

第十二章

通信路符号化の基礎

冗長の加え方による通信路符号化の分類

- 符号
- 符号 = ([情報系列] [冗長系列])
- 生成された符号語中に、情報系列が元の形のまま存在
 - (1, 0, 0, 1) → (1, 0, 0, 1, 1, 0, 0)
- 符号
- 符号 = ([情報系列] * [符号生成系列]) 畳み込み
- 生成された符号語は全く別な系列に変換される
 - (1, 1, 0, 1) → (1, 1, 1, 0, 0, 1, 1, 1)

ブロック符号



- 情報ビット k ビットに検査ビット (冗長ビット) m ビットを付加して全体で $n (= k + m)$ ビットとなるブロック符号を一般に と呼ぶ



冗長ビットは何ビット必要か？

- n ビットの符号系列の中に 1 ビット以下の誤りがあるとする
- このとき m ビットの検査符号の情報 (最大 2^m 個の組み合わせ) から誤りの発生したビットを特定する必要がある
- 誤りのない場合
- 第1ビット目に誤り
- 第2ビット目に誤り (n+1) 通りの区別をする必要有
- ⋮
- 第nビット目に誤り
- (検査符号で表される情報) \geq (誤りが発生する場合の数)

冗長ビットは何ビット必要か? (cont'd)

5

(検査符号で表される情報) \geq (誤りが発生する場合の数)

$$2^m \geq n + 1 \quad \rightarrow \quad 2^m - m \geq k + 1$$

(n, k) 符号のビット数

符号全体のビット数 (n)	3	7	15	31	63
情報ビット数 (k)	1	4	11	26	57
検査ビット数 (m)	2	3	4	5	6



ハミング符号 (Hamming Code)

6

- 異なるビット位置の組み合わせに対する複数のパリティビットの導入
- (例) 情報 4 ビット (a_1, a_2, a_3, a_4) に対して 1 ビットの誤り訂正が可能な符号を考える
 - 1ビットの誤り訂正 $d_{min} = 3$
 - 3ビットのパリティビットを付加する必要有
 - 生成される符号語 ($a_1, a_2, a_3, a_4, C_1, C_2, C_3$)



ハミング符号 (cont'd)

7

検査ビット (C_1, C_2, C_3) を互いに独立になるように設定

$$C_1 = a_1 + a_2 + a_3$$

$$C_2 = a_1 + a_2 + a_4 \pmod{2}$$

$$C_3 = a_1 + a_3 + a_4$$

(C_1, C_2, C_3) は情報 3 ビットの偶数パリティとなる

$$S_1 = a_{r1} + a_{r2} + a_{r3} + C_{r1} = C'_{r1} + C_{r1} = 0$$

$$S_2 = a_{r1} + a_{r2} + a_{r4} + C_{r2} = C'_{r2} + C_{r2} = 0 \pmod{2}$$

$$S_3 = a_{r1} + a_{r3} + a_{r4} + C_{r3} = C'_{r3} + C_{r3} = 0$$

シンδροーム $S_i = 1$ となる時には誤り有



(7, 4) ハミング符号の例

8

- 情報符号語 (1, 1, 0, 1) を考える

$$C_1 = a_1 + a_2 + a_3 = 0$$

$$C_2 = a_1 + a_2 + a_4 = 1$$

$$C_3 = a_1 + a_3 + a_4 = 0$$

- 伝送途中で 1 ビット目に誤りが生じた

$$S_1 = a_{r1} + a_{r2} + a_{r3} + C_{r1} = 1$$

$$S_2 = a_{r1} + a_{r2} + a_{r4} + C_{r2} = 1$$

$$S_3 = a_{r1} + a_{r3} + a_{r4} + C_{r3} = 1$$

- シンδροーム全てが 1 になるため、全てに含まれる a_{r1} に誤り



ハミング符号のパリティチェック

9

シンδροームと誤り位置

伝送誤り ビット	パリティチェック		
	S_1	S_2	S_3
a_{r1}	1	1	1
a_{r2}	1	1	0
a_{r3}	1	0	1
a_{r4}	0	1	1
c_{r1}	1	0	0
c_{r2}	0	1	0
c_{r3}	0	0	1

練習問題

10

- 1ビットの誤りが訂正できる (7, 4)ハミング符号を用いて信号を送信した結果、受信機で符号 (0, 1, 0, 1, 0, 0, 1) が受信された。
- 伝送途中で誤りは生じたか？誤りのビットを特定せよ。ただし、検査ビットの生成式は本日の講義の通りとする。
- 送信された情報符号を求めよ