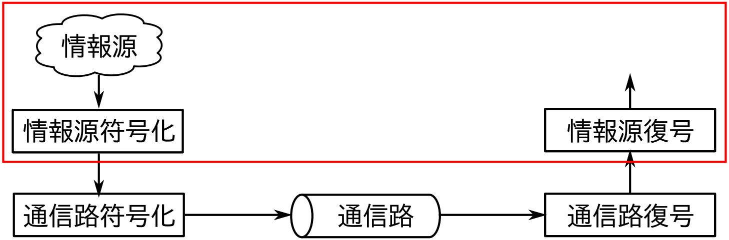
情報理論 第 4 回
– 情報源符号化の基礎 –
(教科書: 6 章, 7 章)

野崎 隆之

概要



今回と次回で取り扱う内容

情報源符号化と情報源符号化定理

1. 情報源のモデル化
2. 情報源符号化の基礎
3. 符号の分類
4. クラフトの不等式
5. 平均符号語長
6. 情報源符号化定理
7. ハフマン符号 (コンパクトな情報源符号)

今日の目的と流れ

今日の目標

- 最も簡単な情報源モデルである **定常無記憶情報源** を理解する
- 情報源符号化と復号法を理解する
- 符号の分類を理解する
- クラフトの不等式とその意味を理解する

今日の流れ

1. 情報源モデル
2. 符号化の基礎 (情報源符号化と復号法)
3. 符号の分類
4. クラフトの不等式

1. 情報源のモデル化

モチベーション

実際の現象 (情報源) をそのまま扱うのは難しい
$\Rightarrow$ 確率を使って情報源を数学的に (簡単に) 記述する

確率モデルの利点

1. **理論がシンプルになる (わかりやすい)**
2. **確率モデルを複雑にすれば現実の現象を記述可能**

この講義では最も簡単な情報源モデルで圧縮の理論を学ぶ
(最も簡単なモデルで理論を組み立てれば，複雑なモデルでの理論の構築法も理解できる)

(1-1) モデルの種類

	定常	非定常
無記憶	定常無記憶情報源	無記憶情報源
記憶あり	定常情報源	一般の情報源

定常：時刻によらず確率が一定
非定常：時刻に確率が依存

無記憶：前の結果に依存しない
記憶あり：前の結果に依存する

(1-1) モデルの種類 (例)

(例) サイコロの出目

サイコロを何回も振って出目を出力する情報源

- サイコロの出目は前の結果に依存しない $\Rightarrow$ 無記憶
- サイコロの出目の確率分布はずっと変わらない $\Rightarrow$ 定常

$\Rightarrow$ 定常無記憶情報源

(例) 英文

英文を出力する情報源

- 出現する文字は手前の文字に依存 $\Rightarrow$ 記憶あり
- 文字の分布は文字の位置によらない $\Rightarrow$ 定常

$\Rightarrow$ 定常情報源

(1-2) 用語の定義 (1:情報源アルファベット)

(定義) 情報源アルファベット $\mathcal{A}$

情報源から出力されるシンボルの集合

(例) 情報源がデジタルデータ

$\mathcal{A} = \{0, 1\}$

(例) 情報源がサイコロの出目

$\mathcal{A} = \{1, 2, 3, 4, 5, 6\}$

(例) 情報源が英文

$\mathcal{A} = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \ldots, \mathrm{z}, ␣\}$, 但し ␣ はスペースを表す記号

(1-2) 用語の定義 (2:情報源アルファベット)

(定義) 情報源 $\boldsymbol{X} := X_1 X_2 \cdots X_n$

長さ n のアルファベットの系列を出力するもの

- $\mathcal{A}^n$: **アルファベット $\mathcal{A}$ を n 個並べた集合**
- $\boldsymbol{x} = x_1 x_2 \cdots x_n \in \mathcal{A}^n$: **情報源の出力** ($x_i \in \mathcal{A}$)

