誤り訂正符号概説

島田良作

An Introduction to Error-Correcting Codes

Ryosaku SIMADA

ABSTRACT

This paper presents the outlines of elementary error-correcting codes. The first section is an introductory representation. The Hamming weight and the Hamming distance are fundamental measurs for linear codes. They are introduced in the second section. Examples of Hamming single-error-detecting code, single-error-correcting code, and double-error-detecting code are illustrated in Table 2.1,2.3 and 2.4. respectively

The third section provides the fundamental theories of finite groups, finite rings, Garois fields, vector spaces and matrixes. These mathematics are indispensable to the theories of error-correcting linear codes. If U is a linear code, and is also closed under the cyclic shift, then it is called a cyclic code. Examples of an encoding circuit and of a decoding circuit are illustrated in Fig.4.1 and Fig.4.2 respectively of the fourth section.

In the fifth section, burst-error-correcting codes are investigated in some detail. Fig.5 shows the code-point-distribution on the (g, r) plane. The sixth section describes the convolutional code in short. This type of code is also useful for burst-error-collection.

1．序説

この概論では，伝送路記号（すなわち，符号アルファベット）として0と1を用いる基本的な2元通信路（binary channel）を考える。コンピュータ・メモリーなどにおいても，時には誤りを生じるので，これらに誤り訂正符号が用いられる。

誤り訂正符号は，誤り記号の存否のみを検査するための誤り検出符号（error detecting code）と，誤りを訂正するために用いる誤り訂正符号（error correcting code）に分けることができる。しかし，一般には，誤り検出符号と誤り訂正符号をまとめて，誤り訂正符号と呼ぶ。

通信路における伝送誤りの対策として，以下のような手段が考えられる。

(1) 伝えようとする記号系列を決められた回数だけ繰り返して送信する方法（連送方式）。

(2) 受信記号列をそのまま送信側に返送し，送信側では返送された記号列を送信した記号列の写しと比較照合し，違いがあれば再送信を行う方法（返送照合方式）。

(3) 受け取った一連の記号系列の中に誤りが有るか否かを受信側で検査し，誤りの存在を検出したとき，再送信を要求する方法（誤り検出再送要求方式）。

(4) 受信側で独自に誤りの訂正を行う方法（前向き誤り訂正方式，FEC）。

(5) 軽微の誤りは受信側で独自に訂正し，訂正の限度を超える誤りは，その存在を検出して再送信を要求する方式（前向き訂正・再送要求混成方式）。

以上の(3)～(5)の誤り制御において，受信側で誤りの検出，あるいは誤りの訂正ができるためには，送信側で，その送信記号系列（block, frame, packet）

受理日：平成16年9月16日

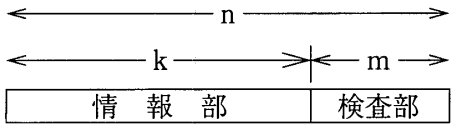
に，いくらかの冗長性を加えておく必要がある。すなわち，伝えるべき記号k個ごとに余分の記号m個を付け加えて送信する。

余分の記号を付加する最も単純な方法は，上記(1)の連送方式であって，その手順の簡単さ，明快さのために実際に使用されることもあるが，情報伝送の能率が著しく低くなるのが問題である。

上述の伝えたいk記号の系列は，情報源符号化の結果得られる符号語，又はそれらの系列であり，その各記号を情報記号(information symbol)と言う。これに対して，誤りの検出や訂正のために付加する余分のm個の記号を検査記号（check symbol）と言う。以上の情報記号と検査記号の合計

$$n = k + m$$

を符号長（code length）と言う。符号語のモデルを以下に例示する。



符 号 語

m個の検査記号のそれぞれは，k個の情報記号の値の組合せによって一意に決められる。このため，符号語の総数は2^kであり，これら2^k個の符号語の集合を符号（code）と言う。

0と1のn記号の系列を2元n項組（binary n-tuple)と言うが，2元n項組は総数2^n（$=2^{m+k}$）個あり，その中の2^k個が符号語として使用されることになる。

送信側でk個の情報記号にm個の検査記号を付加する操作を符号化（coding）と言う。符号化が行なわれて通信路に送り出される符号語を送信語（transmitted word）と言い，通信路を経て受信されるn項組を受信語（received word）と言う。

多くの場合，送信語は正しく伝達され，殆んどの受信語は送信語そのものである。しかし，ときには，伝送途中，雑音のためにいくらか誤った受信語を得る場合がある。受信語に含まれているかも分からない誤りを検出し，あるいは誤りを訂正して正しい送信語，あるいは正しい情報記号列を復元する操作を複号あるいは復号化（decoding）と言う。

この概説では，誤りを検出し，あるいは訂正するための基本的な符号とその符号化，および復号化について概説する。

2．ハミング距離と誤り検出訂正符号

4個の情報記号a_5，a_4，a_3，a_2に1個の検査記号a_1を加えて作った1誤り検出符号（single error detecting code）の例を表2.1に示す。その符号長nは5である。

表2.1　1誤り検出符号の例

a_5	a_4	a_3	a_2	a_1	a_5	a_4	a_3	a_2	a_1
0	0	0	0	0	1	0	0	0	1
0	0	0	1	1	1	0	0	1	0
0	0	1	0	1	1	0	1	0	0
0	0	1	1	0	1	0	1	1	1
0	1	0	0	1	1	1	0	0	0
0	1	0	1	0	1	1	0	1	1
0	1	1	0	0	1	1	1	0	1
0	1	1	1	1	1	1	1	1	0

この符号では，どの符号語においても，記号1の個数が偶数になるように検査記号a_1の値を決めてある。すなわち，この符号は，等式

$$a_5+a_4+a_3+a_2+a_1=0 \qquad (2.1)$$

を満たす5項組（a_5，a_4，a_3，a_2，a_1）すべての集合である。ここに，各記号a_iは0か1であり，その加算は次の式(2.2)，あるいは次の表2.2に従う。

$$0+0=1+1=0,\ 0+1=1+0=1 \qquad (2.2)$$

表2.2　法2に関する加算

+	0	1
0	0	1
1	1	0

このような加算を法2に関する加算（addition modulo 2）と言う。

表2.1の1つの符号語（a_5, a_4, …, a_1）を送信し，5項組の1つ

$$(b_5, \cdots, b_1) = (10011) \qquad (2.3)$$

を受信したとする。その5記号 b_5～b_1の法2に関する加算和が

$$b_5+\cdots+b_1 = 1+0+0+1+1 = 1 \qquad (2.4)$$

であって，この受信語は式（2.1）を満たさない。このことから，上記の受信語（$b_5, \cdots, b_1$）の中の1記号の誤り，すなわち単1誤り（single error）の存在を検出したことになる。

しかし，受信側では，記号 b_5～b_1の中のどの記号が誤ったかは分からず，従って訂正は不可能である。実際，1誤りによって受信語（10011）をもたらす符号語として，

（00011）　（11011）　（10111）
（10001）　（10010）

の5個が考えられる。これらの中のどの符号語に復号すべきかは決められない。このような符号を単1誤り検出記号（single error detecting code）と言う。

また，受信語（b_5, b_4, …, b_1）に対して，法2に関する加算 $b_5+\cdots+b_2+b_1$ を行ない，その結果が0であるか1であるかの検査を行う。これをパリティ検査（parity check），あるいは奇偶検査と言う。

例として，表2.1の符号の2つの符号語（00011）と（00101）を考える。これらの符号語の，同じ位置の記号を，それぞれ比較する。

　　＊＊
（0 0 0 1 1）
（0 0 1 0 1）

この2つの符号語では，上記のように＊印の2個所で記号の値が互に違っている。このように，2つの符号語の間で，互に一致しない記号の位置の数を，その符号語間のハミング距離（Hamming distance），あるいは単に距離と言う。上の2つの符号語の間のハミング距離は $d=2$ である。

ある符号において，その符号語の全ての対（pair）を考える。これらの対をなす2つの符号語の間のハミング距離を調べるものとする。これらのハミング距離の中で最小のものをこの符号の最小距離（minimum distance）と言い，これを d_m で表すものとする。

表2.1の符号は単1誤り検出符号であり，その最小距離は $d_m=2$ である。このような符号の任意の符号語を送信し，伝送中に1誤りが生じたとする。この場合の受信語と送信語の間のハミング距離は1であり，このため，この受信語はどの符号語でもあり得ない。このことから，受信語の中の単1誤りの存在は検知可能であるが，誤った記号を特定することは不可能である。

最小距離が $d_m=2$ の符号，すなわち，単1誤り検出符号は，一般に k 個の情報記号 a_{k+1}, a_k, …, a_2 に1個の検査記号 a_1 を加えて構成される。ここに，等式

$$a_{k+1}+a_k+\cdots+a_2+a_1 = 0, \quad (\mathrm{mod}\,2) \qquad (2.5)$$

が成り立つように，a_1 の値を0か1に決めるものとする。

この符号は，符号化率 $k/(k+1)$ が大きいこと，符号化が簡単であること，受信語の検査が容易であることのために，1誤り検出符号（single error detecting code）として，しばしば使用される。

