \$(n, k)\$ 길쌈符號를 생각해 볼 때 어떤 任意的 情報系列

$$\bar{d} = (\bar{d}_0, \bar{d}_1, \bar{d}_2, \dots) \quad (5)$$

여기서 \$\bar{d}_\lambda = (d_\lambda^{(1)}, d_\lambda^{(2)}, \dots, d_\lambda^{(k)}), \lambda=0, 1, 2, \dots\$

가 符號器에 入力되면 符號系列

$$\bar{c} = (\bar{c}_0, \bar{c}_1, \bar{c}_2, \dots) \quad (6)$$

여기서 \$\bar{c}_\lambda = (c_\lambda^{(1)}, c_\lambda^{(2)}, \dots, c_\lambda^{(n)}), \lambda=0, 1, 2, \dots\$

가 얻어지는데, 이 關係를 行列表現으로 써보면 다음과 같다.

$$\bar{c} = \bar{d} \cdot \bar{C} \quad (7)$$

여기서 \bar{C} 는 半無限大의 行列로서 길쌈符號의 生成行列(generator matrix)이라 부르며 그 構造는 다음과 같다.

$$\bar{C} = \begin{bmatrix} \bar{C}_0 & \bar{C}_1 & \bar{C}_2 & \cdots & \bar{C}_m & \bigcirc \\ & \bar{C}_0 & \bar{C}_1 & \cdots & \bar{C}_{m-1} & \bar{C}_m \\ & & \bar{C}_0 & \cdots & \bar{C}_{m-2} & \bar{C}_{m-1} & \bar{C}_m \\ \bigcirc & \vdots & & & \vdots & \vdots & \vdots \\ & & & & & \vdots & \vdots \end{bmatrix} \quad (8)$$

여기서 \bar{C} 內的 \bigcirc 는 모두 "0"을 表示한다. (8)式을 살펴보면 生成行列 \bar{C} 의 任意的 行은 바로 前行과 同一하나 n 列만큼 右側으로 移動시킨 것에 不過하다. (8)式 內的 各 從屬生成行列(sub-generator matrix) \bar{C}_λ , $\lambda=0, 1, 2, \dots, m$ 는 다음과 같은 $k \times n$ 行列이다.

$$\bar{C}_\lambda = \begin{bmatrix} g_{1,\lambda}^{(1)} & g_{1,\lambda}^{(2)} & \cdots & g_{1,\lambda}^{(n)} \\ g_{2,\lambda}^{(1)} & g_{2,\lambda}^{(2)} & \cdots & g_{2,\lambda}^{(n)} \\ \vdots & \vdots & & \vdots \\ g_{k,\lambda}^{(1)} & g_{k,\lambda}^{(2)} & \cdots & g_{k,\lambda}^{(n)} \end{bmatrix} \quad (9)$$

例題 3. (스칼라行列에 依한 解法)

그림 1에 圖示한 길쌈符號器를 생각하자. $g_1^{(1)} = (101)$ 와 $g_2^{(2)} = (111)$ 를 (9)式에 代入하여 \bar{C}_0 를 求하면 $\bar{C}_0 = (110111)$ 로 된다. 따라서 情報系列을 $\bar{d} = (10101)$ 라 하면 符號系列은 (7)式에 依해서

$$\begin{aligned} \bar{c} &= \bar{d} \cdot \bar{C} \\ &= (10101) \begin{bmatrix} 11 & 01 & 11 \\ & 11 & 01 & 11 \\ & & 11 & 01 & 11 \\ & & & 11 & 01 & 11 \\ & & & & 11 & 01 & 11 \\ & & & & & 11 & 01 & 11 \end{bmatrix} \\ &= (11, 01, 00, 01, 00, 01, 11) \end{aligned}$$

와 같이 되어 앞의 例題 1, 2의 結果와 同一하다. 生成行列 \bar{C} 를 살펴보면 各行 ($k=1$)은 바로 前行과 同一하나 $n=2$ 列만큼 右側으로 移動했음을 알 수 있다.

길쌈符號器에서 並列化된 i 番째 情報系列 $\bar{d}^{(i)} = (d_1^{(i)}, d_2^{(i)}, \dots)$, $i=1, 2, \dots, k$ 를

$$d^{(i)}(D) = d_1^{(i)} + d_2^{(i)}D + d_3^{(i)}D^2 + \cdots \quad (10)$$

와 같이 遲延演算子 D 의 多項式으로 表現하면, 直列化되기 前의 j 番째 符號系列 $\bar{c}^{(j)} = (c_1^{(j)}, c_2^{(j)}, c_3^{(j)}, \dots)$, $j=1, 2, \dots, n$ 는

$$c^{(j)}(D) = c_1^{(j)} + c_2^{(j)}D + c_3^{(j)}D^2 + \cdots \quad (11)$$

와 같은 多項式表現이 된다. 그런데 $i=1, 2, \dots, k$; $j=1, 2, \dots, n$ 에 對해서

$$g_i^{(j)}(D) = g_{i,0}^{(j)} + g_{i,1}^{(j)}D + \cdots + g_{i,m}^{(j)}D^m \quad (12)$$

라 놓으면 j 番째 符號系列 $c^{(j)}(D)$ 는

$$c^{(j)}(D) = \sum_{i=1}^k d^{(i)}(D) g_i^{(j)}(D), \quad j=1, 2, 3, \dots, n \quad (13)$$

가 된다. 여기서 $g_i^{(j)}(D)$ 를 길쌈符號의 生成多項式(generator polynomial)이라 부르는데 이것은 入力 i 와 出力 j 에 關聯된 符號器의 傳達函數로 생각할 수 있다. k 入力, n 出力의 符號化 回路에서 $k \times n$ 生成多項式行列(generator polynomial matrix) $\bar{C}(D)$ 를 形成할 수 있는데 이 $\bar{C}(D)$ 의 要素는 모두 nk 個의 從屬生成多項式(sub-generator polynomial)으로 이루어져 있다. 即

$$\bar{C}(D) = \begin{bmatrix} g_1^{(1)}(D) & g_1^{(2)}(D) & \cdots & g_1^{(n)}(D) \\ g_2^{(1)}(D) & g_2^{(2)}(D) & \cdots & g_2^{(n)}(D) \\ \vdots & \vdots & & \vdots \\ g_k^{(1)}(D) & g_k^{(2)}(D) & \cdots & g_k^{(n)}(D) \end{bmatrix} \quad (14)$$

여기서 $g_i^{(j)}(D)$ 는 (12)式으로 表現되는 生成多項式이다. 記憶素子가 m 段인 (n, k) 길쌈符號의 符號化 方程式은

$$c^{(j)}(D) = \sum_{i=1}^k d^{(i)}(D) g_i^{(j)}(D), \quad j=1, 2, \dots, n \quad (15)$$

로 表示되므로 (14)式과 (15)式으로부터

$$\bar{c}'(D) = \bar{d}(D) \bar{C}(D) \quad (16)$$

와 같이 簡單한 表現으로 쓸 수 있다. 여기서 $\bar{d}(D) = [d^{(1)}(D), d^{(2)}(D), \dots, d^{(k)}(D)]$ 는 情報系列多項式이며 $\bar{c}'(D) = [c^{(1)}(D), c^{(2)}(D), \dots, c^{(n)}(D)]$ 는 符號系列多項式이다. 따라서 並直列變換器를 거쳐 直列化된 符號語는

$$c(D) = c^{(1)}(D^n) + D c^{(2)}(D^n) + \cdots + D^{n-1} c^{(n)}(D^n) \quad (17)$$

가 된다.