注意：情報源は確率変数と確率分布によって定義される．

(例)$\mathcal{A} = \{0, 1\}$, $n = 2, 3$ の場合

$\mathcal{A}^2 = \{00, 01, 10, 11\}$
$\mathcal{A}^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}$

(1-3) 情報源モデル (1: 一般の情報源)

(定義) 一般の情報源モデル

X_i : i 番目のシンボルを表す確率変数．
一般の情報源モデルは次式で定義される

$$P_{X_1 X_2 \cdots X_n}(x_1 x_2 \cdots x_n)$$

(この確率は 1 番目のシンボルが x_1 でかつ 2 番目のシンボルが x_2 でかつ $\ldots$ n 番目のシンボルが x_n である確率である)

意味
情報源は一般に確率変数と確率分布によって定義される．

注意
アルファベット $\mathcal{A}$ の要素数を $|\mathcal{A}|$ と書くとき，一般情報源を記述するには $|\mathcal{A}|^n$ 種類の確率を取り扱う必要がある．
$\Rightarrow$ かなり複雑なモデル！

(1-3) 情報源モデル (2: 無記憶情報源)

(定義) 無記憶情報源

$X_1, X_2, \ldots, X_n$ が互いに独立のときに無記憶と呼ぶ

$$\begin{aligned} P_{X_1 X_2 \cdots X_n}(x_1 x_2 \cdots x_n) &= P_{X_1}(x_1) P_{X_2}(x_2) \cdots P_{X_n}(x_n) \\ &= \prod_{i=1}^{n} P_{X_i}(x_i) \end{aligned}$$

意味

i 番目のシンボルは他のシンボルに依存しない

参考

一般の情報源では i 番目のシンボルは 1 番目から $i-1$ 番目のシンボルに依存 (条件付き確率の性質)

$$\begin{aligned} P_{X_1 X_2 X_3}(x_1 x_2 x_3) &= P_{X_3 \mid X_1 X_2}(x_3 \mid x_1 x_2) P_{X_1 X_2}(x_1 x_2) \\ &= P_{X_3 \mid X_1 X_2}(x_3 \mid x_1 x_2) P_{X_2 \mid X_1}(x_2 \mid x_1) P_{X_1}(x_1) \end{aligned}$$

(1-3) 情報源モデル (3: 定常無記憶情報源)

(定義) 定常無記憶情報源

$X_1, X_2, \ldots, X_n$ が互いに独立でかつ同分布のときに定常無記憶と呼ぶ

$$P_{X_1 X_2 \cdots X_n}(x_1 x_2 \cdots x_n) = P_{X_1}(x_1) P_{X_2}(x_2) \cdots P_{X_n}(x_n) = \prod_{i=1}^{n} P_{X_i}(x_i)$$

$X_1, X_2, \ldots, X_n$ は同分布なので，ひとつの確率変数 X で表せば

$$P_{X_1 X_2 \cdots X_n}(x_1 x_2 \cdots x_n) = \prod_{i=1}^{n} P_X(x_i)$$

注意

定常無記憶情報源を記述するには $P_X(x)$ のみを取り扱えばよい
$\Rightarrow$ $|\mathcal{A}|$ 種類の確率だけを取り扱えばよい
$\Rightarrow$ 簡単なモデル!

(1-3) 情報源モデル (4: 例)

(例) 定常無記憶情報源

$P_X(0) = 0.6, P_X(1) = 0.4$ の場合，
定常無記憶情報源 $X^2 = X_1 X_2$ は以下の通りの分布をもつ

$$\begin{aligned}
P_{X^2}(00) &= P_X(0)P_X(0) = 0.6 * 0.6 = 0.36 \\
P_{X^2}(01) &= P_X(0)P_X(1) = 0.6 * 0.4 = 0.24 \\
P_{X^2}(10) &= P_X(1)P_X(0) = 0.4 * 0.6 = 0.24 \\
P_{X^2}(11) &= P_X(1)P_X(1) = 0.4 * 0.4 = 0.16
\end{aligned}$$

