

# 卒業論文

( 題 目 )

## 高次元パリティ符号の 構造と伝送順序に関する検討

指導教官

中野良平  
内匠逸

教授

助教授

知能情報システム学科 情報科学講座

平成 10 年 4 月 入学 ( 10117908 )

( 氏名 )

岩田浩明

( 平成 14 年 2 月 12 日提出 )



## 目次

1	はじめに	1
2	通信系のモデルと誤り訂正符号について	2
2.1	通信系のモデル	2
2.1.1	情報源	2
2.1.2	符号化器	3
2.1.3	通信路	3
2.1.4	復号器	4
2.2	誤り検出と訂正の原理	4
2.2.1	ハミング距離	4
2.2.2	最小距離	4
2.2.3	ハミング重み	5
2.2.4	誤り訂正の原理	5
2.2.5	誤り検出の原理	6
2.3	単一パリティ検査符号	6
3	高次元パリティ符号の構造と基本的な性質	8
3.1	2次元パリティ符号の構成	8
3.2	高次元パリティ符号の構成	9
3.3	高次元パリティ符号の基本的な性質	12
4	伝送順序と復号法	14
4.1	伝送順序の計算	14
4.2	復号法	20
5	実験	22
5.1	ハミング距離の分布	22
5.2	誤りを入れた符号語からのハミング距離	23
5.3	新しい伝送順序	23
5.3.1	計算量の低減	24
5.3.2	新しい訂正能力の評価法	24
5.3.3	得られた伝送順序	26
5.3.4	適応閾値判定法での比較	26
6	まとめ	30
	謝辞	31
	参考文献	32

## 図目次

2.1	通信系のモデル	2
2.2	最小距離と誤り訂正能力	5
2.3	最小距離と誤り検出能力	6
3.1	2次元パリティ符号の構成 ( $m = 4$ )	8
3.2	3D <sub>m3</sub> 符号の構造と伝送順序	9
3.3	$r_0-r_1, r_1-r_2, r_2-r_0$ 平面の2次元パリティ符号を用いた, 3次元パリティ符号 (サイズ $m$ ) の分割表現	10
3.4	4D <sub>m3</sub> 符号の構造と伝送順序	11
4.1	2次元パリティ符号 ( $m = 4$ ) の伝送順序 $o$ と座標 $(r_0, r_1)$ との関係	15
4.2	3次元パリティ符号 ( $m = 3$ ) の伝送順序 $o$ と座標 $(r_0, r_1, r_2)$ との関係	16
4.3	4次元パリティ符号 ( $m = 3$ ) の伝送順序 $o$ と座標 $(r_0, r_1, r_2, r_3)$ との関係	18
4.4	$n$ 次元パリティ符号における, 各ビットを貫く $n$ 本の検査線	20
5.1	[例] $r_0$ 方向の2ビット目と3ビット目を入れ換える	25
5.2	訂正能力の評価法	25
5.3	これまでに見つけた中で最も訂正能力の高い伝送順序 (一例)	27
5.4	バースト中の誤り率 0.2	28
5.5	バースト中の誤り率 0.4	28
5.6	バースト中の誤り率 0.6	28
5.7	バースト中の誤り率 0.8	28
5.8	バースト中の誤り率 1.0	29

## 表目次

3.1	$n$ 次元トラスパリティ符号の性質 ( $n = 2 \sim 6, m = 3 \sim 8$ ) . . . . .	13
4.1	2次元パリティ符号 ( $m = 4$ ) の伝送順序 $o$ と座標 $(r_0, r_1)$ との関係 . . . . .	16
4.2	3次元パリティ符号 ( $m = 3$ ) の伝送順序 $o$ と座標 $(r_0, r_1, r_2)$ との関係 . . . . .	17
4.3	4次元パリティ符号 ( $m = 3$ ) の伝送順序 $o$ と座標 $(r_0, r_1, r_2, r_3)$ との関係 . . . . .	19
5.1	$3D_{m3}$ 符号の距離分布 . . . . .	22
5.2	$4D_{m3}$ 符号の距離分布 . . . . .	22
5.3	ランダム誤りの訂正能力 . . . . .	23
5.4	バースト誤りの訂正能力 . . . . .	23
5.5	ランダム誤りの訂正能力 (2) . . . . .	26
5.6	バースト誤りの訂正能力 (2) . . . . .	26
5.7	図 5.3 の伝送順序の訂正能力 . . . . .	26



## 第 1 章

### はじめに

近年，極めて膨大な情報が，デジタル化されており，今後もデジタル通信の発達はさらに急激に進むことが予想される．これに伴い，通信品質の向上や，記録媒体の信頼性確保等のために，誤り訂正符号の研究（符号理論の研究）と，産業界への応用がますます期待されている．また，携帯電話の消費電力を抑えるための手段としても，誤り訂正符号は重要視されている．

従来の誤り訂正符号では，平均ビット誤り率 (BER : bit error rate) が  $10^{-3}$  程度までの動作が一般的であり，より劣悪な BER においては，復号によりかえって BER が増大する傾向にあった．しかし，本符号は，BER が  $10^{-1} \sim 10^{-2}$  程度の劣悪な環境においても良好な誤り訂正能力を有していることが知られている．水平-垂直パリティ符号に相当する 2 次元パリティ符号をもとに，順次次元拡張することによって，より高い次元のパリティ符号を段階的に構成することができる．

一般に，誤り訂正符号の誤り訂正能力  $t$  は最小距離  $d_{min}$  から  $t \leq \frac{d_{min}-1}{2}$  という式で求まる．しかし高次元パリティ符号では特定の誤りの時，誤り訂正能力を越えて誤りを訂正できるものが存在する．このことより符号語の分布が不均等になっているのではないかと考えられる．

本論文では，高次元パリティ符号の構造，及び誤り訂正能力が良くなっている理由について考える．また，その理由の一つと考えられる伝送順序について検証する．そして，従来のものより訂正能力の良いものが無いか探し，その違いを比較検討する．

## 第 2 章

### 通信系のモデルと誤り訂正符号について

本章では，次章以降に必要な前提知識となる，通信系のモデルや，誤り訂正符号の概念等について説明を行う．

#### 2.1 通信系のモデル

本節では，図 2.1 を使って，符号理論における通信系のモデルを説明する．ここで，通信とは，相手に情報を送ったり，情報を記録したりすることを指す．

##### 2.1.1 情報源

情報源とは，送信しようとする情報の発生源のことであり，この情報源からは，‘0’ または ‘1’ のビット列 (情報ビット列) が発生する．符号化器では，情報ビット列を， $k$  ビットのブロック毎に区切って取り扱う．区切られたビット列を，それぞれ，通報と呼び，

$$\mathbf{i} = (i_1, i_2, \dots, i_k) \quad (2.1)$$

と表すことにする．

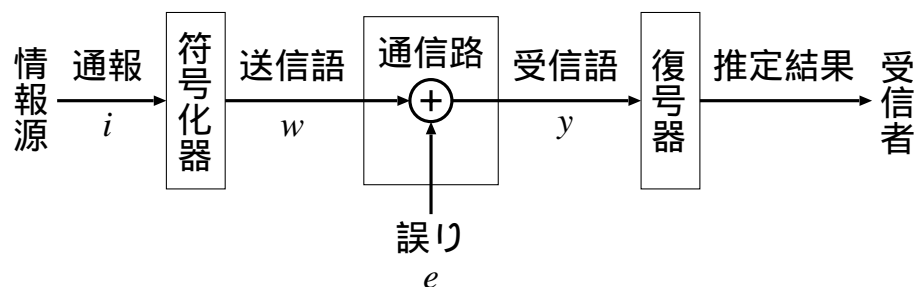


図 2.1: 通信系のモデル



### 2.1.2 符号化器

符号化器は、通報  $i$  に対応した  $n$  ビット (ただし  $n > k$ ) のビット列

$$\mathbf{w} = (x_1, x_2, \dots, x_n) \quad (2.2)$$

を出力する。すなわち、符号化器では、 $n - k$  ビットの余分な (冗長な) ビットが付加されることになる。これを、符号化と呼ぶ。

なお、 $n$  を符号長といい、 $k$  を情報ビット数という。また、符号化器から出力される  $w$  を符号語と呼ぶ。なお、特定の通報  $i$  を問題にすると、通信路に送り出された符号語を、送信語ともいう。

実際の伝送や、記録媒体では、送信語の各ビット  $x_i$  に対し、0 の場合はパルス無し、1 の場合はパルスあり、といった信号の波形として送り出されるが、具体的な実現方法に関しては、ここでは考えない。