例題 4. (多項式行列에 依한 解法)

그림 1에 圖示한 遲延素子 $m=2$ 段인 $(2, 1)$ 길쌈符號를 다시 생각해 보자. 이 符號의 生成多項式行列은

$$\bar{C}(D) = [1+D^2, 1+D+D^2]$$

으로 表示되는 1×2 行列이다. 따라서 情報系列 $\bar{d}(D) = (1+D^2+D^4)$ 에 對한 符號化 方程式은

$$\begin{aligned} \bar{c}'(D) &= \bar{d}(D) \bar{C}(D) \\ &= (1+D^2+D^4) [1+D^2, 1+D+D^2] \\ &= [1+D^6, 1+D+D^3+D^5+D^6] \\ &= [c^{(1)}(D), c^{(2)}(D)] \end{aligned}$$

이 된다. 그러므로 直列化된 符號語는

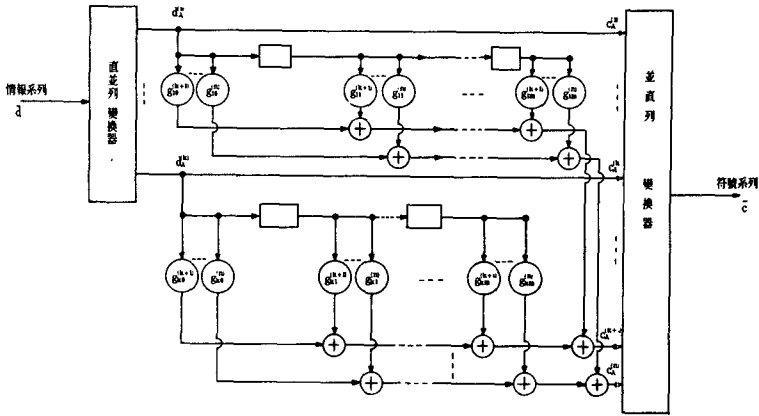


그림 3. TYPE II 組織形 길쌈符號器

態가 된다.

$$\bar{C}(D) = \begin{bmatrix} 1 & 0 & \dots & 0 & g_1^{(k+1)}(D) & g_1^{(k+2)}(D) & \dots & g_1^{(n)}(D) \\ 0 & 1 & \dots & 0 & g_2^{(k+1)}(D) & g_2^{(k+2)}(D) & \dots & g_2^{(n)}(D) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & g_k^{(k+1)}(D) & g_k^{(k+2)}(D) & \dots & g_k^{(n)}(D) \end{bmatrix} \quad (24)$$

$\bar{C}(D)$ 는 組織構造를 가지므로 (24)式에 對應하는 $(n-k) \times n$ 檢査多項式行列 $H(D)$ 는

$$H(D) = \begin{bmatrix} g_1^{(k+1)}(D) & \dots & g_k^{(k+1)}(D) & 1 & 0 & \dots & 0 \\ g_1^{(k+2)}(D) & \dots & g_k^{(k+2)}(D) & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ g_1^{(n)}(D) & \dots & g_k^{(n)}(D) & 0 & 0 & \dots & 1 \end{bmatrix} \quad (25)$$

와 같이 表現된다. 萬若 $\bar{C}(D)$ 의 行空間 內의 어느 系列이든지 $H(D)$ 의 行에 直交하면 $\bar{C}(D)H^T(D) = 0$ 이고 $\bar{c}'(D)$ 가 符號系列多項式인 境遇에 限하여

$$\bar{c}'(D)H^T(D) = 0 \quad (26)$$

의 關係가 成立된다.

다음에는 그림 4에 圖示한 記憶素子 $m=2$ 段인 (2,1) 組織形 길쌈符號의 符號化 問題에 對해 解答을 求해 보자.

例題 5. (스칼라行列을 利用한 解法)

그림 4의 組織符號器에서 生成行列 \bar{C} 의 第 1 行은

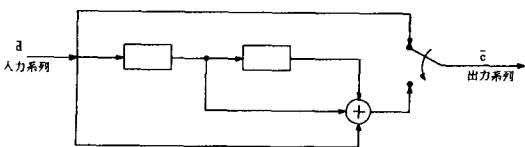


그림 4. $m=2$ 段인 (2,1) 組織形 길쌈符號

$\bar{g}_1^{(1)} = (100)$ 과 $\bar{g}_2^{(1)} = (111)$ 를 交錯(interlace) 시킴으로써 $\bar{C}_0 = (110101)$ 가 얻어진다. 情報系列이 $\bar{d} = (10101)$ 일 때 符號系列 \bar{c} 는

$$\begin{aligned} \bar{c} &= \bar{d} \cdot \bar{C} \\ &= (10101) \begin{bmatrix} 11 & 01 & 01 \\ & 11 & 01 & 01 \\ & & 11 & 01 & 01 \\ & & & 11 & 01 & 01 \end{bmatrix} \\ &= (11, 01, 10, 01, 10, 01, 01) \end{aligned}$$

와 같이 된다. 情報系列은 화살표가 指示하는 것과 같이 (10101)이며 檢査行列은 (1101011)이다.

그림 4의 符號化 問題에 對해 例題 1, 2, 4와 같이 遲延素子에 依한 解法, 離散對合에 依한 解法 그리고 多項式行列에 依한 解法等으로 讀者는 各各 물어 例題 5의 結果와 同一함을 確認해 주기 바란다.

IV. 非組織길쌈符號

非組織길쌈符號는 d_{min} 이 組織길쌈符號보다 크므로 符號의 效率性이 좋다. 그러나 하드웨어 構成이 複雜해지고 또한 符號語로부터 情報系列만을 復元하기爲한 變換回路가 따로 必要하게 된다.