定常無記憶情報源の記述法

定常無記憶情報源 X でシンボル a_i が確率 p_i で発生するとき (すなわち，$P_X(a_i) = p_i$)，以下のように情報源 X を記述する．

$$X : \begin{pmatrix} a_1 & a_2 & \cdots & a_k \\ p_1 & p_2 & \cdots & p_k \end{pmatrix} \tag{1}$$

2. 符号化の基礎 (1:概要)

復習：情報の圧縮 (2 元符号化)

元の情報	I am a student.	(情報源からの出力)
(符号化)	$\Downarrow$	
符号語	01101...	(圧縮されたディジタルデータ)
(復号)	$\Downarrow$	
	I am a student.	

モチベーション

- どうやって英文 (情報源アルファベット) をディジタルデータに変換するのか?
  $\Rightarrow$ 情報源符号化
- ディジタルデータをどうやって元に戻すのか? $\Rightarrow$ 復号法

2. 符号化の基礎 (2:符号化)

$\mathcal{A}$: **情報源アルファベット**
$\mathcal{B}$: **符号アルファベット (符号語のシンボルの集合)**

(例) 英文のディジタルデータへの変換

$\mathcal{A} = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \ldots, \mathsf{z}, \sqcup\}$,
$\mathcal{B} = \{0, 1\}$

この講義では $\mathcal{B} = \{0, 1\}$ の場合のみを扱う (2 元符号)

(定義) 符号

情報源アルファベットの系列 $\mathcal{A}^*$ を符号アルファベットの系列 $\mathcal{B}^*$ に対応づける関数 (対応表) を符号と呼ぶ

$$\begin{aligned} f: \quad & \mathcal{A}^* \to \mathcal{B}^* \\ & x_1 x_2 \ldots \mapsto f(x_1 x_2 \ldots) \end{aligned}$$

符号化：符号語 $f(x_1 x_2 \ldots)$ を生成する動作

2. 符号化の基礎 (3:符号化)

この講義では符号が符号語の連結で表せる場合のみを扱う

$$f(x_1x_2x_3\cdots) = f(x_1)f(x_2)f(x_3)\cdots$$

(例) 符号化

符号 f

x	$f(x)$
a	1
b	01
c	001
d	000

bacbd **の符号化**

$$\begin{aligned} f(\mathsf{bacbd}) &= f(\mathsf{b})f(\mathsf{a})f(\mathsf{c})f(\mathsf{b})f(\mathsf{d}) \\ &= 01\ 1\ 001\ 01\ 000 \end{aligned}$$

(例題)

$f(\mathsf{bbadc}) =$

3. 符号化の基礎 (4:復号)

復号：符号語を情報アルファベットの系列に戻す作業

(例) 復号

符号 f

x	$f(x)$
a	1
b	01
c	001
d	000

01100101000 の復号
(頭から順に直す)

$$
\begin{array}{lllllll}
 & 0\,1\,1\,0\,0\,1\,0\,1\;0\,0\,0 \\
\Rightarrow & \mathrm{b}\;\;1\,0\,0\,1\,0\,1\;0\,0\,0 \\
\Rightarrow & \mathrm{b}\;\;\mathrm{a}\,0\,0\,1\,0\,1\;0\,0\,0 \\
\Rightarrow & \mathrm{b}\;\;\mathrm{a}\;\;\mathrm{c}\;\;0\,1\;0\,0\,0 \\
\Rightarrow & \mathrm{b}\;\;\mathrm{a}\;\;\mathrm{c}\;\;\mathrm{b}\;0\,0\,0 \\
\Rightarrow & \mathrm{b}\;\;\mathrm{a}\;\;\mathrm{c}\;\;\mathrm{b}\;\;\mathrm{d}
\end{array}
$$

(例題)

001000101 の復号

3. 符号の分類 (1:導入)

モチベーション

ちゃんと復号のできる性質の良い符号の条件は?