さて、符号長  $n$ 、情報ビット数  $k$  の符号を  $(n, k)$  符号という。また、符号長と情報ビット数の比

$$R = \frac{k}{n} \quad (2.3)$$

を、符号化率または伝送速度と呼ぶ。これは、符号の能率を表す量であり、一般に符号化率の高い符号ほど符号語の中の冗長ビットの割合が少なくなり伝送の効率が良い。しかし、情報ビットの割合が増えるほど符号語の種類が多くなり最小距離が小さくなる。つまり、誤り訂正能力が低下する。逆に、符号化率が低いと誤り訂正能力は高くなるが符号語の中の冗長ビットの割合が多くなり伝送の効率が悪くなる。

### 2.1.3 通信路

通信路では、送信語  $w$  が入力されると、 $n$  ビットの受信語

$$\mathbf{y} = (y_1, y_2, \dots, y_n) \quad (2.4)$$

を出力する。

伝送・記録媒体の雑音等の影響が無ければ、送信語  $w$  と同じビット列が受信されることになるが、通常は、雑音などのため、ある確率で、送信語の各ビットの 0, 1 が異なって受信されてしまう。これを、誤りという。ここでは、各ビットの誤りを  $e_i$  とし、ビット列

$$\mathbf{e} = (e_1, e_2, \dots, e_n) \quad (2.5)$$

を誤りパターンとすると，受信語  $y$  は，

$$y = w \oplus e \quad (2.6)$$

と表すことができる．ただし， $e_i \in \{0,1\}$  であり， $\oplus$  は，排他的論理和を表す．

### 2.1.4 復号器

復号器では，受信語  $y$  をもとにして，いずれの符号語が送信されたかを推定し，送信語  $w$  の推定値，または，通報  $i$  の推定値を得る．推定値を得る際，用いられる方法を，復号アルゴリズムまたは復号法と呼び，通信路で発生した誤りを訂正したり検出したりする．

## 2.2 誤り検出と訂正の原理

本節では，ハミング距離，ハミング重み，最小距離という概念について説明した後，最小距離が誤り検出と訂正に，どのような影響を及ぼすのかについて述べる．

### 2.2.1 ハミング距離

2つのビット列  $u, v$  の，それぞれ対応する位置のビットが異なっている箇所の数を，ハミング距離といい， $d(u, v)$  で表す．ハミング距離は，いわゆる距離としての性質を備えており，次の3つの距離の公理を満たす．

任意のビット列  $u, v, w$  に対し，

1.  $u = v$  のとき  $d(u, v) = 0$   
 $u \neq v$  のとき  $d(u, v) > 0$
2.  $d(u, v) = d(v, u)$
3.  $d(u, v) + d(v, w) \geq d(u, w)$  (三角不等式)

### 2.2.2 最小距離

$n$  ビットのビット列のうち，符号語として  $w_1, w_2, \dots, w_M$  の  $M$  種類を用いるとする．このとき，相異なる符号語間のハミング距離を求め，その最小値を，符号の最小距離といい， $d_{\min}$  と表す．すなわち，

$$d_{\min} = \min_{i \neq j, i, j=1, \dots, M} d(w_i, w_j) \quad (2.7)$$

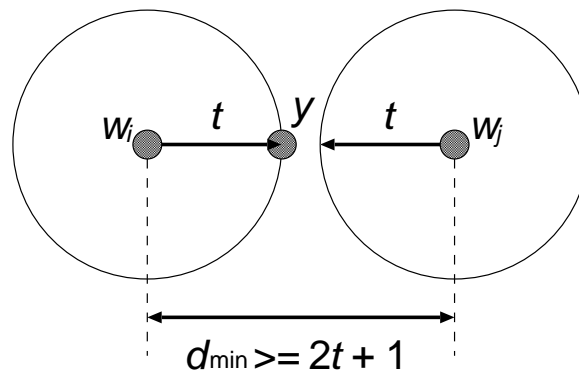


図 2.2: 最小距離と誤り訂正能力

よって、任意の、相異なる符号語同士は、 $d_{\min}$  か所以上のビットで異なっていることになる。

### 2.2.3 ハミング重み

ビット列  $u$  の非零のビットの数をハミング重みと呼び、 $w(u)$  で表す。すなわち、 $0$  を、全零のビット列とすると、

$$w(u) = d(0, u) \quad (2.8)$$

である。

### 2.2.4 誤り訂正の原理

符号語  $w$  から、ハミング距離  $t$  以下のビット列の集合を、 $w$  を中心とする半径  $t$  の小球で表すことにする。

各符号語を中心とする半径  $t$  の小球は、

$$d_{\min} \geq 2t + 1 \quad (2.9)$$

であれば、共通部分を持たない。送信語として、 $w$  を送り、 $t$  か所以下の誤り ( $t$  重誤りと呼ぶ) を生じて  $y$  が受信されたとする。すなわち、

$$y = w + e, w(e) \leq t \quad (2.10)$$

のとき、 $y$  と  $w$  の間のハミング距離は  $t$  以下であるから、 $y$  に最も近い符号語は、 $w$  である。よって、受信側で、最も近い符号語である  $w$  に復号すれば、この誤りは訂正できることになる。

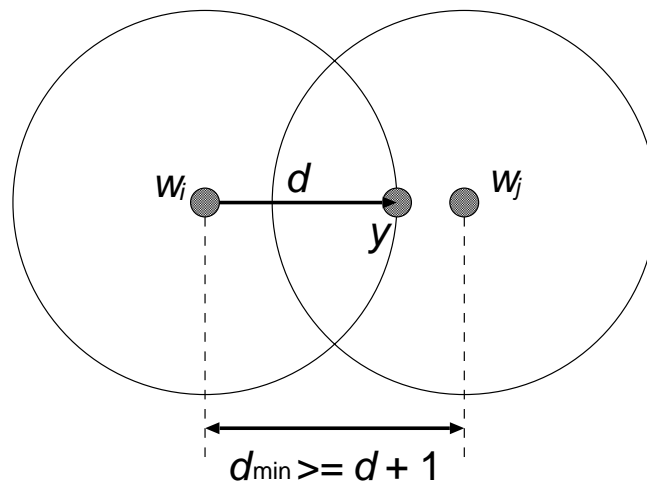


図 2.3: 最小距離と誤り検出能力

このように，任意の  $t$  重誤りを訂正できる符号のことを， $t$  重誤り訂正符号という．なお， $t = 1$  の符号を，単一誤り訂正符号という．

さて，以上のように，符号語間のハミング距離が離れていることが，誤り訂正のできる基本となる．

### 2.2.5 誤り検出の原理

次に，誤り検出について考える．受信語が，符号語としてあり得ないものならば，誤りが発生したと判断できる．これが，誤り検出である．

送信語  $w$  に対して， $d$  個以下の誤り ( $d$  重誤り) が加わり，受信語  $y$  になったとする．このとき， $y$  は， $w$  を中心とする半径  $d$  の小球内にある．この半径  $d$  の小球は，

$$d_{\min} \geq d + 1 \quad (2.11)$$

ならば，その中に他の符号語はない．したがって，この  $y$  が他の符号語と一致することなく，受信側では誤りとして検出できる．このように，最小距離  $d_{\min}$  の符号を使って誤り検出を行う場合，式 (2.11) を満足する  $d$  個以下の全ての誤りを検出できる．

なるべく多くの個数の誤りまで検出するためには，誤り訂正の場合と同様に，最小距離  $d_{\min}$  が大きいことが必要になってくる．

## 2.3 単一パリティ検査符号

次章で説明するように，高次元パリティ符号は，単一パリティ検査符号を基本として構成されている．そこで，本節では，この単一パリティ検査符号について説明する．

長さ  $k$  の情報ビット列 (すなわち通報) を

$$\mathbf{i} = (i_1, i_2, \dots, i_k) \quad (2.12)$$

とする。このとき、この情報ビット列に、1 ビット余分なビット  $c$  を付け加えて、符号長  $n = k + 1$  の符号語

$$\mathbf{w} = (i_1, i_2, \dots, i_k, c) \quad (2.13)$$

を作る。このとき、ビット  $c$  は、符号語内のビット '1' の個数が偶数になるように決める。なお、 $c$  の値は、

$$\begin{aligned} c &= i_1 \oplus i_2 \oplus \dots \oplus i_k \\ &= (i_1 + i_2 + \dots + i_k) \bmod 2 \end{aligned} \quad (2.14)$$

で求めることもできる。

例えば、長さ 7 の情報ビット列 (0110010) を、単一パリティ検査符号で符号化すると、符号語は (01100101) となる。

ビット '1' の個数が偶数か奇数かをパリティと呼び、ビット  $c$  は、情報を担っておらず、パリティを検査するためだけに余分に付加されたビットなので、これをパリティビットと呼ぶ。式 (2.13) の符号は、パリティを検査するビットが 1 つなので、単一パリティ検査符号と呼ばれる。