次の表2.3に示す符号は，$k=4$ 個の情報記号 a_7, a_6, a_5, a_3 に，$m=3$ 個の検査記号 a_4, a_2, a_1 を加えて作った符号であり，その符号長は $n=7$ である。一般に符号長が n であり，情報記号数が k

である符号を（n,k）符号と言う。表2.3の符号は（7,4）符号である。

表2.3　1誤り訂正（7,4）符号

a_7	a_6	a_5	a_4	a_3	a_2	a_1
0	0	0	0	0	0	0
0	0	0	0	1	1	1
0	0	1	1	0	0	1
0	0	1	1	1	1	0
0	1	0	1	0	1	0
0	1	0	1	1	0	1
0	1	1	0	0	1	1
0	1	1	0	1	0	0
1	0	0	1	0	1	1
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	0	1	0	1
1	1	0	0	0	0	1
1	1	0	0	1	1	0
1	1	1	1	0	0	0
1	1	1	1	1	1	1

以上の情報記号を，以下のように重複を許して3組に分け，その各組に，検査記号a_4，a_2，a_1を重複しないように分配し，各組の和が0になるように，a_4，a_2，a_1の値を決める。その情況を次式に示す。

$$\left.\begin{aligned} a_7+a_6+a_5+a_4 &= 0 \\ a_7+a_6+a_3+a_2 &= 0 \\ a_7+a_5+a_3+a_1 &= 0 \end{aligned}\right\} \quad (2.6)$$

この（7,4）符号は，上式を満たす2元7項組（binary 7 -tuple）（a_7, a_6, …, a_1）の全て16個の集合である。

例えば，符号語$u=(a_7, \cdots, a_1)=(0011001)$を送信して，相手側で受信語$v=(b_7 \cdots b_1)=(0111001)$を得たとする。この受信語$v$に対して，式（2.6）と同じパリティ検査を行うとき，次の結果を得る。

$$\left.\begin{aligned} b_7+b_6+b_5+b_4 &= 1 \\ b_7+b_6+b_3+b_2 &= 1 \\ b_7+b_5+b_3+b_1 &= 0 \end{aligned}\right\} \quad (2.7)$$

この場合，上部の2式の検査結果が不合格であり，下部の1式の検査結果のみが合格である。これらの結果から，上部の2式の検査に含まれ，下部の1式の検査に含まれない記号$b_6=1$が誤っていると判定することができる。このb_6に1を加算（法2に関する加算）を行って，その値を0に訂正することにより，受信語vを送信語uに訂正することができる。

この例のように，受信語の任意の1記号に限って，誤りの訂正が可能な符号を単1誤り訂正符号（single error correcting code）と言う。

一般に（n,k）符号が単1誤り訂正符号であって，式（2.7）の形のパリティ検査により，誤り記号の指摘ができるためには，符号語のn個の記号のそれぞれがn−k（=m）個の検査方程式のどれかに属し，その属し方がことごとく違っていなければならない。このことから，符号長nと検査方程式の個数mは，等式又は不等式

$$n \leqq \binom{m}{1}+\binom{m}{2}+\cdots+\binom{m}{m}=2^m-1 \quad (2.8)$$

を満たさなければならない。

上式において，特に等式が成り立つ場合の（n, k）符号は，2つの等式

$$n=2^m-1,\ k=2^m-1-m \quad (2.9)$$

を満たす符号，即ち，（2^m-1，2^m-1-m）符号である。このような符号を，一般に，狭い意味でハミング符号（Hamming code）と呼んでいる。表2.3の符号は，m=3の場合の例であり，これは（7,4）ハミング符号である。

以上の（7,4）ハミング符号の各符号語に，1個のパリティ検査記号a_0を加えて，最小距離が$d_m=4$の符号を作ることができる。これを表2.4に示す。

更に，これらの符号の最小距離d_mと，誤り検出能力d_t，及び誤り訂正能力c_rの関係の一部を表2.5に示す。

表2.4　1誤り訂正2誤り検出(8,4)符号

a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1
0	0	1	1	0	0	1	1
0	0	1	1	1	1	0	0
0	1	0	1	0	1	0	1
0	1	0	1	1	0	1	0
0	1	1	0	0	1	1	0
0	1	1	0	1	0	0	1
1	0	0	1	0	1	1	0
1	0	0	1	1	0	0	1
1	0	1	0	0	1	0	1
1	0	1	0	1	0	1	0
1	1	0	0	0	0	1	1
1	1	0	0	1	1	0	0
1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1

表2.5　d_m,d_t,c_r の関係

d_m	d_t	c_r
2	1	0
3	2	0
	0	1
4	3	0
	2	1
5	4	0
	3	1
	0	2
6	5	0
	4	1
	3	2
⋮	⋮	⋮

3．線形ブロック符号

集合Gの任意の元に対して1つの2項演算∘が定義されており，これが次の公理G1～G4を満たすとき，集合Gを群（group）と言う。

G1．（閉塞）Gの任意の元a,bに対してa∘bはGの元である。

G2．（結合則）Gの任意の元a,b,cに対して，等式（a∘b）∘c=a∘（b∘c）が成り立つ。

G3．（恒等元）Gの任意の元aに対して，等式a∘I=I∘a=aを満たす元IがGに存在する。このような元Iを恒等元（identity element）と言う。

G4．（逆元）Gの任意の元aに対して，等式a∘a′=a′∘a=Iを満たす元a′がGに存在する。このようなa′をaの逆元(inverse element）と言う。

演算∘として加算記号＋を用いるとき，群Gを加法群又は加群と言う。加群においては恒等元を0で表し，元aの逆元を－aで表す。また，演算∘として乗算記号・を用いるとき，群Gを乗法群と言う。乗法群においては，恒等元を1で表し，元aの逆元をa^{-1}で表す。

集合｛0，1｝は，式（2.2），あるいは表2.2に示した加算，すなわち法2に関する加算の下に群を成す。

集合Rの元に対して加算＋と乗算・が定義されており，これらが次の公理R1～R4を満たすとき，集合Rを環（ring）と言う。（しばしば，乗算記号・を省略する）。

R1．（加群）Rは加算の下に可換群，すなわち，交換則a+b=b+aを満たす群をなす。

R2．（閉塞）Rの任意の元a,bに対して，積abはRの元である。

R3．（結合則）Rの任意の元a,b,cに対して，等式（ab）c=a（bc）が成り立つ。

R4．（分配則）Rの任意の元a,b,cに対して，等式a（b+c）=ab+acおよび（a+b）c=ac+bcが成り立つ。

特に乗算における交換則ab=baを満たす環を可換環（commutative ring）と言う。整数すべての集合は，もっとも身近な可換環である。集合R=

｛0，1｝は，式（2.2）に示した法2に関する加算と通常の乗算の下に環を成す。一般に，任意の正整数rに対して，法rに関する整数の剰余の集合｛0，1，…，r－1｝は，法rに関する加算と乗算の下に環を成す。このような環を整数の剰余環（residue class ring）と言う。

一般に，環Rの部分集合Jが加算の下にRの部分群を成し，Jの任意の元aとRの任意の元rの積ar及びraが，いずれも，Jの元であるとき，このようなJを環Rのイデアル（ideal）と言う。

集合Fが可換環であり，Fの0以下の元の集合が乗法群を成すとき，この集合Fを体（field）と言う。

実数すべての集合，複素数すべての集合は，いずれも体を成す。前者を実数体（real field）と言い，後者を複素数体（complex field）と言う。集合｛0，1｝は，式（2.2）あるいは表2.2に示した法2に関する加算と乗算

$$0\cdot 0=0\cdot 1=1\cdot 0=0,\ 1\cdot 1=1 \qquad (3.1)$$

の下に体を成す。2元から成るこのような体を2元のガロア体（Garois field of two elements），あるいは2元体と言い，これをGF(2)のように表す。

任意の素数をpとするとき，集合｛0，1，…，p－1｝は，法pに関する加算と乗算の下にp元体，すなわち，GF(p)を成す。以下では，もっぱら，2元体を扱う。

集合Vが次の公理V1～V5を満たすとき，このVを体F上のベクトル空間（vector space）と言う。

V1．（アーベル群）Vは加算の下にアーベル群を成す。

V2．（スカラー倍）Vの任意の元vと体Fの任意の元cに対して，積cvが定義されており，このcvがVの元である。

V3．（分配則）Vの任意の元u，vと体Fの任意の元cに対して，c$(u+v)$＝cu＋cvが成り立つ。

V4．（分配則）Vの任意の元vと，体Fの任意の元c,dに対して，(c+d)v＝cv＋dvが成り立つ。

V5．（結合則）(cd)v＝c(dv)，1v＝vが成り立つ。

ベクトル空間Vの元u，v，…をベクトル（vector）と言い，体Fの元c，d，…をスカラー（scalar）と言う。2元体GF(2)を含めて，一般に体Fの元a_1，a_2，…，b_1，b_2，…の任意のn項組$(a_1,\cdots,a_n)$，$(b_1,\cdots,b_n)$及びFの任意の元cに対して，スカラー倍及び加算を次のように定義する。

$$c\,(a_1,\cdots,a_n)=(ca_1,\cdots,ca_n), \qquad (3.2)$$

$$(a_1,\cdots,a_n)+(b_1,\cdots,b_n)=(a_1+b_1,\cdots,a_n+b_n). \qquad (3.3)$$

ここに，ca_1，ca_2，…などの乗算，及びa_1+b_1，a_2+b_2，…などの加算は，体Fの乗算及び加算に従う。

体Fの元のn項組すべての集合は，以上のスカラー倍及び加算の下にベクトル空間を成す。誤り訂正符号の理論，すなわち符号理論では，2元体GF(2)を始めとして，一般に有限体のn項組から成るベクトル空間の理論が有用である。