符号の分類の流れ

1. **正則** v.s. **特異符号**
2. **一意復号可能** v.s. **一意復号不能**
3. **瞬時符号** v.s. **非瞬時符号**
4. **可変長符号** v.s. **固定長符号 (どちらでも良い)**

この話題の流れ

(3-1) **符号の分類**

(3-2) **瞬時符号の条件 (語頭条件)**

(3-3) **符号の木**

(3-1) 符号の分類 (2:正則・特異符号)

x	$f_0(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$
a	00	0	0	00	0	00
b	00	1	1	10	10	01
c	10	10	10	01	110	10
d	11	10	11	011	111	11
	特異符号		**正則な符号**			

(定義) 特異符号 (ダメな符号・可逆圧縮でない)

異なる 2 つの情報アルファベットが同一の符号語に割り当てられる符号
(例) $f_1(\mathrm{c}) = f_1(\mathrm{d}) = 10$

(定義) 正則な符号

特異符号でない符号を**正則**と呼ぶ

(3-1) 符号の分類 (3:一意復号可能)

<table>
<tr><th>x</th><th>$f_0(x)$</th><th>$f_1(x)$</th><th>$f_2(x)$</th><th>$f_3(x)$</th><th>$f_4(x)$</th><th>$f_5(x)$</th></tr>
<tr><td>a</td><td>00</td><td>0</td><td>0</td><td>00</td><td>0</td><td>00</td></tr>
<tr><td>b</td><td>00</td><td>1</td><td>1</td><td>10</td><td>10</td><td>01</td></tr>
<tr><td>c</td><td>10</td><td>10</td><td>10</td><td>01</td><td>110</td><td>10</td></tr>
<tr><td>d</td><td>11</td><td>10</td><td>11</td><td>011</td><td>111</td><td>11</td></tr>
<tr><td></td><td colspan="2">特異符号</td><td colspan="4">正則な符号</td></tr>
<tr><td></td><td colspan="3">一意復号不能</td><td colspan="3">一意復号可能</td></tr>
</table>

(定義) 一意復号不能 (ダメな符号)

異なる 2 つの情報系列が同一の符号語に割り当てられる符号

(例) $f_2(\text{ba}) = f_2(\text{c}) = 10$

(定義) 一意復号可能

一意復号不能でないこと

(3-1) 符号の分類 (4: 瞬時符号)

<table>
<tr><th>x</th><th>$f_0(x)$</th><th>$f_1(x)$</th><th>$f_2(x)$</th><th>$f_3(x)$</th><th>$f_4(x)$</th><th>$f_5(x)$</th></tr>
<tr><td>a</td><td>00</td><td>0</td><td>0</td><td>00</td><td>0</td><td>00</td></tr>
<tr><td>b</td><td>00</td><td>1</td><td>1</td><td>10</td><td>10</td><td>01</td></tr>
<tr><td>c</td><td>10</td><td>10</td><td>10</td><td>01</td><td>110</td><td>10</td></tr>
<tr><td>d</td><td>11</td><td>10</td><td>11</td><td>011</td><td>111</td><td>11</td></tr>
<tr><td></td><td colspan="2">特異符号</td><td colspan="4">正則な符号</td></tr>
<tr><td></td><td colspan="3">一意復号不能</td><td colspan="3">一意復号可能</td></tr>
<tr><td></td><td colspan="4">非瞬時符号</td><td colspan="2">瞬時符号</td></tr>
</table>

瞬時符号

各符号語 $f(x)$ を読み終えた時点で復号できる (瞬時復号可能な) 符号

(例) 符号 f_4: 110100111 の復号

非瞬時符号 (あまり良くない符号)

瞬時符号でない符号

(例) 符号 f_3: 011001 の復号

(3-1) 符号の分類 (4: 固定長符号・可変長符号)