いま、符号語  $w$  に誤りが生じて、符号語内に含まれるビット '1' の個数が、仮に奇数個になったとすると、受信側では、符号語として使われるはずのないパターンを受け取ることになるので、送信された符号語 (送信語) に誤りが生じていることが分かる。

誤りが 1 ビットであれば、誤りが生じていることを必ず検出できるので、このような符号を、単一誤り検出符号という。単一パリティ検査符号の最小距離  $d_{\min}$  は 2 なので、式 (2.11) より、この符号は、1 個の誤りを検出能力を持っていることを確認できる。

なお、単一パリティ検査符号の 2 つの符号語

$$\mathbf{w}_1 = (i_{11}, i_{12}, \dots, i_{1k}, c_1) \quad (2.15)$$

$$\mathbf{w}_2 = (i_{21}, i_{22}, \dots, i_{2k}, c_2) \quad (2.16)$$

の和 (排他的論理和)

$$\mathbf{w}_1 \oplus \mathbf{w}_2 = (i_{11} \oplus i_{21}, i_{12} \oplus i_{22}, \dots, i_{1k} \oplus i_{2k}, c_1 \oplus c_2) \quad (2.17)$$

も、やはり符号語になっている。このように、任意の 2 つの符号語の和がやはり符号語となるような符号のことを、線形符号と呼ぶ。また、線形符号においては最小ハミング距離と最小ハミング重みは等しくなっている。

## 第 3 章

### 高次元パリティ符号の構造と基本的な性質

高次元パリティ符号は、水平垂直パリティ符号を次元拡張して得られるブロック符号である。本章では、高次元トラスパリティ符号の構造と、基本的な性質について述べる。

#### 3.1 2次元パリティ符号の構成

符号長  $m^2$  の 2 次元パリティ符号は、図 3.1 のように符号長  $m$  の  $(m, m-1)$  パリティ符号を  $r_1$  という新しい方向に順次 1 ビットずつ巡回シフトさせながら、 $m-1$  個交錯させ、 $m$  番目に  $r_1$  方向の偶数パリティ検査を満たすすべてのビットがパリティビットであるような単一パリティ検査符号を配置する。なお、巡回シフトにおいて、パリティビットの位置は変更しない。また、伝送方向は、一度使った検査線をできるだけ避けるような斜め方向になっている。

以後、高次元パリティ符号の説明においては、2次元パリティ符号を正方形に配列したときの一边のビット数をサイズと表記し、 $m$  で表すものとする。また、一边のサイズが  $a$  の  $n$  次元パリティ符号を、 $nDma$  符号と略記することもある。例えば、図 3.1 の

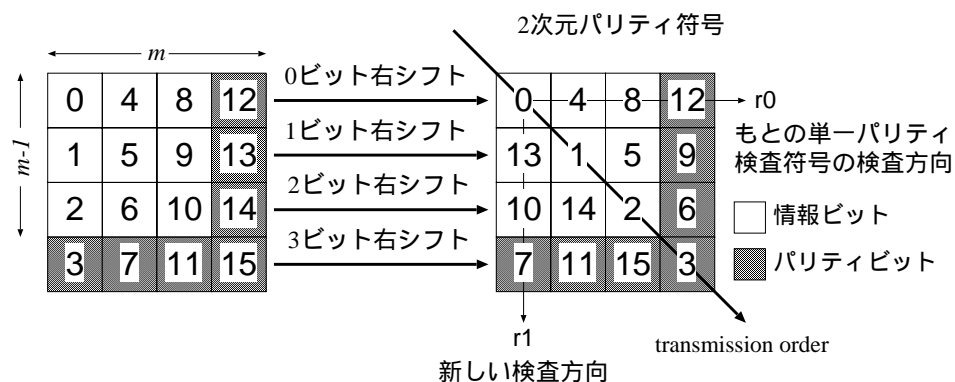
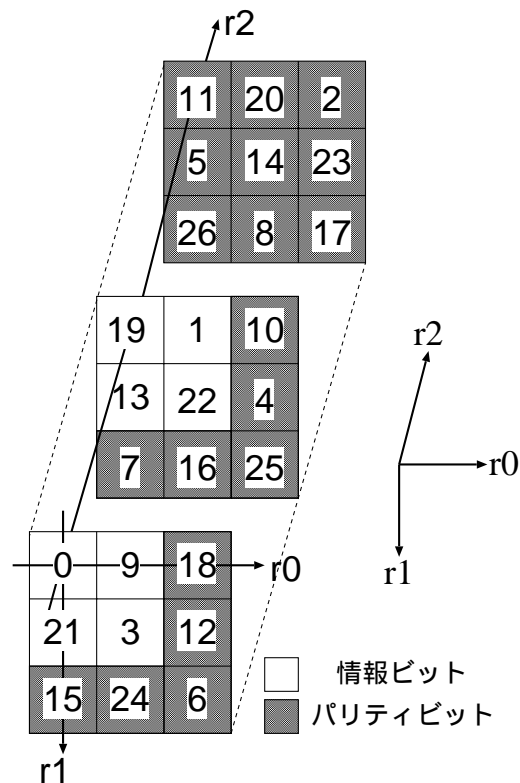


図 3.1: 2次元パリティ符号の構成 ( $m = 4$ )

図 3.2: 3D<sub>m</sub>3 符号の構造と伝送順序

場合，正方形の一辺は 4 ビットであるからサイズは 4 であり，2D<sub>m</sub>4 符号と表す．

### 3.2 高次元パリティ符号の構成

一般に，符号長  $m^n$  の  $n$  次元パリティ符号は，符号長  $m^{n-1}$  の  $n-1$  次元パリティ符号を拡張したものである．具体的には， $n-1$  次元パリティ符号が持つ検査線方向  $r_0, r_1, \dots, r_{n-2}$  に， $r_{n-1}$  という新たな方向の検査線を追加し，この方向に  $n-1$  次元パリティ符号を計  $m$  枚交錯することで構成される．交錯する理由は，バースト誤りをランダム化すべく，インターリーブング (interleaving) を行うためである．伝送順序の計算については次章で述べる．

以下では，3 次元パリティ符号と 4 次元パリティ符号の構成を例にとり説明する．

#### 【例】3 次元パリティ符号の構成

符号長  $m^3$  の 3 次元パリティ符号は，符号長  $m^2$  の 2 次元パリティ符号を拡張したものである．具体的には，2 次元パリティ符号が持つ互いに直交する 2 つの検査線方向  $r_0, r_1$  に対し，直交する方向に新たな検査線方向  $r_2$  を設け，この方向に 2 次元パリティ