이제 非組織길쌈符號에 對해 알아보자. (7)式에 (5), (6), (8), (9)式을 代入하면 符號語 디지트는

$$c_\lambda^{(j)} = \sum_{i=1}^k d_\lambda^{(i)} g_{i,0}^{(j)} + \sum_{i=1}^k d_{\lambda-1}^{(i)} g_{i,1}^{(j)} + \dots + \sum_{i=1}^k d_{\lambda-m}^{(i)} g_{i,m}^{(j)}$$

와 같이 表現되므로 $c_\lambda^{(j)}$ 를 다음 두 가지 形態로 表示할 수 있다.

$$c_\lambda^{(j)} = \sum_{\mu=0}^m \sum_{i=1}^k d_{\lambda-\mu}^{(i)} g_{i,\mu}^{(j)} \quad j=1, 2, \dots, n \quad (27)$$

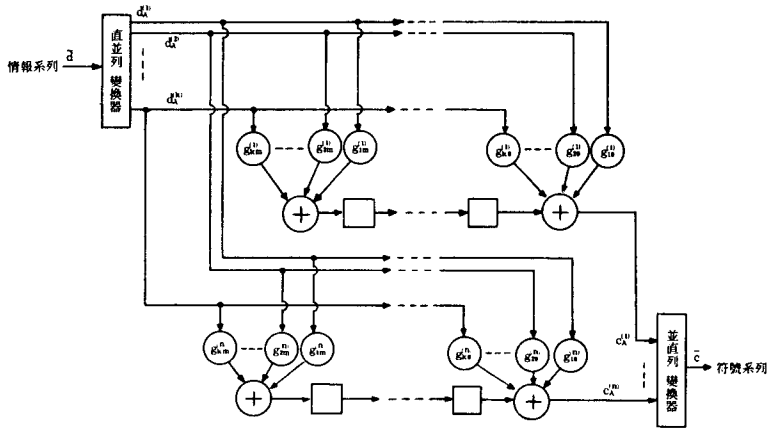


그림 5. TYPE I 非組織길쌈符號器

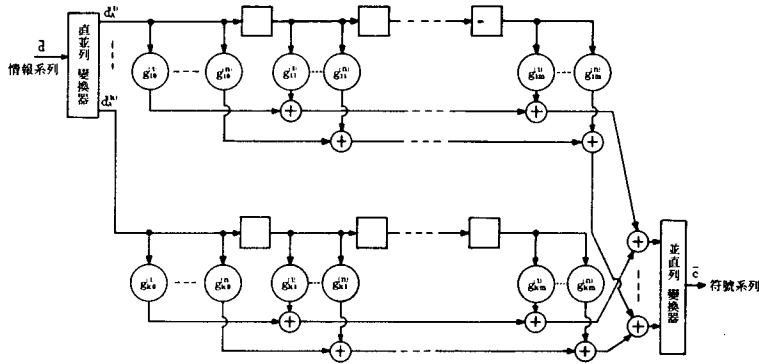


그림 6. TYPE II 非組織길쌈符號器

또는
$$c_j^{(j)} = \sum_{i=1}^k \sum_{\mu=0}^m d_{i-\mu}^{(i)} g_{i,\mu}^{(j)} \quad j=1, 2, \dots, n \quad (28)$$

(27)식을 토대로 한 TYPE I 非組織길쌈符號器를 그림 5에, (28)식을 토대로 한 TYPE II 非組織길쌈符號器를 그림 6에 각각 나타냈다.

以上에서 說明한 非組織길쌈符號의 符號化問題를 그림 7의 符號器를 例로 삼아 풀어보자.

例題 6. (生成多項式行列에 依한 解法)

그림 7은 $m=2$ 段인 (3, 2) 길쌈符號器이다. $k=2$ 이므로 2層(two level)의 遲延素子로 이루어지고 또 3개의 2元加算器와 3개의 出力 端子를 갖는다. 이 符號器의 生成系列은

$$\begin{aligned} \vec{g}_1^{(1)} &= (100) & \vec{g}_1^{(2)} &= (010) & \vec{g}_1^{(3)} &= (110) \\ \vec{g}_2^{(1)} &= (001) & \vec{g}_2^{(2)} &= (100) & \vec{g}_2^{(3)} &= (111) \end{aligned}$$

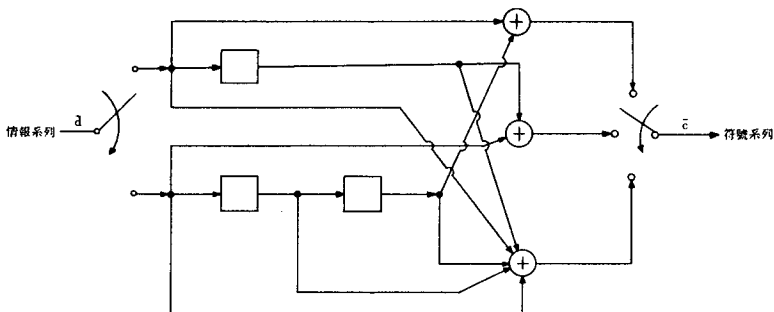


그림 7. $m=2$ 段인 (3, 2) 非組織길쌈符號器

이므로 生成多項式行列은 다음과 같은 2×3 行列로 表現된다.

$$\overline{c}(D) = \begin{bmatrix} 1 & D & 1+D \\ D^2 & 1 & 1+D+D^2 \end{bmatrix}$$

情報系列이 $\overline{d}^{(1)} = (101)$, $\overline{d}^{(2)} = (010)$ 即 $d^{(1)}(D) = 1 + D^2$, $d^{(2)}(D) = D$ 일 때 (16)式을 使用하면

$$\begin{aligned} \overline{c}'(D) &= [1+D^2, D] \begin{bmatrix} 1 & D & 1+D \\ D^2 & 1 & 1+D+D^2 \end{bmatrix} \\ &= [1+D^2+D^3, D^3, 1] \\ &= [c^{(1)}(D) c^{(2)}(D) c^{(3)}(D)] \end{aligned}$$

이 된다. 따라서 符號化方程式은 (17)式에 따라

$$\begin{aligned} c(D) &= c^{(1)}(D^2) + Dc^{(2)}(D^2) + D^2c^{(3)}(D^2) \\ &= 1+D^2+D^6+D^4+D^0 \end{aligned}$$

가 되므로 傳送될 符號系列은

$$\overline{c} = (101, 000, 100, 110, 000)$$

이 된다.

그림 7의 符號器에 對해 遲延素子에 依한 解法, 離散對合에 依한 解法 그리고 스칼라 生成行列에 依한 解法 등으로 各各 풀어 例題 6의 解答과 同一한 지 與否를 比較 檢討해 보기 바란다.

一般的으로 情報系列에 對應하는 符號系列 $c(D)$ 는

$$c(D) = \sum_{i=1}^k d^{(i)}(D^n) g_i(D) \quad (29)$$

와 같이 表現되며, 여기서

$$g_i(D) = g_i^{(1)}(D^n) + Dg_i^{(2)}(D^n) + \dots + D^{n-1}g_i^{(n)}(D^n) \quad (30)$$

이다. 이제 이 公式를 利用하여 符號化해 보기로 하자.

