・コンボリューションコード (Chapter 12)

コンボリューションコードは線形符号であり，生成行列で定義される.
符号化はたたみ込み演算(フィルタリング)とみなされる.
実際多用されており，ハードウェアで実現されている.
これは単なるディジタルフィルタとも見れる.
ブロックコードよりコンボリューションコードは人気がある.
同程度の演算複雑度を仮定すればコンボリューションコードはブロックコードより作りやすい.
軟判定方式(ファジィ)としては早期から使われている.

ブロックコードでは k 個入力より n 個出力を作る.
一方コンボリューションコードは流れ的な符号であって，連続入力(シンボル)を取り扱うが，それは有限長ブロックに分割されていない.
k 入力で n 出力のコードを R = k/n コードいい，ブロックコードでも作れる.
この章のブロックコードは GF(2) ('0' '1' だけのガロア体)を用いる.

x の多項式でシーケンスと伝達関数を示す.
F(x)は多項式の集合で，m(x)はF(x)の要素である.

入力が複数の時m(1) m(2)を使う

$$m(x) = \begin{bmatrix} m^{(1)}(x) & m^{(2)}(x) \end{bmatrix} \in \mathbb{F}[[x]]^2.$$

コンボリューションコードの符号化はいくつかディジタルフィルタで表現されている.

例12.1
図12.1にコンボリューションエンコードの例を示す.
ここでDはD-FFメモリ.

$$c_k^{(1)} = m_k + m_{k-2} \quad \text{and} \quad c_k^{(2)} = m_k + m_{k-1} + m_{k-2}.$$

この例では1つの入力に対して2つの出力が出る．なので R = 1/2 コードと呼ぶ.
通常，初期のD-FFの値は'0'である.
符号の世界では+は EXOR.
GF(2)では'0'か'1'しかない

cの式でカンマは単一の入力時間ごとの入力を分けている.

入力 m(x) = 1 x^0 + 1 x^1 + 0 x^2 ...
数列を多項式にする.

g(1)(x) = 1+x^2
g(2)(x) = 1+x+x^2
C(1)(x) = m(x)g(x) = (1+x+x^4+x^6)(1+x^2) = …(1+1)x^6… = 1+x+x^2+x^4+x^8

R = k/n の演算に k × n 行列を使う.
その k × n 行列はG(x)とする.
G(x)を伝達関数マトリックスと呼ぶ.

例12.2
システマティックCエンコーダーとは，出力の1つに入力がそのまま表れることである.
フィードバックのあるフィルターは分数の式になる.
この式ではC(2)(x)は割り算がいる.
割り切れない時は長い項が出る.

G(x)の中に多項式のみであれば，フィードフォワードエンコーダ，すなわちFIR型である.
分数式であれば，フィードバックエンコーダ，すなわちIIR型という

入力が k 個ある場合

$$\mathbf{m}(x) = [m^{(1)}(x), m^{(2)}(x), \ldots, m^{(k)}(x)]$$

and

$$G(x) = \begin{bmatrix} g^{(1,1)}(x) & g^{(1,2)}(x) & \cdots & g^{(1,n)}(x) \\ g^{(2,1)}(x) & g^{(2,2)}(x) & \cdots & g^{(2,n)}(x) \\ \vdots & & & \\ g^{(k,1)}(x) & g^{(k,2)}(x) & \cdots & g^{(k,n)}(x) \end{bmatrix}. \tag{12.1}$$

The output sequences are represented as

$$\mathbf{c}(x) = [c^{(1)}(x), c^{(2)}(x), \ldots, c^{(n)}(x)] = \mathbf{m}(x)G(x).$$

例12.3
R = 2/3 コード の場合 2 × 3 行列が出来る.
c(1),c(2)は入力がそのまま出力となっている

次回は p455 の図の下から