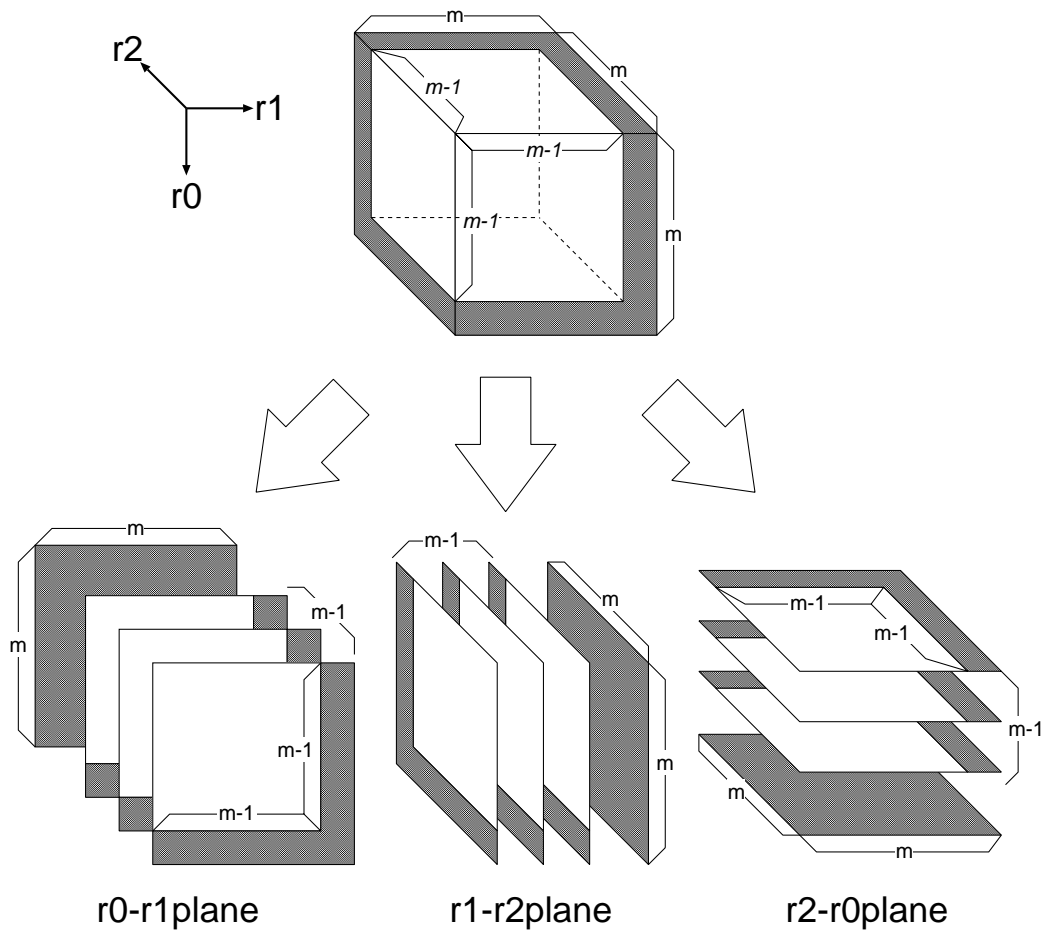
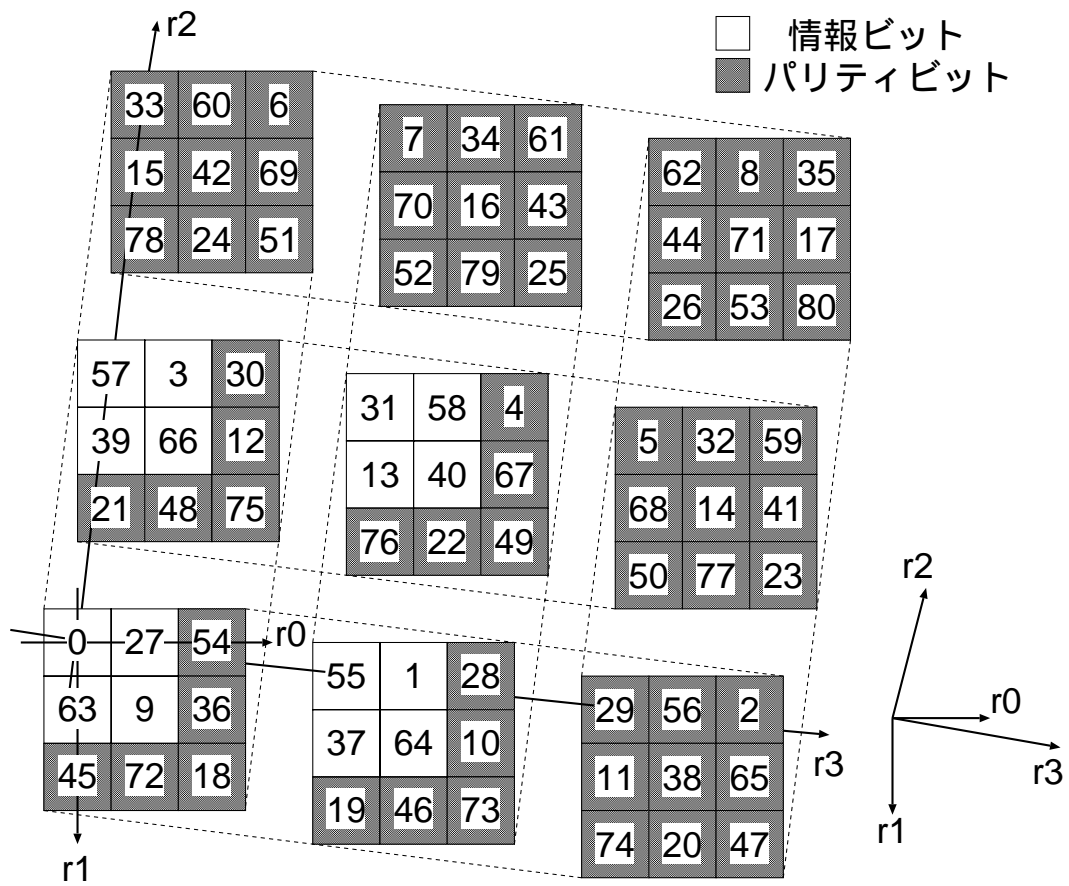


図 3.3:  $r_0$ - $r_1$ ,  $r_1$ - $r_2$ ,  $r_2$ - $r_0$  平面の 2 次元パリティ符号を用いた, 3 次元パリティ符号 (サイズ  $m$ ) の分割表現



図 3.4: 4D<sub>m3</sub> 符号の構造と伝送順序

符号を計  $m$  枚交錯することによって構成される。図 3.2 は、3D<sub>m3</sub> 符号の構造を示したものであり、各ビットに付された数字は伝送順序を表している。

図 3.2 では、 $r_0-r_1$  平面で構成される 2 次元パリティ符号に分割することによって表現しているが、図 3.3 に示すように、 $r_1-r_2$  平面と  $r_2-r_0$  平面で構成される 2 次元パリティ符号を使って表現することもできる。

#### 【例】4 次元パリティ符号の構成

符号長  $m^4$  の 4 次元パリティ符号は、符号長  $m^3$  の 3 次元パリティ符号を拡張したものである。具体的には、3 次元パリティ符号に対し、新たな方向 ( $r_3$  方向) に検査線を追加し、この方向に 3 次元パリティ符号を計  $m$  枚交錯することで構成される。図 3.4 は、4D<sub>m3</sub> 符号の構造と、伝送順序を示したものである。

### 3.3 高次元パリティ符号の基本的な性質

本節では、高次元トーラスパリティ符号の基本的な性質について示す。

#### 符号長と情報ビット数

サイズ  $m$  の  $n$  次元パリティ符号の符号長は  $m^n$ 、情報ビット数は  $(m-1)^n$  で与えられる。

#### 符号化率 (伝送速度)

$n$  次元パリティ符号の符号化率 (伝送速度)  $R$  は、(情報ビット数)/(符号長) すなわち次式で与えられる。

$$R = \frac{(m-1)^n}{m^n} = \left(1 - \frac{1}{m}\right)^n \quad (3.1)$$

#### 最小距離

$n$  次元パリティ符号の最小距離 (最小ハミング距離) は  $2^n$  である。

以上の性質を次元数  $n = 2 \sim 6$ 、サイズ  $m = 3 \sim 8$  の場合についてまとめておく (表 3.1)。

表 3.1:  $n$  次元トラスパリティ符号の性質 ( $n = 2 \sim 6, m = 3 \sim 8$ )

次元	サイズ	符号長	情報ビット数	符号化率
$n$	$m$	$m^n$	$(m-1)^n$	$\left(1 - \frac{1}{m}\right)^n$
2	3	9	4	0.444
2	4	16	9	0.562
2	5	25	16	0.640
2	6	36	25	0.694
2	7	49	36	0.735
2	8	64	49	0.766
3	3	27	8	0.296
3	4	64	27	0.422
3	5	125	64	0.512
3	6	216	125	0.579
3	7	343	216	0.630
3	8	512	343	0.670
4	3	81	16	0.198
4	4	256	81	0.316
4	5	625	256	0.410
4	6	1296	625	0.482
4	7	2401	1296	0.540
4	8	4096	2401	0.586
5	3	243	32	0.132
5	4	1024	243	0.237
5	5	3125	1024	0.328
5	6	7776	3125	0.402
5	7	16807	7776	0.463
5	8	32768	16807	0.513
6	3	729	64	0.088
6	4	4096	729	0.178
6	5	15625	4096	0.262
6	6	46656	15625	0.335
6	7	117649	46656	0.397
6	8	262144	117649	0.449

## 第 4 章

### 伝送順序と復号法

#### 4.1 伝送順序の計算

伝送順序  $o (= 0, 1, \dots, m^n - 1)$  は, パリティビットの計算が完了した符号内の各ビットを何番目に送り出すかを表すものである. 本符号では, バースト誤りを符号内で分散させるため, 擬巡回伝送と呼ばれる特殊な伝送順序を採用しており, これがインターリービング機能を果たしている. ここでは,  $o$  番目に伝送するビットの座標  $r_k$  ( $k = 0, 1, \dots, n - 1$ ) を求める式を次のように定める.