体Fの元のn項組$(a_1,\ a_2,\ \cdots,\ a_n)$全ての集合が成すベクトル空間を$V$とする。この$V$のn個のベクトル

$$\left.\begin{aligned} v_1&=(100\cdots0)\\ v_2&=(010\cdots0)\\ &\cdots\\ v_n&=(000\cdots1)\end{aligned}\right\} \qquad (3.4)$$

と，これらの線形結合（linear combination）

$$v=a_1v_1+a_2v_2+\cdots+a_nv_n=(a_1a_2\cdots a_n),\ (a_i\in F) \qquad (3.5)$$

を考える。このvが零ベクトル（00…0）になるのは，上式の係数$a_1,\cdots a_n$のすべてが0のとき，そのときに限る。このことから，式(3.4)のベクトルの集合｛v_1，v_2，…，v_n｝は線形独立（linearly independent）であると言う。

以上のベクトル空間Vが，線形独立なn個のベクトルv_1～v_nの線形結合，すなわち，1次式

で過不足なく表されることから，そのVを，n個の基底ベクトル（basis vector）が張るn次元のベクトル空間（n-dimensional vector space），あるいは簡単にn次元空間と言う。

n次元のベクトル空間におけるk個のベクトルの集合$\{u_1,\cdots,u_k\}$を考える。ここに，k<nであり，集合$\{u_1,\cdots,u_k\}$は線形独立であるとする。これらのk個のベクトルの線形結合

$$u=b_1u_1+b_2u_2+\cdots+b_ku_k,\ (b_i\in F) \qquad (3.6)$$

を考える。これらの線形結合の全ての集合をUとするとき，Uはk次元のベクトル空間であり，しかも上述のn次元空間の部分空間である。このことから，UをVのk次元の部分空間（k-dimensional subspace）と言う。

特に，GF(2)の元0と1のn項組すべてから成るn次元空間Vのk次元部分空間Uが有用である。このような部分空間Uを2元線形(n, k)符号（binary linear (n, k) code），あるいは単に，線形（n, k）符号と言う。そのnを符号長（code length)と言い，kを情報記号数(number of information symbols）と言う。

表2.3に示した符号Uは，線形（7,4）符号である。その中の4個の符号ベクトル

$$\left.\begin{aligned} u_1&=(1001011)\\ u_2&=(0101010)\\ u_3&=(0011001)\\ u_4&=(0000111)\end{aligned}\right\} \qquad (3.7)$$

は線形独立であって，この部分空間Uを張る。言い換えれば，この符号Uは，ベクトルu_1～u_4が張る線形（7,4）符号である。

n次元空間Vの2つのベクトル$v_1=(a_1,\cdots,a_n)$と$v_2=(b_1,\cdots,b_n)$に対して，その積を

$$\begin{aligned} v_1\cdot v_2&=(a_1,\cdots,a_n)\cdot(b_1,\cdots,b_n)\\ &=a_1b_1+a_2b_2+\cdots+a_nb_n \end{aligned} \qquad (3.8)$$

のように定義して，これをv_1とv_2の内積（inner product, dot product）と言う。

$v_1\cdot v_2=0$のとき，v_1とv_2は互に直交する（be orthogonal）と言う。ベクトル空間Vの2つの部分空間U_1とU_2において，U_1の各ベクトルがU_2の全ベクトルに直交するとき，U_1とU_2は直交すると言う。

ベクトル空間Vの部分空間U_1とU_2が直交し，これらの次元の和がVの次元に等しいとき，U_1はU_2の直交補空間（orthocomplement）と言い，U_2はU_1の直交補空間と言う。

式（3.7）に例示した4個のベクトルu_1～u_4のそれぞれを行ベクトルとして，次の式（3.9）に示す4×7行列Gを作ることができる。これは，表2.3に示した(7,4)符号の生成行列（generating matrix）である。

$$G=\begin{bmatrix}1&0&0&1&0&1&1\\0&1&0&1&0&1&0\\0&0&1&1&0&0&1\\0&0&0&0&1&1&1\end{bmatrix} \qquad (3.9)$$

与えられた4個の情報記号の系列

$x=(a_7\,a_6\,a_5\,a_3)$

を1×4行列と見なして，これに生成行列Gを乗算する。

$$\begin{aligned} u&=xG\\ &=a_7u_1+a_6u_2+a_5u_3+a_3u_4\\ &=(a_7\,a_6\,a_5\,a_4\,a_3\,a_2\,a_1), \end{aligned}$$

但し，

$$\left.\begin{aligned} a_4&=a_7+a_6+a_5\\ a_2&=a_7+a_6+a_3\\ a_1&=a_7+a_5+a_3\end{aligned}\right) \qquad (3.10)$$

このようにして，符号語$u=(a_7a_6\cdots a_1)$を生成することができる。ここに，a_4，a_2，a_1が検査記号（check symbol）である。例えば，情報記号の系列$x=(1011)$は，符号語

$$u=xG=(1010101) \qquad (3.11)$$

に符号化される。

一般に，線形（n,k）符号U_1は，あるk×n行列Gで生成されるk次元のベクトル空間である。その直交補空間U_2は，等式

$$GH^T=0 \qquad (3.12)$$

を満たす（n−k）×n行列Hで生成されるn−k次元のベクトル空間であり，しかも線形（n，n−k）符号である。このような行列Hを，符号U_1のパリティ検査行列（parity check matrix）と言

う。

線形（n，k）符号Uは，その生成行列Gのk個の行ベクトルu_1，u_2,…,u_kの線形結合$b_1u_1+b_2u_2+\cdots+b_ku_k$すべての集合であるが，これは同時に，等式

$$uH^T=0 \tag{3.13}$$

を満たすn項組uすべての集合である。

表2.3の線形（7,4）符号Uにおいては，任意に与えられた情報記号の組a_7，a_6，a_5，a_3に対して，3個の検査記号a_4，a_2，a_1が，式（3.10）に従って決められるとき，従って，2元体の元の7項組$u=(a_7, a_6,\cdots,a_1)$が等式

$$\left.\begin{aligned} a_7+a_6+a_5+a_4 &= 0\\ a_7+a_6+a_3+a_2 &= 0\\ a_7+a_5+a_3+a_1 &= 0 \end{aligned}\right\} \tag{2.4}$$

を満たすとき，そのときに限って，7項組uは，線型（7,4）符号Uの符号語である。

符号Uの符号語$u=(a_7, a_6,\cdots,a_1)$は，1行7列の行列と考えることができる。更に，行列

$$H=\begin{bmatrix}1&1&1&1&0&0&0\\1&1&0&0&1&1&0\\1&0&1&0&1&0&1\end{bmatrix} \tag{3.14}$$

は，上述の線形（7,4）符号Uのパリティ検査行列である。

通信路の一端から符号語uを送信し，その他端において，7項組$v=(b_7\,b_6\cdots b_1)$を受信したとする。受信側では，この受信語vに対して，

$$S=vH^T=(s_4\,s_2\,s_1) \tag{3.15}$$

なる計算を行う。これを受信語vのパリティ検査（parity check）と言い，検査の結果$S=(s_4\,s_2\,s_1)$を受信語vのシンドローム（syndrome）と言う。

受信語vのシンドロームが$S=(000)$であれば，vは正当な符号語である。例えば符号語$u=(0011001)$を送信し，伝送途中に1誤りを生じて受信語

$$v=(0111001) \tag{3.16}$$

を得たとする。この受信語vのシンドロームは，$S=vH^T=(110)$になる。このSを転置行列（transposed matrix）にすれば，

$$S^T=Hv^T=\begin{bmatrix}1\\1\\0\end{bmatrix} \tag{3.17}$$

を得る。これをパリティ検査行列Hの各列ベクトルと比較する。上のS^Tは，Hの右から数えて第6の列ベクトルに一致する。このことから，受信語vの右から第6の文字$b_6=1$の誤りが判明する。これを0に変更することにより，上の受信語$v=(0111001)$を送信語$u=(0011001)$に訂正することができる。

一般に，線形（n,k）符号のパリティ検査行列Hは（n−k）×n行列である。そのn個の列ベクトルの中からd_m個の列ベクトルを選出して，これらの列ベクトルの集合を作るものとする。このような集合は，全部で

$$\binom{n}{d_m}$$

個存在する。これらの集合の中に線形従属（linearly dependent）のものがあれば，この線形（n,k）符号の最小距離はd_m以下である。

また，Hのd_m-1個の列ベクトルの集合のどれもが線形独立（linearly independent）であれば，この線形（n,k）符号の最小距離はd_m以上である。