<table>
<tr><th>x</th><th>$f_0(x)$</th><th>$f_1(x)$</th><th>$f_2(x)$</th><th>$f_3(x)$</th><th>$f_4(x)$</th><th>$f_5(x)$</th></tr>
<tr><td>a</td><td>00</td><td>0</td><td>0</td><td>00</td><td>0</td><td>00</td></tr>
<tr><td>b</td><td>00</td><td>1</td><td>1</td><td>10</td><td>10</td><td>01</td></tr>
<tr><td>c</td><td>10</td><td>10</td><td>10</td><td>01</td><td>110</td><td>10</td></tr>
<tr><td>d</td><td>11</td><td>10</td><td>11</td><td>011</td><td>111</td><td>11</td></tr>
<tr><td></td><td colspan="2">特異符号</td><td colspan="4">正則な符号</td></tr>
<tr><td></td><td colspan="3">一意復号不能</td><td colspan="3">一意復号可能</td></tr>
<tr><td></td><td colspan="4">非瞬時符号</td><td colspan="2">瞬時符号</td></tr>
<tr><td></td><td>固定長符号</td><td colspan="4">可変長符号</td><td>固定長符号</td></tr>
</table>

固定長符号

符号語の長さが全て同じ

可変長符号

符号語の長さが等しくない

(3-2) 瞬時符号の条件 (1)

語頭条件を満たしている符号 (語頭符号) は瞬時符号である
また，瞬時符号ならば語頭符号である

(定義) 語頭

符号語 $y = f(x)$ の先頭部分を**語頭**と呼ぶ．
語頭から y を除いたものを**真の語頭**と呼ぶ．

(例) 語頭

$y = 0100$ の語頭は $\{0, 01, 010, 0100\}$.
$y = 0100$ の真の語頭は $\{0, 01, 010\}$.

(定義) 語頭条件

すべての符号語の語頭が他の符号語にならない

(3-2) 瞬時符号の条件 (2)

(例)

x	$f_3(x)$	$f_4(x)$
a	00	0
b	10	10
c	01	110
d	011	111

$f_3(x)$ の語頭

- $f(\mathsf{a}) = 00$: $\{0\}$
- $f(\mathsf{b}) = 10$: $\{1\}$
- $f(\mathsf{c}) = 01$: $\{0\}$
- $f(\mathsf{d}) = 011$: $\{0, 01\}$
  $(01 = f(\mathsf{c}))$

語頭条件を満たさない

$f_4(x)$ の語頭

- $f(\mathsf{a}) = 0$: $\{\}$
- $f(\mathsf{b}) = 10$: $\{1\}$
- $f(\mathsf{c}) = 110$: $\{1, 11\}$
- $f(\mathsf{d}) = 111$: $\{1, 11\}$

語頭条件を満たす

(3-3) 符号の木 (1:導入)

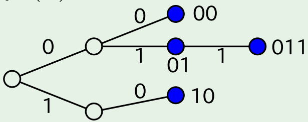
モチベーション

語頭条件をグラフィカルに表現したい (理解しやすくするため)

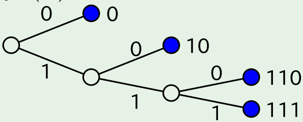
(例) 符号の木の作成 (板書)

x	$f_3(x)$	$f_4(x)$	$f_5(x)$
a	00	0	00
b	10	10	01
c	01	110	10
d	011	111	11

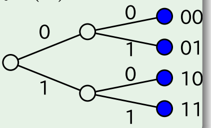
$f_3(x)$ **非瞬時符号**



$f_4(x)$ **瞬時符号**



$f_5(x)$ **瞬時符号**



語頭符号 $\Longleftrightarrow$ 符号語が全て葉 (端点) に割り当てられている

(3-3) 符号の木 (2:例題)