**【 $n$  次元パリティ符号 (サイズ  $m$ ) の伝送順序  $o$  に対する座標  $(r_0, r_1, r_2, \dots, r_{n-1})$ 】**  
 $n$  次元パリティ符号 (サイズ  $m$ ) の伝送順序  $o$  に対する座標  $(r_0, r_1, r_2, \dots, r_{n-1})$  は次の式で定義される.

$$\begin{aligned} r_0 &= \left( \lfloor o/m^{n-1} \rfloor + \lfloor o/m^{n-2} \rfloor + \lfloor o/m^{n-3} \rfloor + \dots + \lfloor o/m \rfloor + o \right) \bmod m \\ r_1 &= \lfloor o/m^{n-2} \rfloor \bmod m \\ r_2 &= \lfloor o/m^{n-3} \rfloor \bmod m \\ &\vdots \\ r_{n-2} &= \lfloor o/m \rfloor \bmod m \\ r_{n-1} &= o \bmod m \end{aligned}$$

すなわち

$$r_0 = \left( \sum_{i=0}^{n-1} \lfloor o/m^i \rfloor \right) \bmod m \quad (4.1)$$

$$\begin{aligned} r_{k'} &= \lfloor o/m^{n-(k'+1)} \rfloor \bmod m \quad (4.2) \\ &\quad (k' = 1, 2, \dots, n-2, n-1) \end{aligned}$$

である.

$r_0$  の計算に当たっては, 式 (4.2) を用いて  $r_{k'}$  ( $k' = 1, 2, \dots, n-2, n-1$ ) の値を計算す

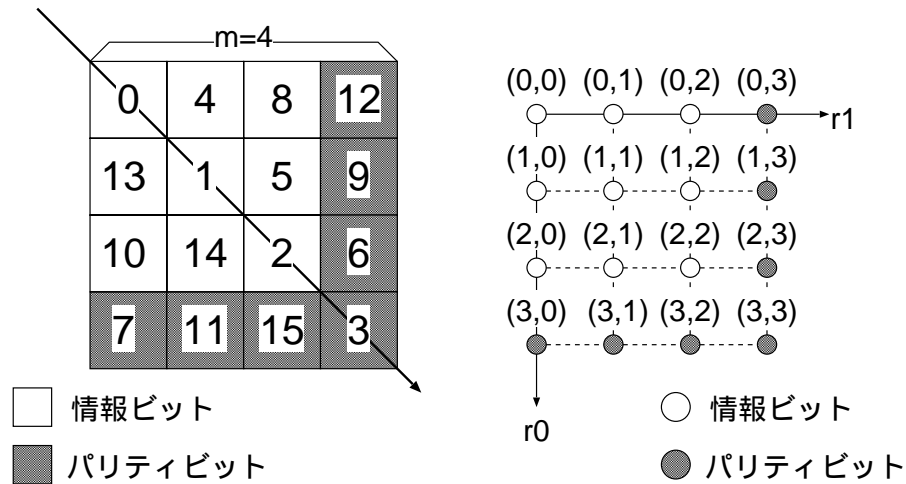


図 4.1: 2次元パリティ符号 ( $m = 4$ ) の伝送順序  $o$  と座標  $(r_0, r_1)$  との関係

ると同時に和をとっていき，最後に  $\lfloor o/m^{n-1} \rfloor$  を加えてもよい．つまり， $r_0$  については

$$r_0 = \left( \lfloor o/m^{n-1} \rfloor + \sum_{k'=1}^{n-1} r_{k'} \right) \bmod m \tag{4.3}$$

という式を用いて計算することもできる．

【例】2次元パリティ符号 ( $m = 4$ ) の伝送順序  $o$  に対応する座標  $(r_0, r_1)$

2次元パリティ符号の場合は，式(4.1)(4.2)において  $n = 2$  とすればよいので，

$$r_0 = (\lfloor o/m \rfloor + o) \bmod m \tag{4.4}$$

$$r_1 = o \bmod m \tag{4.5}$$

となる．図 4.1 は  $m = 4$  の場合を示したものである．このとき，式(4.4)(4.5)より導かれる式

$$r_0 = (\lfloor o/4 \rfloor + o) \bmod 4$$

$$r_1 = o \bmod 4$$

によって，表 4.1 のように伝送順序  $o$  から座標  $(r_0, r_1)$  を求めることができる．

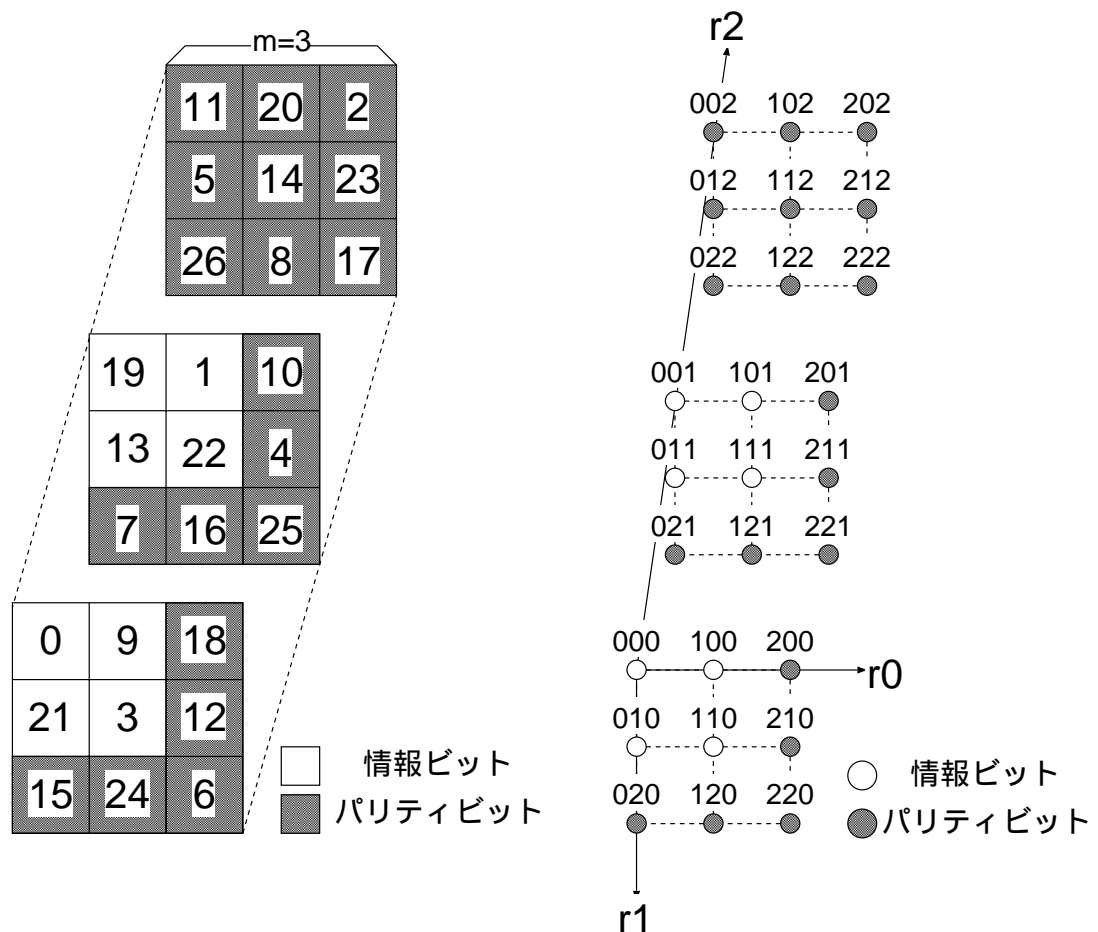
【例】3次元パリティ符号 ( $m = 3$ ) の伝送順序  $o$  に対応する座標  $(r_0, r_1, r_2)$

3次元パリティ符号の場合は，式(4.1)(4.2)において  $n = 3$  とすればよいので，

$$r_0 = (\lfloor o/m^2 \rfloor + \lfloor o/m \rfloor + o) \bmod m \tag{4.6}$$

表 4.1: 2 次元パリティ符号 ( $m = 4$ ) の伝送順序  $o$  と座標  $(r_0, r_1)$  との関係

$o$	$r_0$	$r_1$	$o$	$r_0$	$r_1$
0	0	0	8	2	0
1	1	1	9	3	1
2	2	2	10	0	2
3	3	3	11	1	3
4	1	0	12	3	0
5	2	1	13	0	1
6	3	2	14	1	2
7	0	3	15	2	3

図 4.2: 3 次元パリティ符号 ( $m = 3$ ) の伝送順序  $o$  と座標  $(r_0, r_1, r_2)$  との関係

$$r_1 = \lfloor o/m \rfloor \bmod m \quad (4.7)$$

$$r_2 = o \bmod m \quad (4.8)$$

となる．図 4.2 は  $m = 3$  の場合を示したものである．このとき，式 (4.6)(4.7)(4.8) より導かれる式

$$r_0 = (\lfloor o/9 \rfloor + \lfloor o/3 \rfloor + o) \bmod 3$$

$$r_1 = \lfloor o/3 \rfloor \bmod 3$$

$$r_2 = o \bmod 3$$

によって，表 4.2 のように伝送順序  $o$  から座標  $(r_0, r_1, r_2)$  を求めることができる．

表 4.2: 3 次元パリティ符号 ( $m = 3$ ) の伝送順序  $o$  と座標  $(r_0, r_1, r_2)$  との関係

$o$	$r_0$	$r_1$	$r_2$	$o$	$r_0$	$r_1$	$r_2$	$o$	$r_0$	$r_1$	$r_2$
0	0	0	0	9	1	0	0	18	2	0	0
1	1	0	1	10	2	0	1	19	0	0	1
2	2	0	2	11	0	0	2	20	1	0	2
3	1	1	0	12	2	1	0	21	0	1	0
4	2	1	1	13	0	1	1	22	1	1	1
5	0	1	2	14	1	1	2	23	2	1	2
6	2	2	0	15	0	2	0	24	1	2	0
7	0	2	1	16	1	2	1	25	2	2	1
8	1	2	2	17	2	2	2	26	0	2	2

