

誤り訂正符号 Error Correcting Code

株式会社ティ・エフ・ディ

理論、学習面で役立つ環境



- MATLAB + Communication Toolbox
- GAP, GUAVA
群論の計算システム環、体を構成して
計算可能なプログラミング環境
- Risa/Asir
富士通研究所で開発されたオープンソース
計算機代数システム

前回のまとめ

- 誤り訂正のゴールは符号化利得
- 符号／非符号語は誤り訂正の重要な概念
- シヤノン理論は究極の情報と通信の理論
- 通信路符号化定理
通信容量 C ，情報伝送速度 R とすると $R < C$ であれば、
誤り率は無限に小さくすることができる。
(エラーフリー通信が成立する。でも、やり方がわからない。)

覚えていただきたいこと



- ① 誤り訂正の基本概念
符号語と非符号語の関係
- ② ガロア体の雰囲気をつかむことが重要
取っつきにくいですが難しくはない。実装の知識があれば役立ちます。
- ③ 符号化は容易、復号化は複雑
シンδροームの計算、誤り位置の計算、値の計算
- ④ リード・ソロモン符号は組込マイコンで実装可能
- ⑤ 畳み込み符号はソフトウェア処理に向かない → FPGA
- ⑥ リード・ソロモン符号と畳み込み符号の併用は非常に効果が高い

ガロア体 ($1+1=0$)



① デジタル符号は、実数とは違い有限の元
例えば、1byte(8bits) → 256の元がある。

② 四則演算が可能

③ 加法可換群

④ 乗法可換群

有限の元(ガロア体)で、一次方程式が解けるような演算を定義できる。(重要)

誤り訂正の基本(1)

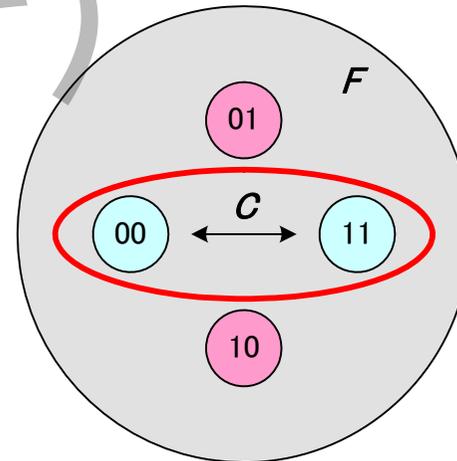
2bitで構成される符号集合をFとする。
その中で通信路符号化で使用する符号を符号語Cとする。

$$F = \{00, 01, 10, 11\}$$

$$C = \{00, 11\}$$

もし、伝送路で何らかの原因で誤りが発生し、
“01”が受信された。

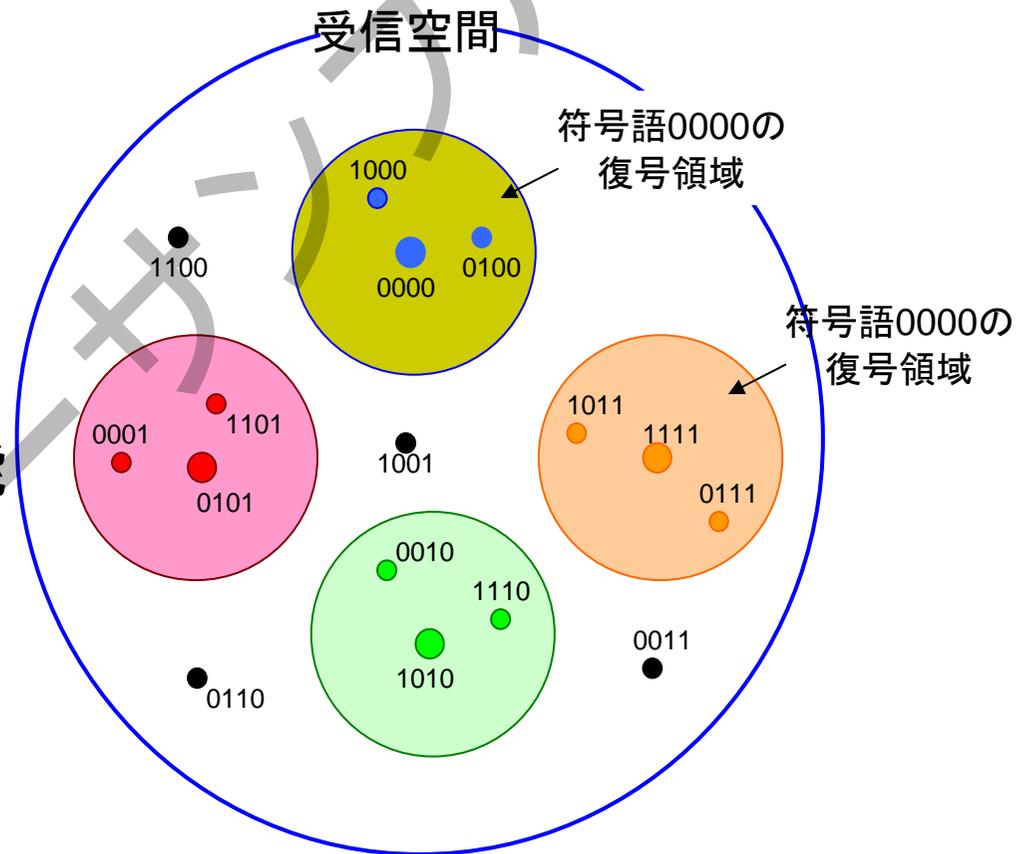
送り手と受け手の間で符号語は
“あらかじめ分かっている”ので、
誤りが発生したことがわかる。