例えば，式（3.14）のパリティ検査行列Hの2列の集合は，すべて線形独立である。他方，線形従属の3列の集合は多数存在する。例えば，Hの右端の3列は，等式

$$\begin{bmatrix}0\\1\\1\end{bmatrix}+\begin{bmatrix}0\\1\\0\end{bmatrix}+\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix} \tag{3.18}$$

を満たす。従って，この符号の最小距離は$d_m=3$である。

以上のように最小距離が$d_m=3$である線形（n,k）符号Uの各符号語$(a_n\cdots a_1)$に1個の検査記号a_0を加えて，n+1項組$(a_n\cdots a_1a_0)$を作る。ここに，a_0はa_n〜a_0の全記号のパリティ検査を行うためのものである。このようにして得られる新しい線形（n+1，k）符号の最小距離は$d_m=4$で

あって，この符号は，1誤り訂正2誤り検出符号，あるいは，単純な3誤り検出符号として使用することができる。

次に示す行列Hは，前例の線形（7,4）符号の各符号語に検査記号a_0を加えて作った線形（8,4）符号の検査行列である。この符号の最小距離は$d_m=4$である。

$$H=\begin{bmatrix}1&1&1&1&0&0&0&0\\1&1&0&0&1&1&0&0\\1&0&1&0&1&0&1&0\\1&1&1&1&1&1&1&1\end{bmatrix} \tag{3.19}$$

以上に，最小距離が$d_m=2$，3，4の3種類の線形符号について，それらの概要を示した。これらは，R. W. Hammingによって提示されたものであって，明快な代数構造を持つ最初の符号である。以来，この50年余りの間に，有用な多くの誤り訂正符号が開発され，広く使用されるようになった。

積符号（product code）は，表3.1に例示するように，情報記号を長方式に並べ，その各行各列に検査記号を付加し，その各行を1つの線形符号U_1の符号語に，そして各列もまた，1つの線形符号U_2の符号語に符号化して，長方形の配列全体を1つの符号語にしたものである。そのU_1を行符号と言い，U_2を列符号と言う。また，このような積符号を$U_1\times U_2$符号とも言う。

行符号U_1及び列符号U_2の最小距離が，それぞれ，d_1及びd_2であるとき，積符号$U_1\times U_2$の最小距離は次式で与えられる，

$$d_m=d_1\times d_2. \tag{3.20}$$

小さい最小距離の簡単な2つの線形符号U_1とU_2を組み合せて，大きな最小距離の符号が得られるのは，積符号の1つの特長である。しかし，積符号の大きな最小距離d_mをいっぱいに使用する誤り訂正は，必ずしも簡単でない。

最も簡単で，よく使用される積符号は，行符号U_1及び列符号U_2として，最小距離がそれぞれ，$d=2$のハミング符号を用いるものである。この積符号の最小距離は$d_m=4$である。このような積符号は，しばしば，水平垂直パリティ検査符号と言う。その列を表3.1（b）に示す。この例では，どの行も，また，どの列も，記号1の個数が偶数になるように各検査記号を決めてある。

1記号の誤りは，パリティ検査で不合格の1行と1列の交点から見付け出し，容易に訂正することができる。2記号の誤りは，

（1）パリティ検査に不合格の2行，

（2）パリティ検査に不合格の2列，

（3）パリティ検査に不合格の2行2列

のどれかの形で，誤りの存在のみが検出できる。2誤りの訂正は不可能である。

表3.1　積符号の例

情報記号	行検査記号
列検査記号	

(a) 一般的な配列

1	1	0	1	1
1	0	0	0	1
1	0	1	0	0
0	1	1	0	0
0	0	1	1	0
0	1	1	1	1
1	1	0	1	1

(b) 符号語の例

4．巡回符号

ある2元線形（n, k）符号Uの任意の符号語を

$$u=(a_{n-1},\ a_{n-2},\cdots,a_0) \tag{4.1}$$

とする。この符号語を巡回シフトして得られるn項組

$$u_1=(a_{n-2},\cdots,a_0,\ a_{n-1}) \tag{4.2}$$

が同じ符号Uの符号語であるとき，このような符号Uを巡回符号（cyclic code）と言う。ここでは，2元巡回符号の概要を述べる。

2元のガロア体GF(2)の元$a_{n-1},\cdots,a_1,\ a_0$を係数とする多項式

$$f(X)=a_{n-1}X^{n-1}+\cdots+a_1X+a_0 \tag{4.3}$$

を，一般に，GF(2)上の多項式（polynomial over

GF(2))と言う。このような多項式f(X)は，GF(2)の元のn項組 $(a_{n-1},\cdots,a_1,\ a_0)$ のもう1つの表現形式である。

GF(2)上のm次の多項式

$$g(X)=a_mX^m+\cdots+a_1X+a_0,\quad (a_m=a_i=a_0=1) \tag{4.4}$$

を考える。ここに，m次の項及び0次の項を含めて3項以上の係数が1であるものとする。更に，X^n-1 の形の多項式の中で，g(X)が割り切る最小次数のものを改めて X^n-1 とする。

上のg(X)が割り切るn次未満の多項式全ての集合を U とする。ここで，

$$k=n-m \tag{4.5}$$

とする。集合 U の中のk個の多項式の集合

$$\{g(X),Xg(X),\cdots,X^{k-1}g(X)\} \tag{4.6}$$

は線形独立であり，しかも，これらk個の多項式の線形結合でもって，集合 U のすべての多項式を表すことができる。このため，集合 U は，式(4.6)に示したk個の多項式 $g(X),Xg(X),\cdots,X^{k-1}g(X)$ が張るk次元のベクトル空間であり，従って，線形(n, k)符号である。

以上の符号 U の任意の符号語を次式の u，あるいはu(X)とする。

$$\left.\begin{aligned} u&=(a_{n-1},\cdots,a_1,\ a_0)\\ u(X)&=a_{n-1}X^{n-1}+\cdots+a_1X+a_0\end{aligned}\right\} \tag{4.7}$$

ここに，u は符号語のベクトル表現であり，u(X)は同じ符号語の多項式表現である。これらの符号語の巡回シフト(cyclic shift)は，それぞれ，

$$\left.\begin{aligned} u_1&=(a_{n-2},\cdots,a_0,\ a_{n-1})\\ u_1(X)&=a_{n-2}X^{n-1}+\cdots+a_0X+a_{n-1}\end{aligned}\right\} \tag{4.8}$$

であるが，この多項式表現の $u_1(X)$ は，次式のように表すことができる。

$$u_1(X)=X\cdot u(X)-a_{n-1}(X^n-1). \tag{4.9}$$

式(4.4)の多項式g(X)が X^n-1 及び符号語u(X)を割り切るので，このg(X)は上の $u_1(X)$ を割り切る。従って，この $u_1(X)$ も線形(n, k)符号 U の符号語である。以上により，U は巡回シフトの下に閉じており，従って巡回(n, k)符号である。また，以上の多項式g(X)を，この巡回符号の生成多項式(generating polynomial)と言う。

例えば，多項式 X^7-1 は，次式のように素因数に分解することができる。

$$X^7-1=(X+1)(X^3+X+1)(X^3+X^2+1) \tag{4.10}$$

その最後尾の多項式

$$g(X)=X^3+X^2+1 \tag{4.11}$$

は，巡回(7,4)符号 U を生成する。この U の4個の符号語

$$\left.\begin{aligned} X^3g(X)&=X^6+X^5+X^3\\ X^2g(X)&=X^5+X^4+X^2\\ Xg(X)&=X^4+X^3+X\\ g(X)&=X^3+X^2+1\end{aligned}\right\} \tag{4.12}$$

は基底として，この巡回符号 U を張る。これに対応する行列

$$G=\begin{bmatrix}1&1&0&1&0&0&0\\0&1&1&0&1&0&0\\0&0&1&1&0&1&0\\0&0&0&1&1&0&1\end{bmatrix} \tag{4.13}$$

は，U の生成行列の1つである。この行列 G の第1行に第2行と第3行を加算し，第2行に第3行と第4行を加算し，第3行に第4行を加えることによって，次の行列 G' を得る。この G' も，同じ符号の生成行列であるが，その左部の4行4列の部分に単位行列 I_4 を持つようになる。このような型の行列を既約梯陣形(reduced echelon form)と言う。

$$G'=\begin{bmatrix}1&0&0&0&1&1&0\\0&1&0&0&0&1&1\\0&0&1&0&1&1&1\\0&0&0&1&1&0&1\end{bmatrix} \tag{4.14}$$