【例】4 次元パリティ符号 ( $m = 3$ ) の伝送順序  $o$  に対応する座標  $(r_0, r_1, r_2, r_3)$

最後に，4 次元の場合を示す．4 次元パリティ符号の場合は，式 (4.1)(4.2) において  $n = 4$  とすればよいので，

$$r_0 = (\lfloor o/m^3 \rfloor + \lfloor o/m^2 \rfloor + \lfloor o/m \rfloor + o) \bmod m \quad (4.9)$$

$$r_1 = \lfloor o/m^2 \rfloor \bmod m \quad (4.10)$$

$$r_2 = \lfloor o/m \rfloor \bmod m \quad (4.11)$$

$$r_3 = o \bmod m \quad (4.12)$$

となる．図 4.3 は  $m = 3$  の場合を示したものである．このとき，式 (4.9)(4.10)(4.11)(4.12) より導かれる式

$$r_0 = (\lfloor o/27 \rfloor + \lfloor o/9 \rfloor + \lfloor o/3 \rfloor + o) \bmod 3$$

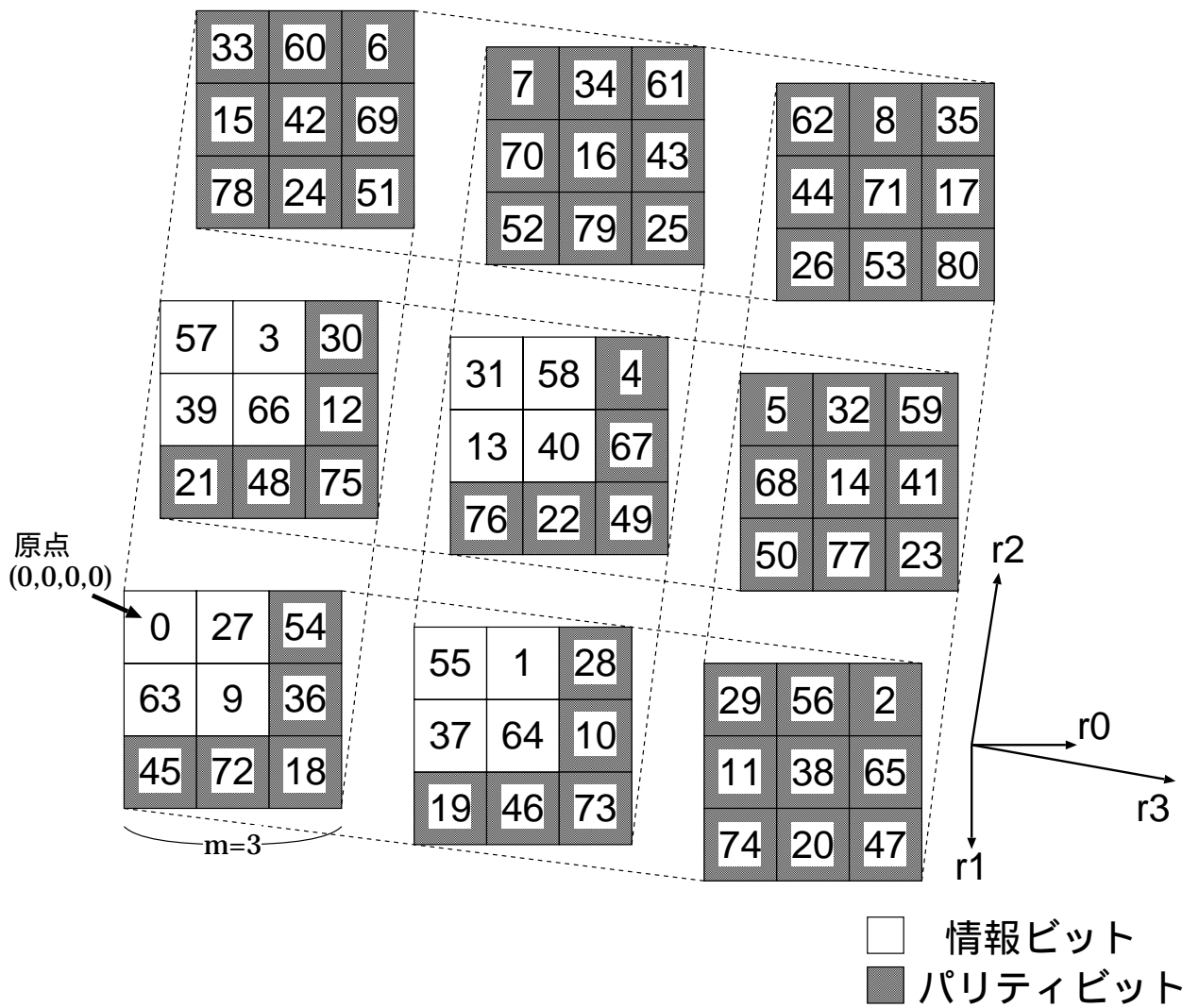


図 4.3: 4次元パリティ符号 ( $m = 3$ ) の伝送順序  $o$  と座標  $(r_0, r_1, r_2, r_3)$  との関係



$$r_1 = \lfloor o/9 \rfloor \bmod 3$$

$$r_2 = \lfloor o/3 \rfloor \bmod 3$$

$$r_3 = o \bmod 3$$

によって, 表 4.3 のように伝送順序  $o$  から座標  $(r_0, r_1, r_2, r_3)$  を求めることができる.

表 4.3: 4 次元パリティ符号 ( $m = 3$ ) の伝送順序  $o$  と座標  $(r_0, r_1, r_2, r_3)$  との関係

$o$	$r_0$	$r_1$	$r_2$	$r_3$	$o$	$r_0$	$r_1$	$r_2$	$r_3$	$o$	$r_0$	$r_1$	$r_2$	$r_3$
0	0	0	0	0	27	1	0	0	0	54	2	0	0	0
1	1	0	0	1	28	2	0	0	1	55	0	0	0	1
2	2	0	0	2	29	0	0	0	2	56	1	0	0	2
3	1	0	1	0	30	2	0	1	0	57	0	0	1	0
4	2	0	1	1	31	0	0	1	1	58	1	0	1	1
5	0	0	1	2	32	1	0	1	2	59	2	0	1	2
6	2	0	2	0	33	0	0	2	0	60	1	0	2	0
7	0	0	2	1	34	1	0	2	1	61	2	0	2	1
8	1	0	2	2	35	2	0	2	2	62	0	0	2	2
9	1	1	0	0	36	2	1	0	0	63	0	1	0	0
10	2	1	0	1	37	0	1	0	1	64	1	1	0	1
11	0	1	0	2	38	1	1	0	2	65	2	1	0	2
12	2	1	1	0	39	0	1	1	0	66	1	1	1	0
13	0	1	1	1	40	1	1	1	1	67	2	1	1	1
14	1	1	1	2	41	2	1	1	2	68	0	1	1	2
15	0	1	2	0	42	1	1	2	0	69	2	1	2	0
16	1	1	2	1	43	2	1	2	1	70	0	1	2	1
17	2	1	2	2	44	0	1	2	2	71	1	1	2	2
18	2	2	0	0	45	0	2	0	0	72	1	2	0	0
19	0	2	0	1	46	1	2	0	1	73	2	2	0	1
20	1	2	0	2	47	2	2	0	2	74	0	2	0	2
21	0	2	1	0	48	1	2	1	0	75	2	2	1	0
22	1	2	1	1	49	2	2	1	1	76	0	2	1	1
23	2	2	1	2	50	0	2	1	2	77	1	2	1	2
24	1	2	2	0	51	2	2	2	0	78	0	2	2	0
25	2	2	2	1	52	0	2	2	1	79	1	2	2	1
26	0	2	2	2	53	1	2	2	2	80	2	2	2	2

