

## 情報通信を行なう際の要請

(What is required in communication ?)

- 効率的に (efficiently)  
→ 情報理論 (Information Theory)
- 確実に (certainly)  
→ 符号理論 (Coding Theory)
- 安全に (safely)  
→ 暗号理論 (Cryptography)

## 情報理論 (情報源符号化・情報量の理論)

### (Information Theory, Source coding)

- 伝えるべき情報をより効率良く伝えるには  
(How to communicate more efficiently)
- 「効率の良さ」を計る  
(quantify (formulate) “efficiency”)
  - ★ 伝えるべき「情報の量」を計る  
(quantify “information”)
  - ★ 伝える為の「手間」を計る  
(quantify “cost to transmit”)

→ Shannon

「情報の量は伝えるのに必要な手間と一致」  
(“quantity of information = minimum cost”)

## 情報の符号化 (Coding)

**Analog data (continuous data)**

⇓ **sampling**

**Digital data (discrete, finite)**

↑ “**情報 (information)**” to be treated here  
(**情報源, source**)

⇓ ← “**符号化 (coding)**” to be treated here

**Digital data for transmission (伝送用データ)**

… 特定 (一般には少数) の種類の文字の列  
(**a sequence of specific alphabets**)

## 情報源符号化の定式化

### (Formulation of source coding)

**source (情報源) alphabet**  $S$  : a finite set

$S^+ := \bigsqcup_{n \geq 1} S^n$  :  $S$  の元の 1 個以上の列全体

$\varepsilon$  : 空語 (the empty word),  $S^0 := \{\varepsilon\}$

$S^* := \bigsqcup_{n \geq 0} S^n$  :  $S$  の元の 0 個以上の列全体

$$= S^+ \sqcup \{\varepsilon\}$$

$w \in S^n$  に対し、 $|w| := n$

(the length of a sequence)

## 情報源符号化の定式化

### (Formulation of source coding)

**code alphabet**  $T$  : a finite set

(typically  $T = \{0, 1\}$ )

$C : S \longrightarrow T^+$  : **符号 (code)**

$w \in \text{Im}C$  : **符号語 (code-word)**

→ 文字列を並べて  $C^* : S^* \longrightarrow T^*$  に延長  
(extended by concatenation)

## 符号への要請 (Requirement for good codes)

- **Uniquely decodable** (一意復号可能) ?
- Furthermore, **instantaneously decodable**  
(瞬時復号可能) ?
- Furthermore, **efficient** (効率的) ?

## 一意復号可能でない例

(Ex. not uniquely decodable)

$$S = \{a, b, c\}$$
$$T = \{0, 1\}$$
$$C : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 001 \end{cases}$$

「001」が ab か c か判らない  
(cannot distinguish between “ab” and “c”)

→ 一意復号可能でない!!  
(NOT uniquely decodable!!)

## 一意復号可能でない例

(Ex. not uniquely decodable)

$$S = \{a, b, c\}$$
$$T = \{0, 1\}$$
$$C : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 001 \end{cases}$$

「001」が ab か c か判らない  
(cannot distinguish between “ab” and “c”)

→ 一意復号可能でない!!  
(NOT uniquely decodable!!)



## 一意復号可能でない例

(Ex. not uniquely decodable)

$$S = \{a, b, c\}$$
$$T = \{0, 1\}$$
$$C : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 001 \end{cases}$$

「001」が ab か c か判らない  
(cannot distinguish between “ab” and “c”)

→ 一意復号可能でない!!  
(**NOT** uniquely decodable!!)