つまり、何らかの約束事に基づいて送受信する
符号が決められていれば、誤りを発見し、訂正することが
可能。

誤り訂正の基本(2)

- 情報ビット = 2
- 冗長ビット = 2
- 符号空間 = 16
- 符号語 = 4
- 復号領域なら訂正可能
- それ以外なら誤り検出



ブロック誤り率

もし、誤り訂正をせずに通信をおこなった場合は、どうなるか？

$$P_b = 1 - (1 - P_s)^N$$

ブロック誤り率: P_b

通信路誤り率: P_s

符号長: N

無線通信ではあり得るBER=10E-3回線で、1ブロック128シンボルを送ると、

$$\begin{aligned} P_b &= 1 - (1 - 10E-3)^{128} \\ &= 0.12 \end{aligned}$$

となり、9ブロックを送信すると1ブロックの誤りが発生する。
これは無視できない。

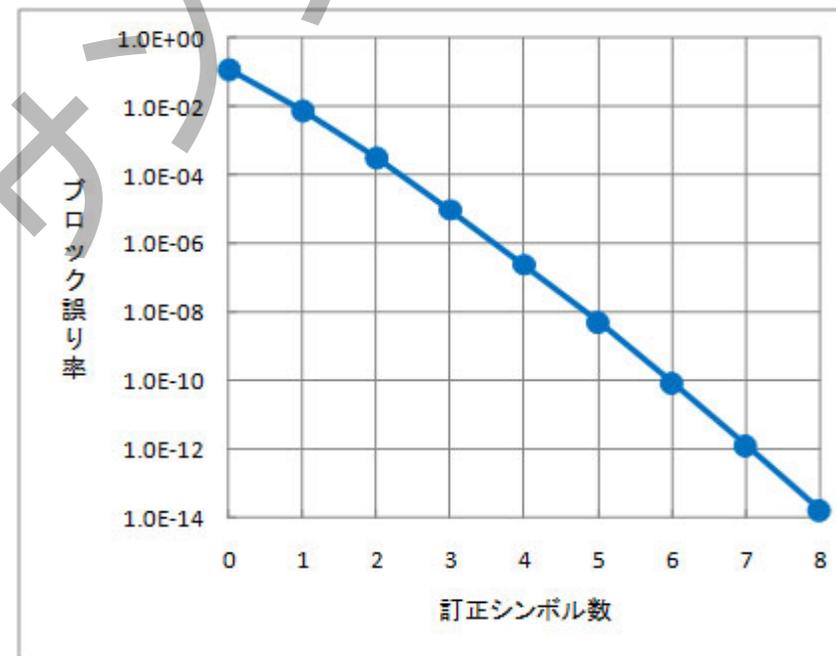
ブロック誤りの改善

誤り訂正によって、訂正シンボル数を増加させたことによるブロック誤り率の改善の様子。

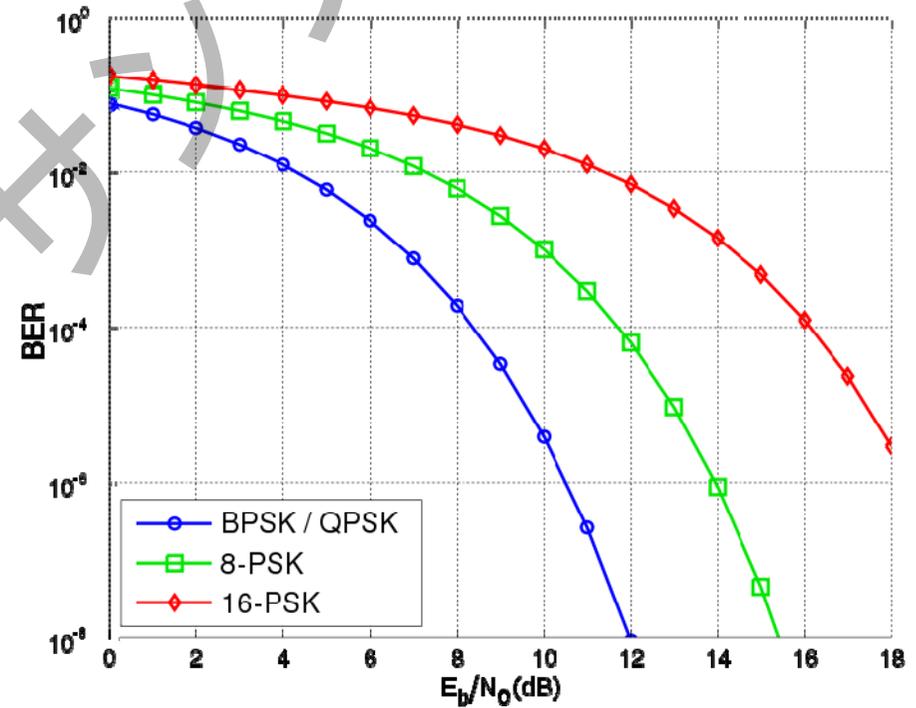