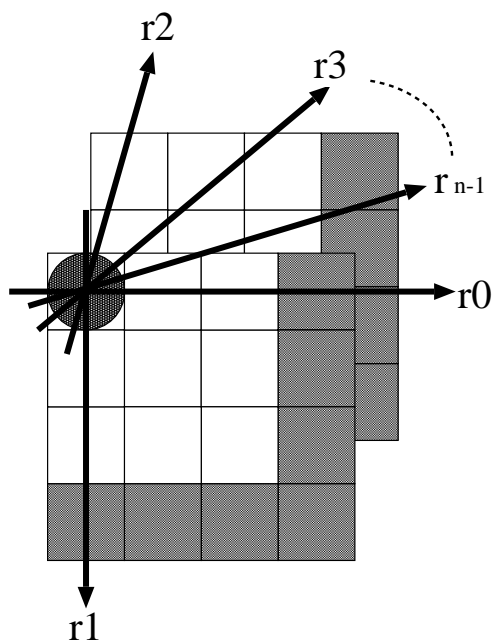


図 4.4:  $n$  次元パリティ符号における, 各ビットを貫く  $n$  本の検査線

## バースト誤り

バースト誤りとは伝送ビット中に集中して発生する誤りのことである。対して飛び飛びに発生している誤りをランダム誤りという。

符号内でバースト誤りを分散させるために本節で述べているような伝送順序を用いている。

## バースト長

誤りが集中している区間の長さ。本研究ではその区間のビットの数を単位とする。

## バースト中誤り率

バースト区間の各ビットの誤り率。バースト区間外のビットの誤り率は一律に 0。

## 4.2 復号法

本研究では伝送順序による訂正能力の違いを比較するために適応閾値判定法を用いている。本節では適応閾値判定法について簡単に説明する。

### 適応閾値判定法

$n$  次元パリティ符号の各ビットは、図 4.4 に示されているように、 $r_0, r_1, r_2, \dots, r_{n-1}$  方向の計  $n$  本のパリティ検査線で偶数パリティチェックされている、よって、もしあるビットが誤っていれば、そのビットを貫く  $n$  本のパリティ検査線の多くで誤り検出されるはずである。

適応閾値判定法は、誤りを検出したパリティ検査線が、符号内の各ビットを何本貫いているかを数値化し、その数値が最大のビットを反転して訂正するという作業を規定回数行うものである。なお、各パリティ検査線は、単一パリティ検査符号に相当するため、検査線上に 1 個 (または奇数個) の誤りが発生した時のみ、誤りを検出できる。

## 第 5 章

### 実験

本章では今までに行ってきた様々な実験について説明し，その結果を示す．

#### 5.1 ハミング距離の分布

符号語の分布がハミング距離の分布に関係しているのではないかと考え，高次元パリティ符号のハミング距離の分布を調べた．調べた符号は  $3D_{m3}$ ,  $4D_{m3}$  符号の二つである．

#### 結果

すべての符号語について距離分布を調べたが符号語による違いは見られず，表 5.1，5.2 のような一様の結果が得られた．サイズや次元の大きなものについて計算すれば何か偏りが現れるかも知れないが存在する符号語の総数が多くなり，計算量が増大するため触れていない．

表 5.1:  $3D_{m3}$  符号の距離分布

ハミング距離	0	8	12	14	16	18
符号語の数 (個)	1	27	54	108	54	12

表 5.2:  $4D_{m3}$  符号の距離分布

ハミング距離	0	16	24	28	30	32	34	36
符号語の数 (個)	1	81	324	1296	648	648	3888	6732
	38	40	42	44	46	48	50	54
	7776	9234	14472	12636	5184	1944	648	24

## 5.2 誤りを入れた符号語からのハミング距離

3D<sub>m</sub>3 符号のハミング距離から見た訂正能力を調べるため，実際に符号語に誤りをすべての組合せ入れ(ランダム誤り)，そこから元の符号語以外で一番近い符号語へのハミング距離を調べる．一番近い符号語が元の符号語になるようならば，その誤りの入り方は訂正可能であると言える．ここでは簡単にするため基本となる符号語をすべてのビットが0の符号語にする．

また，誤りの入れ方をバースト誤り(バースト中誤り率 1)に限定したものについても同様に調べる．伝送順序は図 3.2 の従来のもを用いる．

### 結果

表 5.3: ランダム誤りの訂正能力

誤りの個数	3	4	5	6	7	8	9
訂正可能(通り)	2925	15741	50355	73179	15444	0	0
訂正不可能(通り)	0	1809	30375	222831	872586	2220075	4686825

表 5.4: バースト誤りの訂正能力

バースト長(誤りの個数)	3	4	5	6	7	8	9
訂正可能(通り)	25	24	23	14	7	0	0
訂正不可能(通り)	0	0	0	8	14	20	19

表 5.3 に見られるようにランダム誤りでは3個まで誤り訂正可能になっている．この数は式 2.9 に3次元パリティ符号の最小距離である  $8(=2^3)$  を代入した時の  $t$  の値に等しい．しかし，誤りの入れ方をバースト誤りに限定した場合は表 5.4 が示すように誤り5個まで誤り訂正可能になっている．これは伝送順序を用いたバースト誤りの誤りの入り方が，訂正能力を越えて誤りを訂正できる時の誤りの入り方になっているからだと考えられる．

## 5.3 新しい伝送順序

表 5.3 より 3D<sub>m</sub>3 符号では最大で7つまで誤り訂正可能な場合が存在する．よって，その時の誤りの入り方を組み合わせればバースト長が7の時に誤りがすべて訂正できる伝

送順序を作れるのではないかと考えられる．そこで， $3D_{m3}$  符号での新しい伝送順序を考える．

バースト長を表 5.3 で最大に誤り訂正可能な個数の 7 を基本とする．その時の誤りの入り方は 21 通りである．また，表 5.3 より誤りを 7 つ入れた時に誤り訂正可能な誤りの入り方は 15444 通りある．そこで，バースト誤りの 21 通りすべてが誤り訂正可能な誤りの入り方 15444 通りに含まれるような伝送順序を見つければバースト長 7 ですべて誤り訂正可能な伝送順序が得られると考えられる．

しかし，この考え方ですべてを求めようとするると計算量がとても大きなものになってしまう．そこで計算量を減らす方法について考えた．

### 5.3.1 計算量の低減

1. 伝送順序の最初のビットを図 4.2 の座標 (0,0,0) に固定する．伝送順序で一番最初のバースト誤りの入り方が 15444 通りから 4004 通りに，他の 20 通りは 11440 通りに減少する．
2. 最初のバースト誤りの入れ方の 4004 通りの中で以下の (a) から (d) の変換をして同じになるものは削除．削除したものを最初のバースト誤りの入れ方にしてそれぞれ伝送順序を求めた時，同じ結果が得られることは確認してある．4004 通りから 93 通りに減少する．今後，(a) から (d) の変換で同じになる符号語を同じ構造の符号語と呼ぶことにする．

ここで，*code* とは 3 次元パリティ符号を 3 次元配列で表したものである．

- (a) 1 つの方向に固定し，残りの 2 つの方向を反転 (3 通り) ．

例  $code[r0][r1][r2] \Rightarrow code[r0][r2][r1]$

- (b) 各方向を回転 (3 通り) ．

例  $code[r0][r1][r2] \Rightarrow code[r1][r2][r0]$

- (c) 各方向の 2 ビット目と 3 ビット目を入れ換える．3 つの方向の 1 つから 3 つを組合せる (8 通り) ．図 5.1 参照 ．

- (d) (a) ，(b) の中の 1 通りと (c) の中の 1 通りのの組合せ ．(全 48 通り)

### 5.3.2 新しい訂正能力の評価法

まず，全体の約 1% について求めたところ 6 万通り以上の伝送順序が得られた．この得られた伝送順序について訂正能力を調べてみたところ違いは見られなかった．そこで

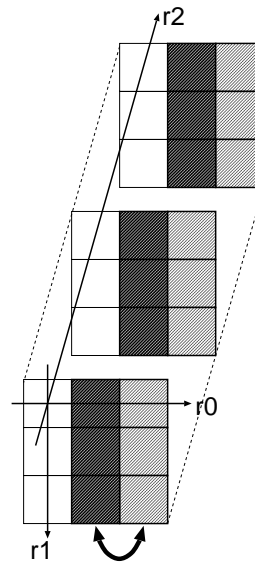


図 5.1: [例] $r_0$  方向の 2 ビット目と 3 ビット目を入れ換える

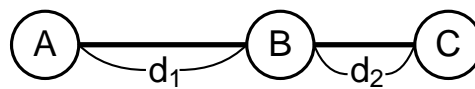


図 5.2: 訂正能力の評価法

訂正能力の新しい評価法について考える。

図 5.2 の  $A$  を元の符号語， $B$  を誤りを入れた符号語， $C$  を  $A$  以外で  $B$  に最も近い符号語とする． $d_1, d_2$  はそれぞれ各符号語間のハミング距離とする．