行列 G，あるいは G' の行ベクトルの任意の線形結合を考える。その線形結合に対応する多項式が，式(4.11)のg(X)で整除されることは明らかである。符号 U の任意の符号語を

$$u(X)=a_6X^6+a_5X^5+a_4X^4+a_3X^3+a_2X^2+a_1X+a_0 \tag{4.15}$$

とする。この符号語を1桁だけ巡回シフトしたものを $u'(X)$ とすると，これは次のように書くこ

とができる。

$$u'(X)=a_5X^6+a_4X^5+\cdots+a_0X+a_6$$
$$=(a_6X^6+a_5X^5+\cdots+a_1X+a_0)X$$
$$-a_6X^7+a_6$$
$$=Xu(X)-a_6(X^7-1) \qquad (4.16)$$

のように書くことができる。g(X)がu(X)及びX^7-1を整除することから，g(X)はu′(X)を整除する。このことは，この符号Uが巡回シフトのもとに閉じた線形符号であること，すなわち，巡回符号であることを示している。

この符号の生成多項式g(X)がm＝3次であり，符号長がn＝7であるから，この巡回符号Uの情報記号数は，k＝n－m＝4である。4個の情報記号（a_6，a_5，a_4，a_3）を

$$f(X)=a_6X^6+a_5X^5+a_4X^4+a_3X^3 \qquad (4.17)$$

なる多項式で表す。これを生成多項式$g(X)=X^3+X^2+1$で割った余りを

$$r(X)=a_2X^2+a_1X+a_0 \qquad (4.18)$$

とする。すなわち，$f(X)=g(X)h(X)+r(X)$とするとき，

$$f(X)-r(X)=g(X)h(X) \qquad (4.19)$$

はg(X)の倍数であり，従って，巡回符号Uの符号語である。

例えば，情報記号として，（a_6，a_5，a_4，a_3）＝（1，0，0，1）が与えられたとすると，

$$f(X)=X^6+X^3$$
$$=(X^3+X^2+1)(X^3+X^2+X+1)$$
$$+(X+1) \qquad (4.20)$$

であり，余りは$r(X)=X+1$である。上式の両辺からr(X)を減算して，符号語

$$u(X)=f(X)-r(X)$$
$$=X^6+X^3+X+1 \qquad (4.21)$$

を得る。これを符号ベクトル

$$u=(1001011) \qquad (4.22)$$

として伝送路に送り出す。

以上の原理にもとづく符号器のブロック構成を図4.1に示す。これは，f(X)をg(X)で割った余りr(X)を発生するための回路であり，帰還シフトレジスタ（feedback shift register）と言う。

送出するべき情報記号a_6，a_5，a_4，a_3を順次入力し，同時に，これらを伝送路に送り出す。最後の情報記号a_3を送り出したとき，検査記号a_2，a_1，a_0がシフトレジスタ内に作られている。この時点で，スイッチS_1，S_2を同時に切り替え，今送り出した情報記号列に引き続いて，検査記号a_2，a_1，a_0を順に送り出す。

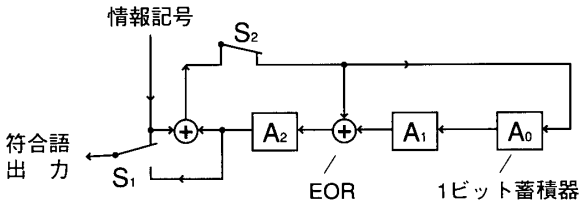


図4.1　巡回(7.4)符号の符号器

例えば，情報記号（a_6，a_5，a_4，a_3）＝（1，0，0，1）を順次入力するとき，帰還シフトレジスタ（A_2，A_1，A_0）の内容は，順次

（1 0 1），（1 1 1），（0 1 1），（0 1 1）

のようになる。この最後の時点でスイッチS_1及びS_2を切り換え，上の情報記号列（1001）に引き続いて，検査記号列（011）を送り出す。つまり，情報記号列に検査記号列を付け加えて（1001011）の形の符号語を送り出す。

受信語υ(X)の誤りの検査は，υ(X)を生成多項式g(X)で割った余りs(X)で以って行う。このs(X)をυ(X)のシンドローム（syndrome）と言う。υ(X)が符号語であればs(X)＝0であり，符号語でなければs(X)≠0になる。たとえば，符号語$u(X)=X^6+X^3+X+1$を送信し，1誤りを生じて$\upsilon(X)=X^6+X+1$を受信したとする。このとき，

$$\left.\begin{aligned}\upsilon(X)&=X^6+X+1\\&=(X^3+X^2+1)(X^3+X^2+X)+(X^2+1)\\s(X)&=X^2+1\end{aligned}\right\} \qquad (4.23)$$

となる。

この符号は，前節で述べた（7,4）ハミング符号に等価であり，1誤りの訂正能力を持つ。その復号器を図4.2に例示する。受信語υ(X)を下部の7ビット・レジスタに入力するとともに，上部の帰還シフトレジスタ（すなわち，シンドローム

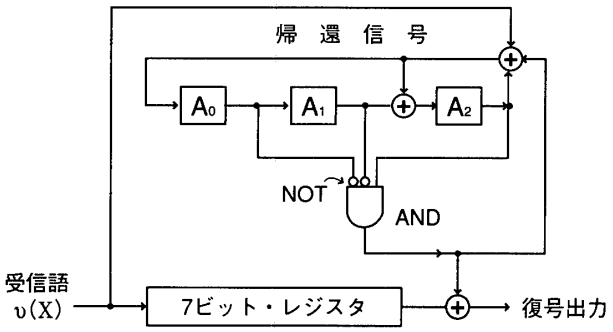


図4.2　巡回(7,4)符号の複号器

発生器）で受信語のシンドロームを計算する。

この複号器では，受信語$\upsilon(X)$の代りに$X^{n-k}\upsilon(X)$を$g(X)$で割った余り$s'(X)$を計算する。これは，式（4.23）の$s(X)=X^2+1$をシンドローム発生器で$n-k=3$回シフトしたものに一致する。式（4.23）の受信語$v=(b_6\ b_5\cdots b_0)=(1000011)$を入力すると，シフトレジスタの中に$(A_0\ A_1\ A_2)=(011)$を得る。

7ビット・レジスタから受信記号b_6～b_0を順次読み出すとともに，シンドローム発生器をシフトする。その状況を表4.1に示す。

表4.1　誤りの訂正

	A_0	A_1	A_2	
b_6	0	1	1	
b_5	1	0	0	
b_4	0	1	0	
b_3	0	0	1	（訂正）
b_2	0	0	0	
b_1	0	0	0	
b_0	0	0	0	

丁度誤り記号b_3を読み出す時点において，シフトレジスタの内容が$(A_0\ A_1\ A_2)=(001)$になり，ANDゲイトの出力が1になる。これにより，7ビット・レジスタから出て来た$b_3=1$が0に訂正され，また同時に，帰還信号も0にされる。このようにして，受信語$\upsilon(X)=X^6+X+1$が元の送信語$u(X)=X^6+X^3+X+1$に訂正される。

もっとも重要な巡回符合はBCH符号（Bose-Chaudhuri-Hocquenghem code）及びRS符号（Reed-Solomon code）であると言われている。これらについては省略する。

5．バースト誤り訂正符号

伝送する符号語の各記号が，前後の記号に関係なく，それぞれ独立に変化する誤りをランダム誤り（random error）と言う。これに対して，比較的に短い範囲の複数の記号がまとまって変化する誤りをバースト誤り（burst error）と言う。例えば，以下に例示するように，ある符号語uを送信し，相手側で受信語vを得たとする。

$u=$（101101110000）

$v=$（101011010000）

$e=$（000110100000）

その差（各記号ごとの排他的論理和）eにおける第4記号の“1”から第7記号の“1”までを誤りの範囲と見て，この範囲の記号の個数$b=4$をバースト長（burst length）と言う。このようなバースト誤りを検出し，あるいは訂正するための符号をバースト誤り訂正符号（burst error correcting code）と言う。

例えば，ISOやJISで標準化されたハイレベルデータリンク制御手順（high level date link control procedure, HDLC）における巡回冗長検査（cyclic redundancy check, CRC）として，多項式

$$G(X)=X^{16}+X^{12}+X^5+1 \qquad (5.1)$$

で生成される巡回符号が用いられる。また，イサーネット（Ethernet）では，多項式

$$G(X)=X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1 \qquad (5.2)$$

で生成される巡回符号が用いられる。

これらの符号をバースト誤りの検出に用いるなら，前者は長さ16まで，後者は長さ32までのバースト誤りの検出が可能である。

バースト誤り訂正符号を構成するための1つの方法として，例えば，1誤り訂正（n,k）符号Uのi個の符号語$(a_{n-1},\cdots,a_0),(b_{n-1},\cdots,b_0),\cdots,(c_{n-1},\cdots,c_0)$を組み合せて，（n×i）項組

$$(a_{n-1},\ b_{n-1},\cdots,c_{n-1},\ a_{n-2},\ b_{n-2},\cdots,c_{n-1},\ \cdots,a_0,\ b_0,\ \cdots,c_0) \qquad (5.3)$$

を作る。このような（n×i）項組を符号語とする新しい符号U'は，長さiのバースト誤り訂正能力を持つ（ni,ki）符号である。このようにして，より大きなバースト誤り訂正能力を持つ符号を作る方法を記号交鎖（symbol interleaving）と言い，パラメータiを交鎖の度数（degree of interleaving）と言う。