## 瞬時復号可能でない例

(Ex. not instantaneously decodable)

$$S = \{a, b, c\} \quad T = \{0, 1\} \quad c : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 11 \end{cases}$$

- Uniquely decodable (一意復号可能ではある)
- “011...”  $\implies$  “ac...” or “bc...” ?  
(“0111”  $\implies$  “bc”, “01111”  $\implies$  “acc”)  
 $\longrightarrow$  NOT instantaneously decodable  
(瞬時復号可能でない)

## 瞬時復号可能でない例

(Ex. not instantaneously decodable)

$$S = \{a, b, c\} \quad C : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 11 \end{cases}$$
$$T = \{0, 1\}$$

- **Uniquely decodable** (一意復号可能ではある)
- **“011...”  $\Rightarrow$  “ac...” or “bc...” ?**  
(“0111”  $\Rightarrow$  “bc”, “01111”  $\Rightarrow$  “acc”)  
 $\rightarrow$  **NOT instantaneously decodable**  
(瞬時復号可能でない)

## 瞬時復号可能でない例

(Ex. not instantaneously decodable)

$$S = \{a, b, c\} \quad C : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 11 \end{cases}$$
$$T = \{0, 1\}$$

- Uniquely decodable (一意復号可能ではある)
- “011...”  $\implies$  “ac...” or “bc...” ?  
(“0111”  $\implies$  “bc”, “01111”  $\implies$  “acc”)  
 $\longrightarrow$  NOT instantaneously decodable  
(瞬時復号可能でない)

## 瞬時復号可能でない例

(Ex. not instantaneously decodable)

$$S = \{a, b, c\} \quad C : \begin{cases} a \mapsto 0 \\ b \mapsto 01 \\ c \mapsto 11 \end{cases}$$
$$T = \{0, 1\}$$

- Uniquely decodable (一意復号可能ではある)
- “011...”  $\implies$  “ac...” or “bc...” ?  
(“0111”  $\implies$  “bc”, “01111”  $\implies$  “acc”)  
 $\longrightarrow$  **NOT** instantaneously decodable  
(瞬時復号可能でない)

## 瞬時復号可能な例

(Ex. instantaneously decodable)

$$\begin{array}{l} S = \{a, b, c\} \\ T = \{0, 1\} \end{array} \quad \mathcal{C} : \begin{cases} a \mapsto 0 \\ b \mapsto 10 \\ c \mapsto 11 \end{cases}$$

→ **How can one distinguish  
instantaneously decodable codes  
by looking only at  $\mathcal{C}(S) = \{0, 10, 11\} \subset T^+$  ?**

## 符号への要請 (Requirement for good codes)

- 一意符号 (uniquely decodable code):

$$\mathcal{C}^* : S^* \longrightarrow T^* : \text{injective (単射)}$$

- 瞬時符号 (instantaneously decodable code):

$$\mathcal{C}^*(\mathbf{x}) = \mathcal{C}(s)\mathbf{w} \implies \mathbf{x} = s\mathbf{y}$$

(If the received sequence starts with  $\mathcal{C}(s)$ ,  
the source sequence starts with  $s$ .)

- 効率が良い (efficient)

... the lengths  $|\mathcal{C}(s)|$  of code-words are small

## 瞬時符号の性質

### (Properties of instantaneously decodable codes)

- $\mathcal{C}$ : instant. decodable  $\implies \mathcal{C}$ : uniq. decodable
- $\mathcal{C}$  : instant. decodable  
 $\iff \mathcal{C}$  : **prefix code** (語頭符号)  
 $(\mathcal{C}(s') = \mathcal{C}(s)\mathbf{x} \implies s' = s, \mathbf{x} = \varepsilon)$

## 瞬時符号の作り方

### (How to construct instant. decodable codes)

符号木 (code tree) を考えよう



## 瞬時符号の性質

### (Properties of instantaneously decodable codes)

- $\mathcal{C}$ : instant. decodable  $\implies \mathcal{C}$ : uniq. decodable
- $\mathcal{C}$  : instant. decodable  
 $\iff \mathcal{C}$  : **prefix code** (語頭符号)  
 $(\mathcal{C}(s') = \mathcal{C}(s)\mathbf{x} \implies s' = s, \mathbf{x} = \varepsilon)$

