§3. 符号

送信したい文章を、2進数で送ることを考える。例えば、

math はアルファベットなので、これを2進数に変換して送る。

アルファベットは $2^4 < 26 < 2^5$ なので、5ケタを振り分けて、

$a : 00000$, $b : 00001$, $\cdots$, $z : 11001$ などとすれば、

math : 01100, 00000, 1, 0011, 00111 となる。

このように、文字の集合に対し、2進数を対応させることを符号化という。

ここで X を文字の集合とすれば、符号化は

$\sigma : X \longrightarrow \bigcup_{n=1}^{\infty} \{0, 1\}^n$: とみることができる。

㊟ 一般には値域を2進数以外で考えることもできる。

σ のみたすべき条件として、単射であることが必要なのは容易にわかる。

一方、送られた側は、送られた2進数からもとの文章に戻す(復号)必要があるため、

σ は復号できることも条件に入る。

例 $\sigma : \{a, b, c\} \longrightarrow \{0, 1, 01\}$ を

$\sigma(a) = 0$, $\sigma(b) = 1$, $\sigma(c) = 01$ とする。

送られた側にもし、01 が届いたとき、これを復号すると、

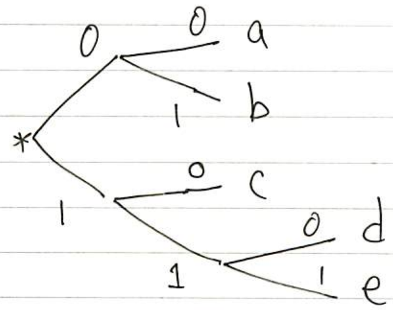
ab or c となる。これは復号できていない。

例 復号できるようにするために、

(1) 長さを一定にする (2) ある文字列を頭に含ませる などの方法もある

例. 復号できるσの例として.樹形図を用いた符号化がある.

例えば. {a, b, c, d, e} に対し.



のように対応を決める.

a : 00 , c : 10 , d : 110 である.

これで. acd を送ると. 0010110 となる.

復号化は必ず左から行い.

$\underbrace{00}_{a}\underbrace{10}_{c}\underbrace{110}_{d}$ とやると.一意に復号できる.

符号化にはさらに次が求められる.

(1) 効率

(2) エラーに対する強さ.

さきほどの例は エラーには非常によわく. 1つのエラーで文章の復号が不可能になってしまう.