これまでに種々のバースト誤り訂正符号が研究されているが，その中でファイア符号（Fire code）は有名な符号である。これは，

$$g(X)=(X^{2b-1}-1)\cdot P(X) \qquad (5.4)$$

という形の多項式で生成される巡回符号であり，長さb以下の全てのバースト誤りの訂正が可能である。但し，P(X)はb次以上の既約多項式であり，しかも$X^{2b-1}-1$を割り切らないものとする。

例えば，GF(2)上の多項式

$$g(X)=(X^5-1)(X^3+X^2+1)$$
$$=X^8+X^7+X^5+X^3+X^2+1 \qquad (5.5)$$

は，2元（35, 27）ファイア符号を生成し，そのバースト誤り訂正能力はb=3である。この符号の復号器を図5.1に例示する。

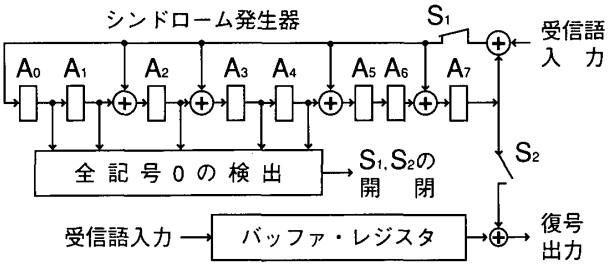


図5.1 (35,27)ファイア符号の複号器

この符号の復号処理は，次の3つの過程から成る。

(1) スイッチS_1を閉じ，スイッチS_2を開いた状態で，受信語$\upsilon(X)$を，その高次の項から順番に，シンドローム発生器のA_0～A_7とバッファ・レジスタの両方に，同時に入力する。

(2) バッファ・レジスタから$\upsilon(X)$の各記号を1記号ずつ順次読み出すと共に，受信語入力を0として，シンドローム発生器をシフトする。

(3) シンドローム発生器の左5桁A_0～A_4のすべてが0になれば，スイッチS_1を開き，S_2を閉じて，バッファ・レジスタから読み出される$\upsilon(X)$の各記号の訂正を逐次行う。

簡単のために，符号語$u(X)=0$を送信し，伝送途中に長さ3のバースト誤りが発生して$\upsilon(X)=X^{26}+X^{24}$を受信したとする。この受信語全部を複号器に入力し終ったときのシンドローム発生器の内容は，

$$(A_0\ A_1\cdots A_n)=(10101100) \qquad (5.6)$$

である。以後，入力を0として，これをシフトしたときの状態を表5.1に示す。この表のb_iは，各時点でバッファ・レジスタから読み出される受信記号，すなわち$\upsilon(X)$のX^iの項の係数b_iである。この表から分かるように，丁度$b_{26}=1$が読み出されるときに$(A_0, A_1,\cdots,A_4)$が(00000)にな

表5.1 長さ3のバースト誤りの訂正

b_i	A_0	A_1	A_2	A_3	A_4	A_5	A_6	A_7	
b_{34}	1	0	1	0	1	1	0	0	
b_{33}	0	1	0	1	0	1	1	0	
b_{32}	0	0	1	0	1	0	1	1	
b_{31}	1	0	1	0	0	0	0	0	
b_{30}	0	1	0	1	0	0	0	0	
b_{29}	0	0	1	0	1	0	0	0	
b_{28}	0	0	0	1	0	1	0	0	
b_{27}	0	0	0	0	1	0	1	0	
b_{26}	0	0	0	0	0	1	0	1	(訂正)
b_{25}	0	0	0	0	0	0	1	0	
b_{24}	0	0	0	0	0	0	0	1	(訂正)
b_{23}	0	0	0	0	0	0	0	0	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

り，このときの$A_7=1$が加算されて$b_{26}=1$が0に訂正される。$b_{24}=1$も同様にして0に訂正される。このような復号方法をエラー・トラップ復号（error-trap decoding）と言う。

GF(2)上のm次の多項式G(X)を考える。

$$G(X)=a_mX^m+\cdots+a_1X+a_0 \qquad (5.7)$$

ここに，a_m及びa_0を含めて3項以上の係数が1であるものとする。このとき，G(X)が割り切るX^n-1なる多項式が存在する。このような多項式X^n-1の中で，最小次数のものを，改めてX^n-1とする。その次数nを多項式G(X)が属するべき数（exponent）と言う。GF(2)上のいくつかの多項式と，それらが属するべき数を表5.2に例示する。

表5.2　GF(2)上の多項式G(X)とそれが属するべき数nの例

G(X)	n
X^2+X+1	3
X^3+X+1	7
X^3+X^2+1	7
X^3+X^2+X+1	4
X^4+X+1	15
X^4+X^2+1	6
X^4+X^2+X+1	7
X^4+X^3+1	15
X^4+X^3+X+1	6
$X^4+X^3+X^2+1$	7
$X^4+X^3+X^2+X+1$	5
⋮	⋮

m次の多項式G(X)が属するべき数nは，

$$m+1\leqq n\leqq 2^m-1 \qquad (5.8)$$

の範囲の自然数である。m次の多項式が既約であるとき，これが属するべき数nは，2^m-1であるか，その約数である。とくに，べき数2^m-1に属するm次の既約多項式を原始多項式（primitive polynomial）と言う。例えば，表5.2の多項式X^2+X+1，X^3+X+1，X^3+X^2+1，X^4+X+1，X^4+X^3+1などは原始多項式である。

GF(2)上の多項式の中で，n次未満のものは2^n個ある。これらの多項式の集合をAとするとき，Aは，法X^n-1に関する加算と乗算の下に環(ring)を成す。これは，法X^n-1に関する多項式の環である。この環Aのn個の多項式の集合

$$\{1,\ X,\ X^2,\cdots,X^{n-1}\} \qquad (5.9)$$

は線形独立であり，しかも，これらの多項式の線形結合により，Aのすべての元（多項式）を表すことができる。従って，この環Aは，n個の基底ベクトル（basis vector）が張るn次元のベクトル空間であり，しかも結合的代数（associative algebra）である。

G(X)は，式（5.7）に示した形のm次の多項式であるとし，G(X)が属するべき数をnとする。上述のn次元の結合的代数Aの元の中で，G(X)の倍数であるものすべての集合を$J(G)$とする。これはAの1つのイデアル（ideal）であり，しかも基底ベクトル

$$\{G(X),\ XG(X),\ X^2G(X),\cdots,X^{k-1}G(X)\}$$

が張るk次元の部分空間（k-dimensional subspace）である。ここに，

$$k=n-m \qquad (5.10)$$

である。このような部分空間$J(G)$は一種の線形(n,k)符号（linear (n,k) code）であり，G(X)をその生成多項式（generating polynomial），nを符号長（code length），kを次元（dimension）と言う。

以上の線形(n,k)符号$J(G)$の任意の元，すなわち，符号語は，

$$\left.\begin{aligned}\upsilon(X)&=a_{n-1}X^{n-1}+\cdots+a_1X+a_0\\ v&=(a_{n-1},\cdots,a_1,\ a_0)\\ &\qquad (a_i\in GF(2))\end{aligned}\right\} \qquad (5.11)$$

のように多項式$\upsilon(X)$で表すことも，また，その係数のn項組vで表すこともできる。

上式の符号語$\upsilon(X)$において，その各係数a_{n-1}，…，a_0を左に1桁だけ巡回的にシフトしたもの$\upsilon'(X)$は，

$$\begin{aligned}\upsilon'(X)&=a_{n-2}X^{n-1}+\cdots+a_0X+a_{n-1}\\ &=X\cdot\upsilon(X)-a_{n-1}\ (X^n-1)\end{aligned} \qquad (5.12)$$

のように表すことができる。この$\upsilon'(X)$はn次未満の多項式であるから結合的代数Aの元であり，しかもG(X)の倍数であるから，線形(n,k)符号$J(G)$の符号語である。すなわち，$J(G)$は巡回シフトの下に閉じている。このような線形符号$J(G)$を巡回(n,k)符号(cyclic (n,K) code)と言う。

巡回(n,k)符号を含めて，一般に線形(n,k)符号においては，任意に与えられたk個の情報記号に冗長性を加えて，n記号から成る符号語に作り上げる。これを符号化と言う。巡回(n,k)符号の場合，与えられたk個の情報記号$a_{n-1},\cdots,a_{n-k}$を符号化するのに，1つの方法として，k−1次の多項式

$$A(X)=a_{n-1}X^{k-1}+\cdots+a_{n-k-1}X+a_{n-k} \qquad (5.13)$$

に対して，$A(X)\cdot X^m$をG(X)で除算する方法がある。すなわち，

$$\left.\begin{aligned} A(X)\cdot X^m&=Q(X)G(X)+R(X)\\ A(X)\cdot X^m-R(X)&=Q(X)G(X)\end{aligned}\right\} \qquad (5.14)$$