例題 7. (生成多項式的 遲延演算子 表現에 依한 解法)

그림 7의 符號器에서

$$\begin{aligned} g_1^{(1)}(D) &= 1 & g_1^{(2)}(D) &= D & g_1^{(3)}(D) &= 1+D \\ g_2^{(1)}(D) &= D^2 & g_2^{(2)}(D) &= 1 & g_2^{(3)}(D) &= 1+D+D^2 \end{aligned}$$

이므로 (30)式에 依해

$$\begin{aligned} g_1(D) &= 1+D(D^2)+D^2(1+D^2) = 1+D^2+D^4+D^6 \\ g_2(D) &= D^6+D+D^2(1+D^2+D^6) = D+D^2+D^5+D^6+D^8 \end{aligned}$$

을 얻을 수 있다. 人力系列을 $d^{(1)}(D) = 1+D^2$, $d^{(2)}(D) = D$ 라 하면 (29)式에 依해 符號多項式은 다음과 같이 된다.

$$\begin{aligned} c(D) &= (1+D^2)(1+D^2+D^4+D^6) + D^2(D+D^2+D^5 \\ &\quad + D^6+D^8) \\ &= 1+D^2+D^6+D^8+D^{10} \end{aligned}$$

따라서 이에 對應하는 符號系列은

$$\overline{c} = (101, 000, 100, 110, 000)$$

와 같이 되어 例題 6의 結果와 同一함을 알 수 있다.

V. 構造的 特性

길쌈符號는 그 構造的 表現이 代數學的(algebraic) 이라기 보다는 오히려 位相學的(topological) 이라는 點에서 注目할 만한 特徵을 지니고 있다. 慣例的으로, 길쌈符號의 構造的 表現을 다루는 데에 必須的인 것으로 狀態圖(state diagram), 樹枝狀表現圖(tree diagram) 및 格子狀表現圖(trellis diagram) 등이 있다. 길쌈符號의 拘束長이 길어지면 符號器 內的 狀態數가 指數關數的으로 增加하므로 狀態圖가 複雜해진다. 따라서 拘束長이 긴 길쌈符號는 Fano 알고리즘에 立脚한 逐次復號法(sequential decoding)이 有效하며 Viterbi 알고리즘에 基礎를 둔 最尤復號法(maximum likelihood decoding)에는 不適當하다. 狀態圖와 格子狀表現圖는 拘束長이 짧은 길쌈符號를 解析하는 데에 매우 有益하게 使用된다. 最尤復號法은 確率의 復號라는 點에서 逐次復號法과 一致하며 또 그 源泉이 逐次復號法에서 派生된 것이므로 이 두 復號法은 類似하다.

(n, k) 길쌈符號器는 k 層의 置換레지스터를 가지며 또 各層의 置換레지스터는 γ_i 段의 遲延素子로 構成된다. 따라서 符號器 內에 包含된 遲延素子의 總數는 $\gamma = \gamma_1 + \gamma_2 + \dots + \gamma_k$ 이며 여기서 γ_i 는 i 層에 存在하는 遲延素子의 數를 表示한다. $k=1$ 일 때는 $\gamma=m$ 即 遲延素子의 個數는 m 이 된다.

時刻 $t=\lambda$ 일 때 符號器의 並列人力을 $d_{\lambda}^{(1)}, d_{\lambda}^{(2)}, \dots, d_{\lambda}^{(k)}$ 라 하면 그 瞬間의 符號器狀態(encoder state) 는 다음과 같은 γ 次元벡터(γ -tuple vector)로 構成된 集合으로 表示된다.

$$\begin{aligned} \overline{s} &= (d_{\lambda-1}^{(1)}, d_{\lambda-2}^{(1)}, \dots, d_{\lambda-1}^{(1)}\gamma_1, d_{\lambda-1}^{(2)}, d_{\lambda-2}^{(2)}, \dots, d_{\lambda-1}^{(2)}\gamma_2, \\ &\quad \dots, d_{\lambda-1}^{(k)}, d_{\lambda-2}^{(k)}, \dots, d_{\lambda-1}^{(k)}\gamma_k) \end{aligned} \quad (31)$$

여기서 集合 \overline{s} 의 各元素는 各並列人力의 γ 個의 情報디지트임을 알 수 있다. 따라서 서로 다른 狀態의 可能的 總數는 2^γ 個이다.

符號器의 狀態들을 $s_0, s_1, \dots, s_{2^\gamma-1}$ 등으로 表示할 때 그 中 任意的 狀態 s_σ 는 $(a_0, a_1, \dots, a_{\gamma-1})$ 와 같이 γ 個의 2元記號 0, 1로 表示된다. 그리고 添字 σ 는 $\sigma = a_0 + a_1 2 + \dots + a_{\gamma-1} 2^{\gamma-1}$ 에 依해서 決定된다. 例를 들면 $m=2$ 段인 $(n, 1)$ 길쌈符號에서는 $\gamma = \gamma_1 = 2$ 이므로 $t=\lambda$ 일 때의 符號器狀態는 $(d_{\lambda-1}, d_{\lambda-2})$ 와 같이 된다. 그런데 添字 $\sigma = a_0 + 2a_1$ 에서 $a=0, 2, 1, 3$ 이 決定되므로 狀態의 集合은 $\{s_0, s_2, s_1, s_3\}$ 이며 이것에 對應하는 것은 $\{00, 01, 10, 11\}$ 이 된다.

m 段의 單一置換레지스터를 가진 $(n, 1)$ 길쌈符號에서 $t=\lambda$ 일 때의 狀態 s_λ 는 2^m 個의 狀態 中의 하나이다.

그러므로 狀態空間(state space)은

$$\zeta = \{s_0, s_1, s_2, \dots, s_{2^m-1}\} \quad (32)$$

가 된다. 時間帶 $0 \leq t \leq L+m$ 內에서 對合過程(convolutional process)이 執行되고, 또 初期 및 最終狀態를 各各 s_0 및 s_{L+m} 이라 假定할 때 狀態系列은 狀態 벡터 $\bar{s} = (s_0, s_1, \dots, s_{L+m})$ 로 表示된다. 여기서 L 은 情報入系列의 有限長이다. $t = \lambda$ 까지의 모든 狀態를 알고 있다는 條件下에서 $t = \lambda+1$ 에서의 遷移確率(transition probability)은 다음과 같은 條件附 確率로 表現되지만 狀態遷移의 現象은 Markov過程(Markov process)이므로 $P(s_{\lambda+1})$ 은 $t = \lambda$ 때의 狀態 s_λ 에 依해서만 影響을 받게 된다. 따라서

$$P(s_{\lambda+1}/s_0, s_1, \dots, s_\lambda) = P(s_{\lambda+1}/s_\lambda) \quad (33)$$

이다. 萬一 $t = \lambda$ 때의 遷移 x_λ 를 狀態雙($s_{\lambda+1}, s_\lambda$)로 定義한다면 可能的 모든 遷移의 數는 2^{2^m} 以下가 된다. 그리고 遷移行列을 $\bar{x} = (x_0, x_1, \dots, x_{L+m-1})$ 로 表現할 때 狀態系列 \bar{s} 와 遷移系列 \bar{x} 는 1對1 對應한다.