## 瞬時符号の作り方

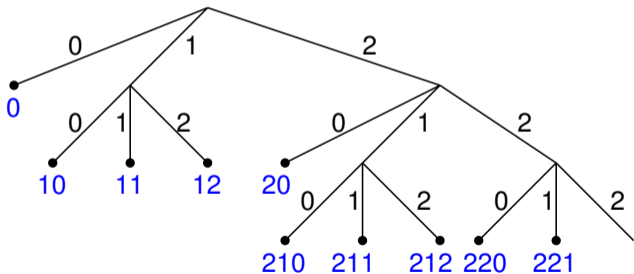
### (How to construct instant. decodable codes)

符号木 (**code tree**) を考えよう

## 符号木 (code tree)

Ex.  $T = \{0, 1, 2\}$

$\mathcal{C}(S) = \{0, 10, 11, 12, 20, 210, 211, 212, 220, 221\}$

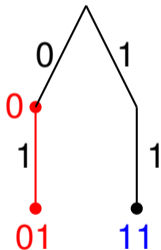


## 符号木と瞬時復号可能性

### (Code trees and instantaneous decodability)

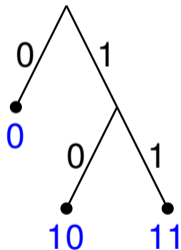
Ex.  $T = \{0, 1\}$

$$\mathcal{C}(S) = \{0, 01, 11\}$$



**not** instant. decodable

$$\mathcal{C}(S) = \{0, 10, 11\}$$



**instant.** decodable

## 瞬時符号の効率

**(Efficiency of instantaneously decodable codes)**

瞬時符号という条件を満たしつつ、

出来るだけ効率良くしたい

**(Under the condition to be instant. decodable,  
make it as efficient as possible.)**

**The list of the lengths of the code-words**

$$(|\mathcal{C}(s)|)_{s \in S} = (\mathcal{C}(s_1), \mathcal{C}(s_2), \dots, \mathcal{C}(s_k))$$

**is to be as “small” as possible.**

## Kraft の不等式 (Kraft's inequality)

$$S = \{s_1, \dots, s_k\}, \quad \#T = r \text{ (r-ary code)}$$

**For a sequence  $(\ell_1, \dots, \ell_k)$  of natural numbers,**

**$\exists$  an r-ary instant. decodable code  $\mathcal{C}$   
with  $|\mathcal{C}(s_i)| = \ell_i \ (\forall i)$**

$$\iff \sum_{i=1}^k \frac{1}{r^{\ell_i}} \leq 1$$

**instant. decodable  $\implies$  uniq. decodable**  
**(the converse is not true)**  
**(Uniq. decodability is a weaker condition.)**

**If we allow uniq. decodable codes,**  
**can we make the list of lengths more small ?**

**$\longrightarrow$  No!!**

**(For any uniq. decodable code,**  
**there exists a instant. decodable code**  
**with the same list of lengths.)**

**instant. decodable  $\implies$  uniq. decodable**  
**(the converse is not true)**  
**(Uniq. decodability is a weaker condition.)**

**If we allow uniq. decodable codes,**  
**can we make the list of lengths more small ?**

**$\longrightarrow$  No!!**

**(For any uniq. decodable code,**  
**there exists a instant. decodable code**  
**with the same list of lengths.)**

## McMillan の不等式 (McMillan's inequality)

$$S = \{s_1, \dots, s_k\}, \quad \#T = r \text{ (r-ary code)}$$

**For a sequence  $(\ell_1, \dots, \ell_k)$  of natural numbers,**

**$\exists$  an r-ary uniquely decodable code  $\mathcal{C}$**   
**with  $|\mathcal{C}(s_i)| = \ell_i$  ( $\forall i$ )**

$$\iff \sum_{i=1}^k \frac{1}{r^{\ell_i}} \leq 1$$