のように符号化する。上式の$A(X)\cdot X^m-R(X)$はn次未満の多項式であり，しかもG(X)の倍数であるから，これは巡回(n,k)符号の符号語である。また，R(X)は，m=n−k次未満の多項式であるから，これを

$$R(X)=a_{n-k-1}X^{n-k-1}+\cdots+a_1X+a_0 \qquad (5.15)$$

とするとき，このように符号化して得られる符号語は，次式のように書くことができる。すなわち，

$$\left.\begin{aligned} \upsilon(X)&=A(X)\cdot X^m-R(X)\\ &=a_{n-1}X^{n-1}+\cdots+a_{n-k}X^{n-k}\\ &\quad+a_{n-k-1}X^{n-k-1}+\cdots+a_1X+a_0,\\ v&=(a_{n-1},\cdots,a_{n-k},\ a_{n-k-1},\cdots,a_0)\end{aligned}\right\} \qquad (5.16)$$

のようになる。その始めのk記号a_{n-1}〜a_{n-k}が情報記号であり，後のn−k(=m)記号は検査記号である。以上のように除算を用いて符号化した巡回(n,k)符号$J(G)$の各符号語には，与えられたk個の情報記号a_{n-1}〜a_{n-k}そのものが現れる。このような符号を組織符号(systematic code)と言う。

以上により，巡回(n,k)符号$J(G)$の符号長n，情報記号数(次元数)k，検査記号数(G(X)の次数)mの間に等式

$$n=k+m \qquad (5.17)$$

が成り立つ。ここに，

$$r=\frac{k}{n}\ ,\quad d=\frac{m}{n} \qquad (5.18)$$

を，それぞれ，符号化率(coding rate)及び冗長度(redundancy)と言う。

2元巡回(n,k)符号$J(G)$の符号語u(X)に部分的な擾乱を受けて受信語$\upsilon(X)$を得たとする。この受信語$\upsilon(X)$と送信された符号語u(X)の差

$$e(X)=\upsilon(X)-u(X) \qquad (5.18)$$

を誤り(error)と言う。符号語u(X)の各項が独立に擾乱を受けて生じるランダム誤り(random error)と，u(X)のある部分に集中的な擾乱を受けて生じるバースト誤り(burst error)が考えられる。ここでは，バースト誤りを訂正するための2元巡回(n,k)符号について検討する。

ランダム誤りの大きさを表すのにハミング距離(Hamming distance)が用いられる。これに対して，バースト誤りの大きさを表すのに，バースト長(burst length)を用いる。バースト誤りを多項式として扱うときには，それをe(X)のように書き，また，バースト誤りをn項組として扱うときにはeのように書くことにする。

例として，バースト誤りe=(01000001)を考える。以下に，このeとその巡回シフトを示す。

$e=$ (01000001)
(10000010)
(00000101) ＊
(00001010)
(00010100)
(00101000)
(01010000)
(10100000)

これらの8項組のそれぞれを，仮りに，8桁の2進数と見る。この時に最小の整数をもたらす8項組に＊を付してある。この8項組の右端の記号“1”から，最も左方に位置する記号“1”まで

の数字の個数3をバースト長とする。このバースト長は，この例の8個のバースト誤りすべてに適用する。

2元巡回符号$J(G)$により長さb以下の全てのバースト誤りの訂正が可能であり，長さb+1のバースト誤りの中に訂正不可能なものが1個でもあるとき，$J(G)$はバースト誤り訂正能力bを持つと言う。

$J(G)$の符号語$u(X)$に，あるバースト誤り$e(X)$が生じて得られる受信語

$$v(X)=u(X)+e(X) \tag{5.19}$$

を考える。法$G(X)$に関する$v(X)$の剰余$s(X)$を$v(X)$のシンドローム(syndrome)と言う。これは，法$G(X)$に関する$e(X)$の剰余であり，従って$e(X)$のシンドロームでもある。

$J(G)$において，長さb以下のすべてのバースト誤りのシンドロームがことごとく相異するとき，長さb以下のすべてのバースト誤りの訂正が可能である。更に，長さb+1以上のバースト誤り$e_1(X)$のシンドローム$s_1(X)$が長さb+1以下の別のバースト誤り$e_2(X)$のシンドロームに一致するとき，バースト誤り$e_1(X)$または$e_2(X)$が訂正不可能であり，$J(G)$のバースト誤り訂正能力はbである。

$J(G)$の生成多項式$G(X)$の次数mが2b未満である場合，長さb以下の2つの基本バースト$e_{01}(X)$と$e_{02}(X)$を用いて，$G(X)$を次のように表すことができる。

$$G(X)=e_{01}(X)\cdot X^c-e_{02}(X) \tag{5.20}$$

これは，長さb以下の2つのバースト誤り$e_{01}(X)\cdot X^c$と$e_{02}(X)$の両シンドロームが一致することを意味する。従って，$J(b)$がバースト誤り訂正能力bを持つためには，その生成多項式$G(X)$の次数mが2b以上でなくてはならない。すなわち，

$$m\geqq 2b \tag{5.21}$$

であることが必要である。

このため，巡回(n,k)符号$J(G)$がバースト誤り訂正能力bを持つなら，この符号の情報記号数k=n−mは，不等式

$$k\leqq n-2b \tag{5.22}$$

によって制限され，これにより不等式

$$r+2\frac{b}{n}\leqq 1,\ (\text{但し，}r=\frac{k}{n}) \tag{5.23}$$

を得る。ここに，rは$J(G)$の符号化率である。また，b/nを正規化バースト誤り訂正能力と言うことにする。

式(5.21)は，多項式$G(X)$で生成される2元巡回符号がバースト誤り訂正能力bを持つための必要条件であるが，十分条件ではない。実際，2b次以上の多項式で生成される符号で，バースト誤り訂正能力がb未満であるものは多数ある。

他方，バースト誤り訂正能力bを持つ2元巡回符号の生成多項式の中で，最小の次数で，しかも最も簡単なもの（項数の少ないもの）は，

$$G(X)=X^{2b}+X^b+1 \tag{5.24}$$

である。

バースト誤り訂正符号としては，符号化率$r(=k/n)$と共に正規化バースト誤り訂正能力b/nの値の大きいものが望まれる。しかし，式(5.23)は$r+2b/n$の値の上限を与え，rとb/nが一種のトレード・オフの関係を示している。

m次の多項式$G(X)$で生成される2元巡回符号$J(G)$がバースト誤り訂正能力bを持ち，そのmとbが不等式

$$m\geqq 2b+1 \tag{5.25}$$

を満たす場合を考える。これらの符号の中で符号化率rが最小のものは，次式の形の多項式$G(X)$で生成される符号である。

$$G(X)=(X^{m-b}+1)\cdot e_0(X). \tag{5.26}$$

ここに，$e_0(X)$は長さb+1の基本バーストであり，しかも多項式$X^{m-b}+1$を整除するものとする。但し，このような条件を満たす基本バースト$e_0(X)$は，mとbの値によって存在する場合と存在しない場合がある。

このような多項式$G(X)$で生成される符号$J(G)$の符号長は

$$n_0=2\ (m-b) \tag{5.27}$$

である。生成多項式の次数とバースト誤り訂正能

力の差m−bの値が等しい符号の集合をS(m−b)とする。このS(m−b)の任意の符号の符号長をnとするとき，

$$n \geqq n_0 \ (= 2(m-b)) \tag{5.28}$$

なる不等式が成り立つ。すなわち，式（5.26）のG(X)で生成される符号は，集合S(m−b)の中で最小の符号長（n_0）の符号である。上の不等式にm=n−kを適用して，不等式

$$r + \frac{b}{n} \geqq \frac{1}{2} \tag{5.29}$$

を得る。

2元巡回符号$J(G)$のそれぞれに対して正規化バースト誤り訂正能力b/nと符号化率rに着目するとき，各符号は（b/n，r）平面上の点で表すことができる。これらの符号点は，図5.2に示すように，式（5.23），式（5.29），ならびにr軸によって限られた三角形の内部に分布する。

以上により，符号化率がrであり，正規化バースト誤り訂正能力がb/nである2元巡回符号$J(G)$を

$$f = r + 2\frac{b}{n} \ (\leqq 1) \tag{5.30}$$

なるfの値で評価することができる。

表5.2に1部例示するように，G(X)の次数を3，4，…と順次大きくして，それらの多項式で生成される巡回符号の符号点（b/n，r）約3万個を図5.2の中にプロットした。これらの符号点の分布には制約があるが、多くは分布制約領域内の上部に集中する。