情報系列을 $\bar{d} = (d_0, d_1, \dots, d_\lambda, \dots)$ 라고 表示할 때 d_λ 는 確率密度分布 $P(d_\lambda)$ 에 依해서 獨立的으로 發生한다. 記憶素子가 m 段인 $(n, 1)$ 길쌈符號를 생각해 볼 때 入力디지트 d_λ 에 對한 m 段의 置換레지스터의 關係를 그림 8에 나타내 보면 時刻 $t = \lambda$ 때의 狀態는 $s_\lambda = (d_{\lambda-1}, d_{\lambda-2}, \dots, d_{\lambda-m})$ 와 같이 表現된다.

勿論 狀態數는 2^m 이며 $t = \lambda$ 때의 遷移는 $x_\lambda = (d_\lambda, d_{\lambda-1}, \dots, d_{\lambda-m})$ 와 같이 遷移數는 2^{m+1} 이다. 以上을 k 層(k -level)의 置換레지스터를 갖는 (n, k, m) 길쌈符號의 境遇로 擴張시키면 狀態의 數는 2^{γ} 이며 遷移의 數는 $2^{\gamma+k}$ 가 된다.

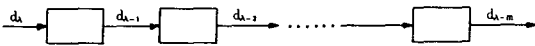


그림 8. 遲延素子와 入力디지트와의 關係

길쌈符號器는 一種의 有限狀態機械(finite-state machine)이므로 狀態圖만 가지면 그 作動內容을 充分히 觀察할 수 있다. 예를 들어 그림 1의 $m=2$ 段인 $R=1/2$ 길쌈符號器를 생각해 보자. 이 符號器의 狀態圖는 그림 9에서 보듯이 狀態의 數는 $2^m=4$ 이며 入力은 遷移를 일으키게 한다. 各 狀態에서 나오는 枝路(branch)의 數는 2個인데, 하나는 入力 "0"에 對應하는 것이며, 또 다른 하나는 入力 "1"에 對應하는 것이다. 또 各 枝路는 入力과 狀態의 遷移에 따른 出力

을 함께 表記한다. 예를 들면 狀態 $s_0=00$ 에서 나오는 두 個의 枝路는 各各 0/00과 1/11과 같이 表示한다. 狀態圖를 그리기 爲해서 入力, 出力 및 遷移狀態 등을 記錄한 狀態表가 必要하게 된다. 그림 1의 길쌈符號器에 對한 狀態表와 狀態圖는 各各 표 1, 그림 9와 같다.

표 1. (2, 1, 2) 길쌈符號의 狀態表

入力 狀態	出力 / 遷移狀態	
	0	1
0 0	0 0 / 0 0	1 1 / 1 0
0 1	1 1 / 0 0	0 0 / 1 0
1 1	1 0 / 0 1	0 1 / 1 1
1 0	0 1 / 0 1	1 0 / 1 1

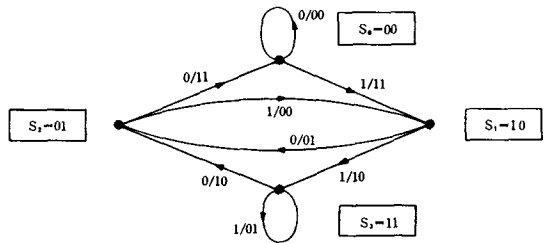


그림 9. $m=2$ 段인 (2, 1) 길쌈符號器의 狀態圖

全零狀態 $s_0 = (00)$ 을 始點으로 假想할 때 주어진 情報系列 \bar{d} 에 對應하는 符號系列 \bar{c} 는 그 情報系列에 對應되는 經路(path)를 거쳐 結局 全零狀態 $s_0 = (00)$ 로 다시 되돌아 오므로써 求해진다. 예를 들면 그림 9의 狀態圖에서 情報系列이 $\bar{d} = (10101)$ 일 때 符號系列은 $\bar{c} = (11, 01, 00, 01, 00, 01, 11)$ 이 된다.

狀態圖, 樹枝狀圖 및 格子狀圖中 어느 것을 擇해 길쌈符號器를 表現해도 符號의 構造의 特性을 觀察할 수 있다. 그림 9의 狀態圖를 土台로 樹枝狀表現圖(tree diagram)를 그리면 그림 10과 같다. 그림 10과 같이 樹枝狀表示에 依해 定해지는 길쌈符號를 一名 樹枝狀符號라 하기도 한다.

樹枝狀圖上에서 最左端에 位置한 節點(node)을 原節點(origin node)이라 부르며 樹枝狀 符號器의 始發點을 뜻한다. 一般的으로 kL 비트의 情報系列을 入力으로 하고 記憶素子가 m 段인 (n, k) 길쌈符號器의 樹枝狀圖는 左로부터 右로 $L+m+1$ 層(level)으로 分割되는데 그 中 L 層까지의 各 節點에서 派生되는 枝路는 2^k 個이다. 예를 들면 $L=5, m=2$ 段인 (2, 1) 樹枝狀符號의 境遇에는 層의 總數가 8이며 그 中 層 4가