これまででは

訂正可能  $d_1 < d_2$

訂正不可能  $d_1 \geq d_2$

としてきたが，ここに誤り検出を加える．誤り検出の条件は，誤りを入れた符号語からのハミング距離が元の符号語とそれ以外で最も近い符号語で等しいことである．式で表すと，

訂正可能  $d_1 < d_2$

誤り検出  $d_1 = d_2$

訂正不可能  $d_1 > d_2$





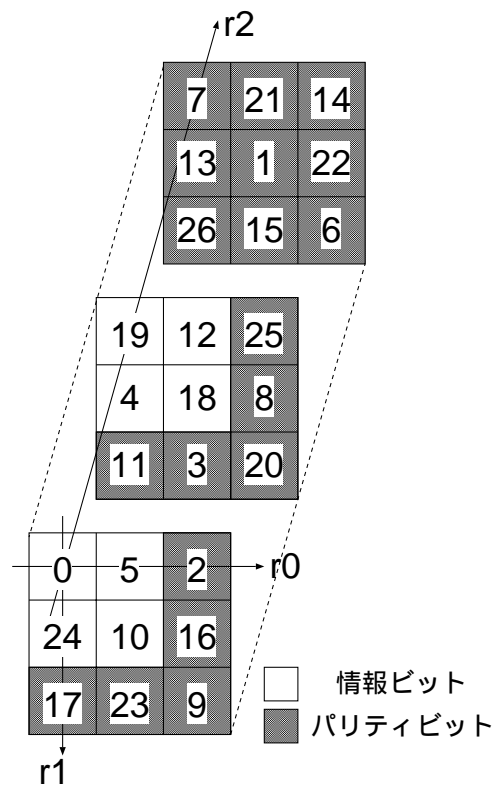


図 5.3: これまでに見つけた中で最も訂正能力の高い伝送順序 (一例)

得られた伝送順序は図 5.3 のものを，比較する従来の伝送順序は図 3.2 のものを用いる．バースト中の誤り率 0.2, 0.4, 0.6, 0.8, 1.0 の 5 つについて訂正後のブロック誤り率を求め比較する．サンプル数は 10 万とする．

## 結果

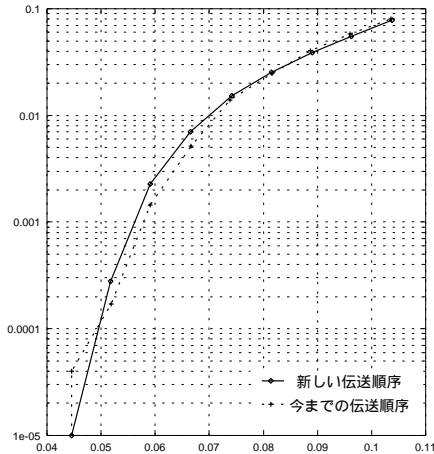


図 5.4: バースト中の誤り率 0.2

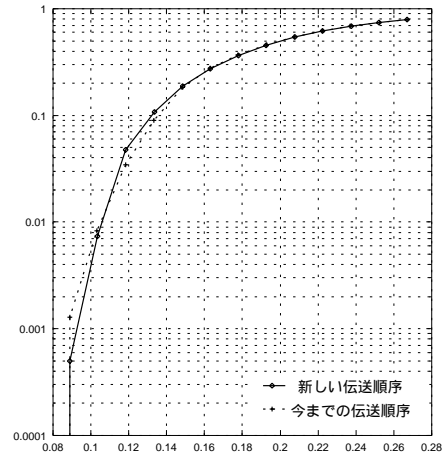


図 5.5: バースト中の誤り率 0.4

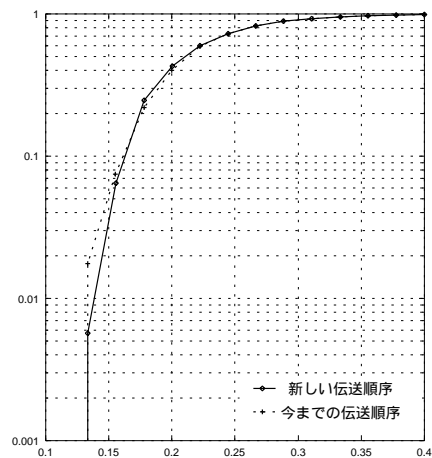


図 5.6: バースト中の誤り率 0.6

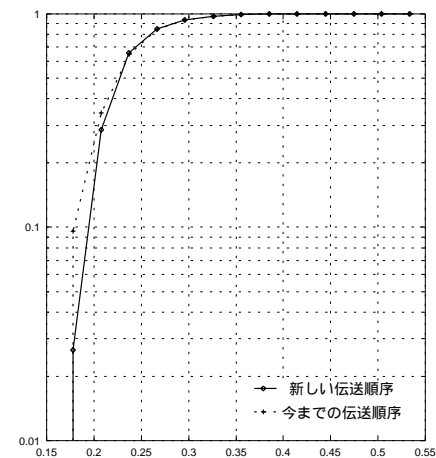


図 5.7: バースト中の誤り率 0.8

図 5.4 ~ 5.8 において横軸は訂正前のビット誤り率，縦軸は訂正後のブロック誤り率である．

バースト中の誤り率が高い場合は新しく見つけた伝送順序の方が訂正後のブロック誤り率が低くなっており若干よい結果になっている．しかし，誤り率を低くすると従来の伝送順序とあまり変わらないもしくは従来のものの方が訂正後のブロック誤り率が低く，よい結果になっている．本研究で求めた伝送順序ではこのような結果が得られたが，他

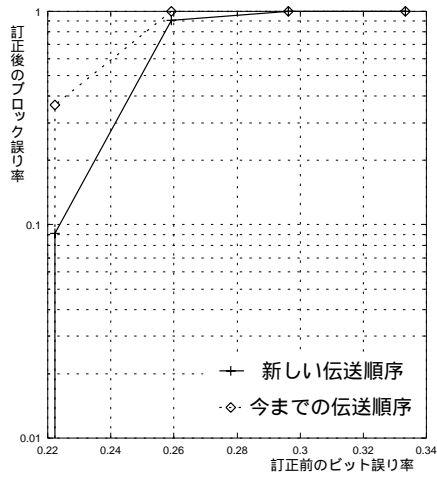


図 5.8: パースト中の誤り率 1.0

に伝送順序を求めたり, 別の復号法で比較する等していけば従来の伝送順序との違いが顕著に現れた結果を得られるかも知れない.

## 第 6 章

### まとめ

本論文では，高次元パリティ符号の構造解析のためまずハミング距離の分布との関係を調べた．しかし，どの符号語を取っても同じ距離にある符号語の個数は同じものだった．個数を数えただけでは関係を測れないものと思われる．

次に  $3D_{m3}$  符号に誤りを入れた符号語からのハミング距離の分布を求め，訂正能力について調べた．その結果から，より長いバースト誤りでも誤り訂正可能な伝送順序を見つけられるのではないかと考え，探索することにした．得られた結果について同じ構造になるもの削り，絞り込んだところいくつかの伝送順序が得られた．その訂正能力は従来のもものよりも良く，より長いバースト誤りでも誤り訂正が可能になっている．

得られた伝送順序について適応閾値判定法を用いて実際に復号し，従来との比較をしたところ，バースト中の誤り率が高い場合には訂正後のブロック誤り率が低く，良い結果になった．しかし，探索がまだ完了していないためさらに良い伝送順序があると考えられる．

今後は，探索の継続及び得られた伝送順序を一般式化し他のサイズや次元の高次元パリティ符号に対応させることがあげられる．また，他の復号法で比較する必要もある．

## 謝辞

本研究の機会を与えて頂いた中野 良平教授，終始適切な御指導を頂いた内匠逸助教授，折りにふれ多くの御助言を頂いた，臼田 毅助手に深く感謝致します。

また，共に研究を進めるにあたり，しばしば議論を交わし本研究にも少なからずご協力頂いた，中野研究室の皆様方，特に格別の御指導を頂いた志村 明夫氏，福田 登志夫氏，熱心な議論に時間を頂いた河村一宏氏に感謝をし，また厚く御礼申し上げます。

## 参考文献

- [1] 藤原良・神保雅一, “符号と暗号の数理”, 共立出版
- [2] (社)映像情報メディア学会 編, “誤り訂正符号とその応用”, オーム社, 1996.
- [3] 今井秀樹, “符号理論”, コロナ社
- [4] 汐崎 陽, “情報・符号理論の基礎”, オーム社
- [5] 虎沢 浩章, “高次元トラスパリティ符号の復号法と性能評価に関する研究”, 修士論文, 2001.