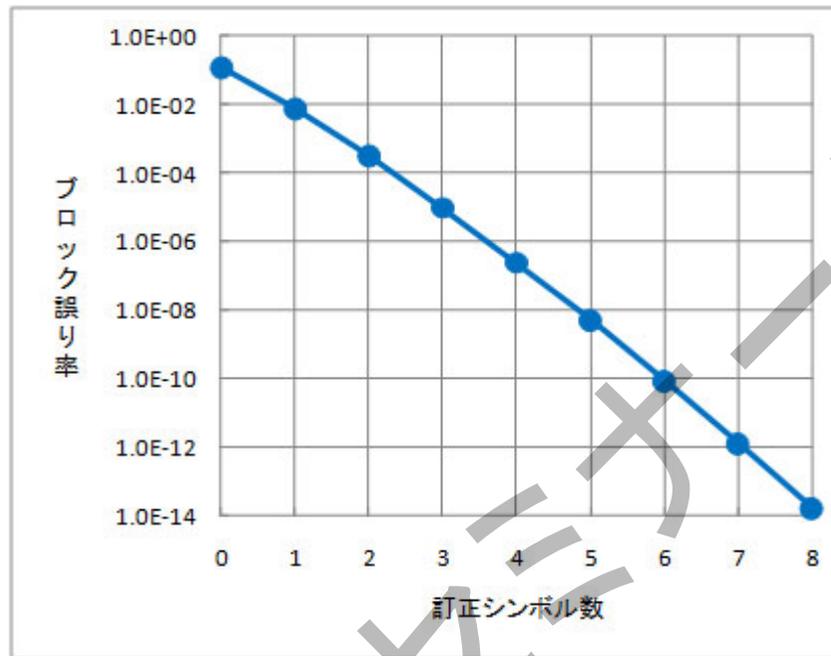
これから、訂正シンボルを増やせば無限に誤り率を向上させることができそう。

しかし、落とし穴が……。

$$P_C = 1 - \sum_{i=0}^T C_{128}^i \cdot P_S^i \cdot (1 - P_S)^{128-i}$$

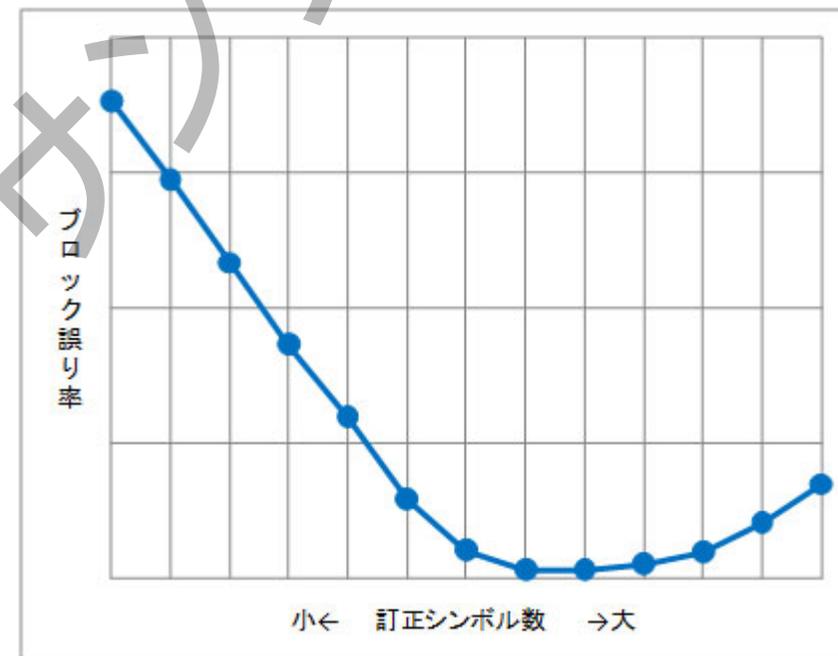


伝送速度とのトレードオフ(1)



伝送速度とのトレードオフ(2)

- 情報伝送速度を一定。
- 訂正シンボルを増やすと、ビットレートを上げるしかない。
- E_b/N_0 が小さくなる。
- 回線の誤り率が増加する。
- 訂正シンボル数も限度がある。



誤り訂正はこれに尽きる。



- ①有限の符号の集合Fから、符号語Cを数学的な関係づけで選べること。
- ②しかも、送りたい情報を符号語Cに関係づけられること。

(例) 15bitの符号集合の元の数は32,768個。

この元から、符号間距離が7と8、“0...0”、“1...1”である32個の元を選んだものが、BCH符号のひとつとなっている。