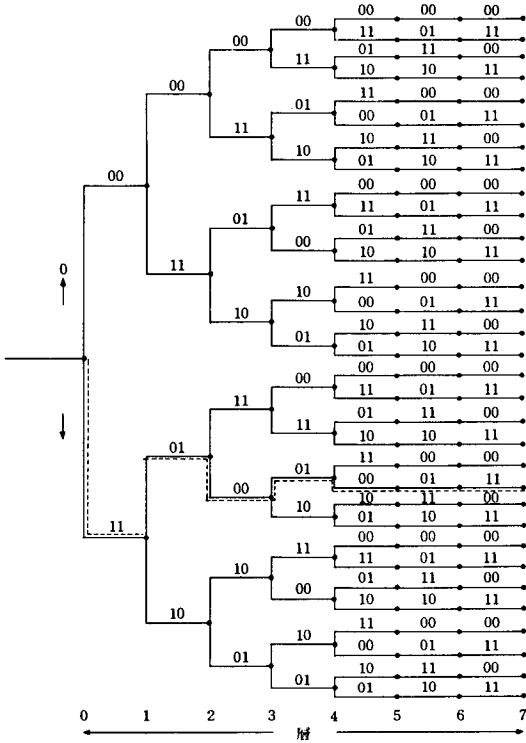


그림10. $m=2$ 段인 $(2, 1)$ 길쌈符號의 樹枝狀表現圖

지의 各層에 屬하는 節點에서 派生되는 枝路는 2個씩이다. 이 樹枝狀 構造의 節點 上에서 枝路의 方向은 情報비트의 選擇에 對應하며, 위로 向하는 枝路는 "0"에, 아래로 向하는 枝路는 "1"에 該當한다. 學者에 따라 "0"과 "1"을 바꾸어 規定하는 境遇도 있다. 그림 1의 $(2, 1, 2)$ 길쌈符號의 境遇에는 入力 情報系列 $\bar{d} = (10101)$ 에 對應하는 枝路의 選擇은 層 0에서는 아래, 層 1에서는 위, 層 2에서는 아래, 層 3에서는 위, 그리고 層 4에서는 아래의 順序이며 이렇게 하여 符號系列은 $\bar{c} = (11, 01, 00, 01, 00, 01, 11)$ 이 된다. 이런 方法으로 任意의 入力系列의 첫 5비트에 對應하는 모든 出力系列을 求하면 그림10에서 보듯이 32個 系列中의 하나에 該當된다. 樹枝狀圖를 살펴보면 左로부터 3番 枝路를 지난 後부터는 構造가 反復됨을 알 수 있다. 4番 枝路의 情報비트가 符號器에 入力되면 첫 番 枝路의 情報비트는 右端으로부터 빠져나감으로 出力 符號系列에는 전혀 影響을 주지 못한다. 따라서 3番 枝路 後부터는 나머지 情報系列의 影響을 받아 符號系列이 生成된다. 이러한 特徵은 樹枝狀圖를 그림11과 같은 格子狀圖(trellis diagram)로 變形할 수 있으며, 또

한 格子狀圖 上의 反復構造는 그림 9와 같은 狀態圖로 縮小할 수 있다. 結局 이 세 가지 表現圖는 서로 비슷하며 같은 特性을 지니고 있다.

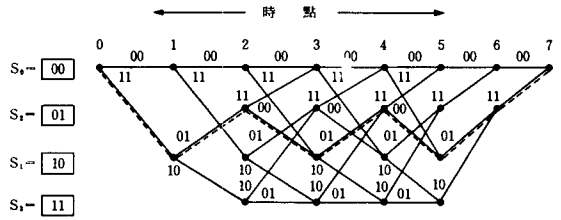


그림11. $m=2$ 段인 $(2, 1)$ 길쌈符號의 格子狀圖

格子狀圖는 $L+m+1$ 個의 時點(time unit) 또는 層(level)으로 區分된다. 一般의 記憶素子가 m 段인 (n, k) 길쌈符號器에 길이 kL 비트의 情報系列이 入力 되었을 때 出力은 그에 對應하는 2^{kL} 個의 符號系列이 된다. 그리고 格子狀圖에 出力하는 枝路는 2^k 個이며 또 格子狀圖를 거쳐나가는 서로 다른 經路(path)는 2^{kL} 個이다. 이 2^{kL} 個의 經路는 2^{kL} 個의 符號語와 같다. 예를 들어 그림11을 살펴보면 $L=5$ 로 假定할 때 時點은 0부터 7까지 區分이 된다. 符號器의 格子狀 表現은 恒常 初期狀態 $s_0 = (00)$ 부터 始作하여 7番 移行해 가면서 結局 $s_0 = (00)$ 로 되돌아오도록 되어 있다. 첫 部分의 두 時點은 s_0 로부터의 始發을 뜻하며 끝 部分의 두 時點은 s_0 로의 歸還을 뜻하며 이 時點 區間에서는 符號器의 모든 狀態를 包含하지는 않는다. 그러나 格子狀圖의 中心部分의 各 時點에 있어서는 $s_0 = (00)$, $s_2 = (01)$, $s_1 = (10)$, $s_3 = (11)$ 의 모든 狀態를 包含한다. 그리고 各 狀態로부터 나가는 枝路는 2個 뿐이다. 時點 i 에서 各 狀態로부터 위로 나가는 枝路는 入力이 $d_i = 0$ 임을 表示하며 아래로 나가는 枝路는 入力이 $d_i = 1$ 임을 意味한다. 各 枝路에는 入力 d_i 에 對應되는 2個의 出力 \bar{c}_i 를 表記한다. 符號長 14인 2^2

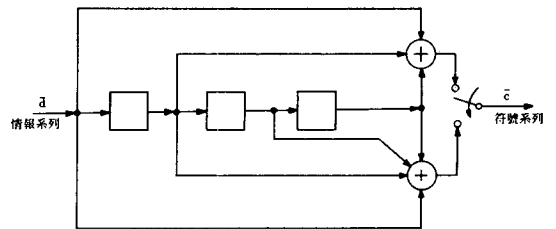


그림12. $m=3$ 段인 $(2, 1)$ 길쌈符號器

=32個 符號語의 各 符號系列은 格子狀圖에서 唯一한 經路와 對應한다. 따라서 符號器 出力은 格子狀圖를 거쳐 通過하는 適切한 經路를 追跡함으로써 求해진다. 人力 情報系列 $\bar{d}=(10101)$ 에 對應하는 出力 符號系列 $\bar{c}=(11, 01, 00, 01, 00, 01, 11)$ 는 點線으로 表示한 바와 같이 格子狀圖 上에서 容易하게 追跡된다.

例題 8. 그림12에 나타낸 $m=3$ 段인 $R=1/2$ 길쌈符號를 생각하자. 이 符號器의 生成多項式은 $g^{(1)}(D)=1+D+D^3$ 과 $g^{(2)}(D)=1+D+D^2$ 이다.

于先 狀態表를 作成한 後 狀態圖와 格子狀表現圖를 그려보자. 표 2는 人力에 따르는 出力과 遷移된 狀態를 收錄한 것이다.

표 2. 그림12에 圖示한 符號器의 狀態表

狀 態	人力	
	0	1
s_0	00 / 000	11 / 100
s_1	11 / 010	00 / 110
s_2	01 / 001	10 / 101
s_3	10 / 011	01 / 111
s_4	00 / 000	00 / 100
s_5	00 / 010	11 / 110
s_6	10 / 001	01 / 101
s_7	01 / 011	10 / 111

표 2의 狀態表를 利用하여 $m=3$ 段인 $R=1/2$ 길쌈 符號器의 狀態圖를 그리면 그림13과 같다.