図5.2の横座標b/nに代えて，新しく

$$g = \frac{2}{1-r} \cdot \frac{b}{n} \ (\leqq 1) \tag{5.31}$$

を用い，直角座標平面上の点（g,r）で以って符号$J(G)$を表すことにすれば，図5.3に示すように，g＝0，g＝1，r＝1の3直線と，

$$r = 1 + \frac{1}{g-2}, \ (0 \leqq g \leqq 1) \tag{5.32}$$

なる直角双曲線に囲まれた範囲内に符号点が分布することになる。更に，図5.3のr軸の下部を圧縮し，上部を伸長する。このように修正した符号点分布を図5.4に示す。

以上のg−r平面における符号点の分布には，次のような特性が見られる。

(a) 符号点のg座標が離散的であり，多数の符号点が垂直線上に並ぶ。

(b) それぞれの垂直線上に分布する符号点は，各上限曲線に近い所に集中する。

(c) 符号点が，バースト誤り訂正能力bの値によってグループ化される。グループによって，rの上限曲線が異なる。

(d) rの上限曲線が右下りであり，rとgの間にも，一種のトレード・オフの関係がある。

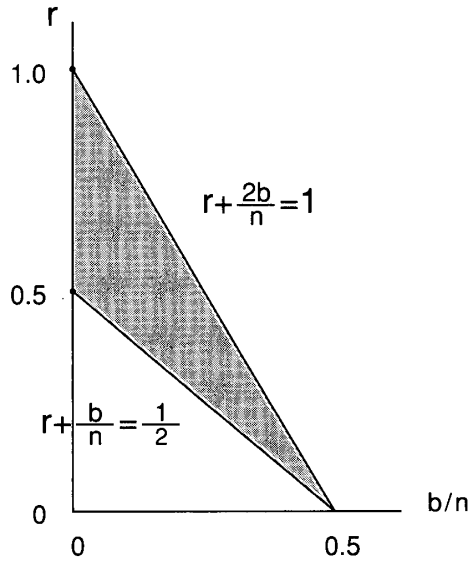


図5.2　符号点（b/n,r）の分布範囲

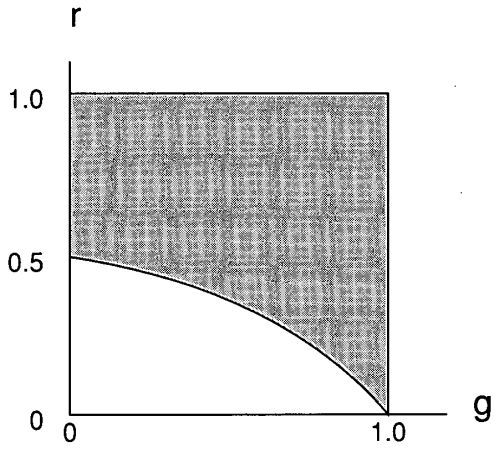


図5.3　符号点（g,r）の分布範囲

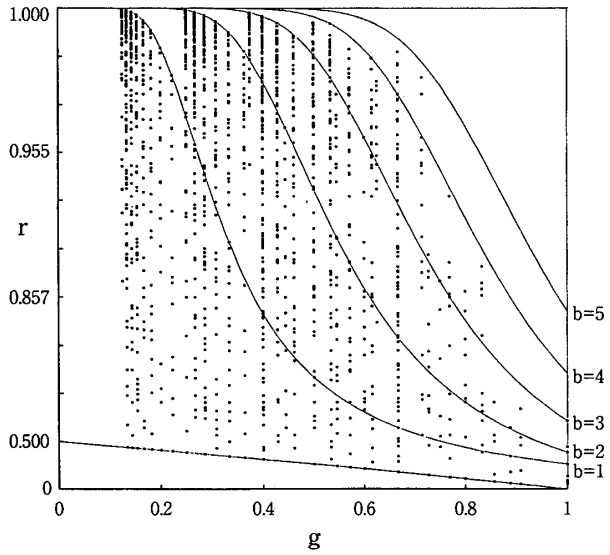


図5.4　符号点分布の拡大図

6．畳込み符号

これまでに述べた符号は，すべてブロック符号（block code）である。これに対して，近年，衛星通信や移動通信に畳み込み符号（convolutional code）が広く使われるようになった。畳込み符号においても，符号化すべき情報記号の系列を，順次，k_0記号の組に分け，その各組にr_0個の検査記号を付け加えて，合計

$$n_0=k_0+r_0 \tag{6.1}$$

個の記号の符号ブロック(code block)を構成する。

ここに，各符号ブロックの検査記号は，そのブロックを含めて，先行する合計m個の符号ブロックの情報記号の線形結合で決める。このため，図6.1に示すように，第1～第m符号ブロックが畳込み符号の1つの符号語を構成し，第2～第m＋1符号ブロックが次の符号語を構成し，等々して，図6.1に例示するように，順次，m－1符号ブロックを重複させて符号語を構成する。

このような符号を一般に，(mn_0, mk_0)畳込み符号と言う。畳込み符号もまた，その符号化及びパリティ検査を，それぞれ，生成行列G及びパリティ検査行列で規定することができる。

以下に（3×4，3×3）畳込み符号の生成行列Gと検査行列Hを例示する。各符号ブロックは，$k_0=3$個の情報記号と$r_0=1$個の検査記号から成り，各符号語はm＝3個の符号ブロックから成る。

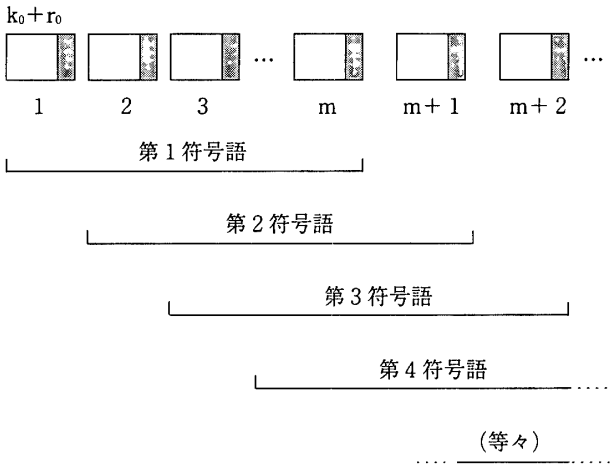


図6.1　符号ブロックの畳込み構造

$$G=\begin{bmatrix}
1&0&0&1&0&0&0&1&0&0&0&1\\
0&1&0&1&0&0&0&0&0&0&0&1\\
0&0&1&1&0&0&0&1&0&0&0&0\\
0&0&0&0&1&0&0&1&0&0&0&1\\
0&0&0&0&0&1&0&1&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&1\\
0&0&0&0&0&0&0&0&1&0&0&1\\
0&0&0&0&0&0&0&0&0&1&0&1\\
0&0&0&0&0&0&0&0&0&0&1&1
\end{bmatrix} \tag{6.1}$$

$$H=\begin{bmatrix}
1&1&1&1&0&0&0&0&0&0&0&0\\
1&0&1&0&1&1&1&1&0&0&0&0\\
1&1&0&0&1&0&1&0&1&1&1&1
\end{bmatrix} \tag{6.2}$$

この畳込み符号の符号語を

$$u=(a_1\,a_2\,a_3\,a_4, a_5\,a_6\,a_7\,a_8, a_9\,a_{10}\,a_{11}\,a_{12}) \tag{6.3}$$

とするとき，$a_1\,a_2\,a_3, a_5\,a_6\,a_7, a_9\,a_{10}\,a_{11}$の9記号が情報記号であり，$a_4$，$a_8$，$a_{12}$の3記号が検査記号である。上の生成行列$G$，あるいは検査行列$H$は，等式

$$\left.\begin{aligned}
&a_1+a_2+a_3+a_4=0\\
&a_1+a_3+a_5+a_6+a_7+a_8=0\\
&a_1+a_2+a_5+a_7+a_9+a_{10}+a_{11}+a_{12}=0
\end{aligned}\right\} \tag{6.4}$$

にもとづいて，3個の検査記号a_4，a_8，a_{12}が決め

られ，また受信語の検査が行われることを示している。

この符号においては，1つの符号語，すなわち，3符号ブロックを受信した時点で，その先頭の1符号ブロックを復号する。上の式（6.4）から分かるように，第2と第3の符号ブロックが正しく受信される限りにおいて，第1符号ブロック内の1誤りの訂正が可能である。

畳込み符号は，もともと，バースト誤りを訂正するための符号として考案されたが，ランダム誤りを訂正するための畳込み符号も各種開発されている。また，その復号方法としても，シンドローム・パターン復号法や多数決論理復号法の他に，畳込み符号独特の復号法が種々考案されている。

参考文献

1）R. W. Hamming, "Error-Detecting and Error-Correcting Codes,"B.S.T.J.,29, (1950)

2）T. Kasami, "Optical Shortened Cyclic Codes for Burst-Error-Correction,"IEEE Trans., IT－9, (1963)

3）T. Kasami, "A Decoding Procedure for Multiple Error-Correcting Cyclic Codes,"IEEE Trans., IT－10, (1967)

4）W. W. Peterson and E. J. Weldon Jr., "Error Correcting Codes," 2 nd ed. MIT-Press, (1972)

5）宮川洋，岩垂好裕，今井秀樹，「符号理論」昭晃堂(1973)

6）嵩忠雄，都倉信樹，岩垂好裕，稲垣康善，「符号理論」コロナ社（1975)

7）今井秀樹，「符号理論」，（電子情報通信学会）コロナ社（1990)

8）島田良作，「バースト誤り訂正2元巡回符号の符号点分布」，徳島大学工学研究報告，第45（2000)

（島田　良作：四国大学　非常勤講師）