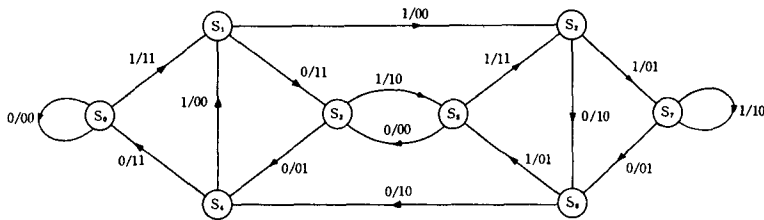


그림13. $m=3$ 段인 (2, 1) 길쌈符號器의 狀態圖

$L=5$ 일 때 遲延素子數가 $m=3$ 인 格子狀表現圖의 時點은 $L+m+1=9$ 로 分割된다. 그리고 $k=1$ 이므로 人力情報系列 $kL=5$ 디지트에 對應하는 出力 符號系列은 $2^{2L}=32$ 個로 派生된다. 따라서 格子狀圖 上에서 서로 다른 經路의 個數는 32個이고 遷移되는 枝路는 2個씩 이다. 따라서 이 길쌈符號器의 格子狀表現圖는 그림14와 같다.

VI. 復 號 法

1961년에 Wozencraft가 길쌈符號의 逐次復號法을 提案한 以來 1963년에 Fano가 逐次復號를 爲한 Fano 알고리즘을, 1966년에 Zigangirov와 1969년에 Jelinek이 各其 獨立의으로 所謂 積立(stack) 또는 ZJ 알고리즘을 發表하여 길쌈符號에 適用함으로써 逐次復號하는 데 많은 貢獻을 하였다.

逐次復號의 特徵은 樹枝狀表現圖 内部의 한 節點 깊숙히 進入하는 데에 必要한 計算數가 確率變數(random variable)라는 것이다. 다시 말해서 Fano 알고리즘으로 逐次復號를 하는 境遇 必要한 計算數는 通信路에서 雜音의 影響을 받은 受信系列 그리고 符號傳送系列 및 길쌈符號器의 構造에 依存하는 確率變數인 것이다. 따라서 特定한 符號系에 對해서 주어진 平均計算數를 求하는 것은 힘든 일이며 바라는 機能(performance)을 達成하기 爲해서는 그만큼 複雜해진다.

이와 같이 逐次復號의 機能은 拘束長에 拘礙받지 않는 利點은 있으나 最尤復號에 比해 떨어진다. 또 이 逐次技法에 依한 復號는 雜音干涉이 甚할 때 많은 量의 計算이 必要하게 된다는 短點을 지니고 있다. 따라서 復號時間의 遲延으로 말미암아 情報의 消失을 招來하는 境遇가 往往 생긴다.

1963년에 Massey는 길쌈符號에도 適用될 수 있는 臨界 或은 多數決論理라는 技法을 提案하였는데 이 技法의 機能은 逐次 또는 最尤復號法에 比해 若干 떨어지나 裝置化가 簡單하다는 長點이 있다. 길쌈符號의

構造가 位相學的 表現이라는 것은 앞에서 記述한 바 있지만 臨界復號의 境遇에는 代數學的 表現이 可能하다. 따라서 臨界復號는 概念的으로나 實際的으로나 巡回블럭符號에 類似하다. 即 誤症計算부터 始作하는 復號節次가 블럭符號의 境遇와 비슷하므로 Viterbi 復號 또는 逐次復號 같은 探索하는 特徵이 없어 概念的으로 볼 때 簡單하다.

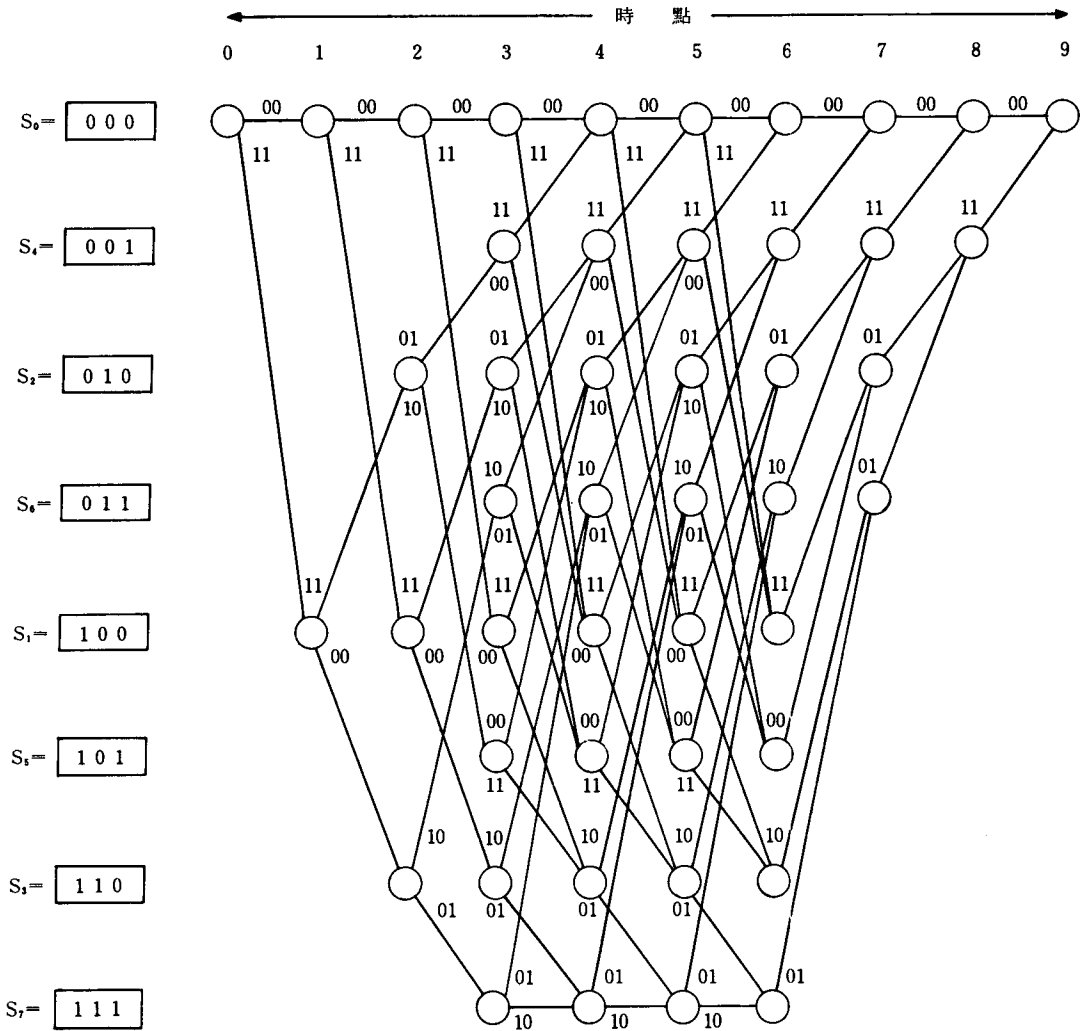


그림 14. $m=3$ 段인 $(2,1)$ 길쌈符號器의 格子狀表現圖

1967년에 Viterbi에 의해 提案된 Viterbi 알고리즘은 最尤復號하는 데에 매우 適合한 解法이다. 最尤復號法은 Viterbi 알고리즘을 使用하여 位相學的 構造를 갖는 格子狀圖 內의 最短經路를 決定하는 技法인데, 受信系列에 Hamming 距離가 가장 가까운 經路를 探索하는 復號法이기 때문에 可能性이 稀薄한 經路를 拋棄하면서 探索을 될 수 있는 대로 簡單하게 履行하는 方法이다. 그러나 誤謬確率은 拘束長 n_A 가 增加함에 따라 指數關數의 으로 增大하므로 Viterbi 復號는 $n_A > 8$ 인 긴 符號에 對해서는 非實用的이다. 다시 말해서 符號器의 記憶素子 $\gamma=m$ 의 制限때문에 不可缺 拘束長이 짧은 符號에만 適用된다. 그 理由는 拘束長 n_A 가 增加함에 따라 格子狀圖가 複雜해질 뿐만 아니라 復號 努

力이 그만큼 增大되어야 하기 때문이다. 그리고 Viterbi 알고리즘을 使用한 復號는 雜音의 影響을 極小化할 수 있지만 復號된 情報系列을 얻기까지는 2^7 의 計算數가 必要하므로 復號하는 데에 浪費가 많다.

以上에서 論한 길쌈符號의 復號技法을 大別하면 確率의 復號法과 代數的 復號法의 두 가지로 區分된다. 그런데 Viterbi 알고리즘에 依存한 最尤復號法과, Fano 알고리즘과 ZJ積立알고리즘에 立脚한 逐次復號法은 確率의 復號에 屬하며, Massey가 提案한 臨界復號法은 代數的 復號에 屬한다. 紙面關係上 길쌈符號의 各 復號法에 對한 詳細한 理論과 解析을 不得已 省略하게 됨을 大端히 遺憾으로 생각한다. 學會를 通해 뜻 있는 讀者의 要請이 있으면 執筆을 約束하면서 이 節

을 마치기로 한다.

Ⅶ. 講座를 마치면서

硬判定(hard decision) 및 軟判定(soft decision)의 길쌈符號 復號法은 近來에 宇宙 및 衛星通信에 널리 應用되고 있다. 1969년에 美國 通信衛星公社(Communications Satellite Cooperation)는 INTELSAT IV 通信衛星을 通過 33.6Mbps의 color TV 信號를 傳送하는 DITEC 시스템(digital television system)을 開發하였는데 이 시스템은 $m=146$ 인 (8, 7)自己直交길쌈符號에 多數決論理復號法을 適用하였다.

또 逐次復號法의 例로 1968년에 美航空宇宙局(NASA)의 Pioneer 9號 太陽系 軌道探査 計劃은 $m=20$ 段의 (2, 1)組織길쌈符號를 利用하였다. 1972년에 Pioneer 10號는 木星을, 1973년에 Pioneer 11號는 土星을 觀測 攝影하는 데 成功했는데 이 두 計劃은 모두 2Kbps의 data 傳送速度를 가진 $m=31$ 段의 (2, 1) 길쌈符號를 使用하였다. 또 1974年 美 California 工大의 JPL 研究所에서 遂行한 Pioneer 12號의 金星(Venus) 軌道旋回 實驗 計劃에는 筆者도 直接 參加한 바 있다.

그리고 最尤復號法의 例로서는 1977년에 NASA 가 火星, 木星 및 土星을 目標로 實施한 一連의 Voyager 宇宙探査 計劃에서 JPL이 開發한 $m=6$ 段인 (2, 1) 및 (3, 1) 길쌈符號用 CODEC (encoder/decoder)을 使用했는데 data 傳送速度는 100Kbps이며 이들 길쌈符號의 復號器에 3비트($Q=8$)의 軟量子化 判定을 試圖하였다. 이 中 $m=6$ 段인 (2, 1) 길쌈符號의 $Q=8$ 軟判

定 復號器는 現在도 美 防衛衛星通信網(DSCS)에서 使用中이다.

NASA의 JPL 研究所 以外에도 宇宙通信 및 衛星通信 分野에 陰으로 陽으로 貢獻한 會社들로 Linkabit社, Harris社, Codex社 및 其他 많은 會社와 研究所 등을 들 수 있다. 그 中 Linkabit社는 1970년에 50Mbps 硬判定 Fano 逐次復號器를 製作하여 現在 NASA에서 宇宙 station과 地上局과의 telemetry 交信에 使用하고 있는데 이 復號器는 가장 高速인 逐次復號器이다.

以上과 같이 宇宙 및 衛星通信 開拓에 中樞的인 役割을 한 것이 길쌈符號라는 事實은 通信專門家라면 누구도 否認하지 못하는 것이다. 그러면 우리 나라의 事情은 어떠한가? 宇宙産業이 全無狀態인 韓國에서는 이 遠大하고도 老大的 計劃 樹立을 엄두도 못내고 있다. 富強한 나라, 科學의 나라가 못되는 우리의 處地이고 보면 꿈같은 이야기지만 다음 世代에서는 徐徐히 宇宙産業에로의 길이 열리지 않겠는가? 多幸히도 政府에서는 1990年 初에 放送衛星을 띄울 計劃으로 있다. 이것이 곧 宇宙産業으로의 첫발을 내딛는 信號가 아닌가!! 未來의 젊은 世代를 爲해 筆者가 조금이라도 貢獻할 수 있다면 多幸으로 생각하며 斯界 專門家の 呼應을 얻어 coding의 土着化가 하루 速히 이루어졌으면 하는 마음 懇切하다.

3회에 걸친 連載講述에서 紙面의 制限으로 因해 未洽한 點이 많으나 讀者 여러분의 諒解를 求하는 바이며 疑問點에 對해서는 筆者의 著書 “符號理論”(8月 出版 豫定)을 參考하기 바란다. *

◆ 用 語 解 說 ◆

OCR(Optical Character Reader)

光學文字判讀裝置. 일반적으로는 郵便番號判讀 區分機라고도 부른다. 度數計 OCR(度數計 필름判讀裝置) 및 料金收納消去用 OCR이 있다.

度數計 OCR은 取扱局에서 料金미터를 촬영한 필름에서 指數를 직접 읽어 종이 테이프에 出力하는 裝置이고 한편 料金收納消去用 OCR은 料金局에서

송부된 領收單通知書의 顧客情報를 읽어 磁氣테이프에 出力하는 裝置이다.

어느 裝置나 키이 펀처에 의한 穿孔處理를 하지 않고 作業의 迅速化·省力化가 도모되고 있다. 또한 OMR(Optical Mark Reader)는 카아드 등의 마아크를 읽어내는 光學 마아크 判讀裝置를 말한다.