(例題)

x	$f_6(x)$	$f_7(x)$
a	1	0
b	01	01
c	001	10
	瞬時	非瞬時

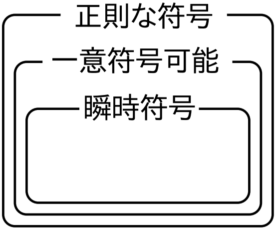
1. 符号の木を作成せよ
2. 語頭符号かどうか判定せよ

3. 符号の分類 (ここまでのまとめ)

- **可変長符号の場合**
  (語頭符号) $=$ (瞬時符号) $\subset$ (一意復号可能) $\subset$ (正則な符号)
- **固定長符号の場合**
  瞬時符号または特異符号になる（特異符号でなければ瞬時符号）

語頭符号であれば符号語が全て葉に割り当てられる

可変長符号の関係



4. クラフトの不等式 (1:導入)

モチベーション

語頭符号になるための符号長が満たすべき条件は?
(短ければ短いほどいいけど，どこまで短く出来る?)

(定義) 符号長 →符号語長の間違え

符号語 $f(a_i)$ の長さ ℓ_i

注意：符号の木において，根から符号語までの高さが符号長

(例)

x	$f_4(x)$	ℓ_i
$a_1 = \mathsf{a}$	$f(a_1) = 0$	$\ell_1 = 1$
$a_2 = \mathsf{b}$	$f(a_2) = 10$	$\ell_2 = 2$
$a_3 = \mathsf{c}$	$f(a_3) = 110$	$\ell_3 = 3$
$a_4 = \mathsf{d}$	$f(a_4) = 111$	$\ell_4 = 3$

$f_4(\mathsf{b}) = f_4(a_2) = 01,\ \ell_2 = 2$

4. クラフトの不等式 (2:定理)

(定理) クラフトの不等式

1. 2 元語頭符号が k 個の符号語をもち，それらの符号長が $\ell_1, \ell_2, \ldots, \ell_k$ であれば，次の不等式を満足する

$$\sum_{i=1}^{k} 2^{-\ell_i} \leq 1. \qquad (2)$$

2. 逆に式 (2) を満たす符号長の組 $\ell_1, \ell_2, \ldots, \ell_k$ が与えられた時に，これらの符号長を有する 2 元語頭符号を構成できる．

式 (2) の等号が成立する符号を**完全な符号**と呼ぶ

(例)

$f_4(x)$ は語頭符号であり，符号長は $1, 2, 3, 3$ であった．

$$[\text{式 (2) の左辺}] = 2^{-1} + 2^{-2} + 2^{-3} + 2^{-3} = 1 \leq 1$$

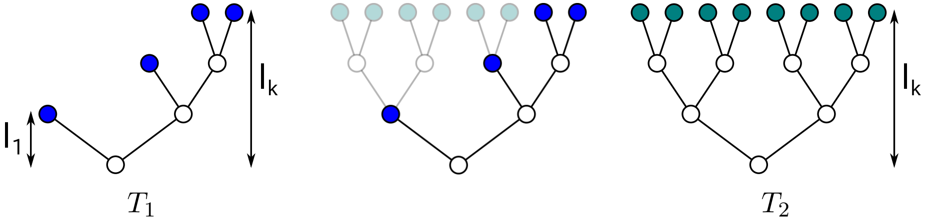
従って，クラフトの不等式を満足する．

4. クラフトの不等式 (3: 1 の証明 (1))

符号語の順番を入れ替えることで，以下を満たすようにする．

$$\ell_1 \leq \ell_2 \leq \cdots \leq \ell_k$$

T_1: **語頭符号に対応する符号語の木**
T_2: **全ての葉の高さが ℓ_k の木 (高さ ℓ_k の完全二分木)**
(証明の概略)
T_2 **の枝を削って** T_1 **を作成する**
[消えた葉の個数] $\leq$ **[T_2 の葉の個数]**



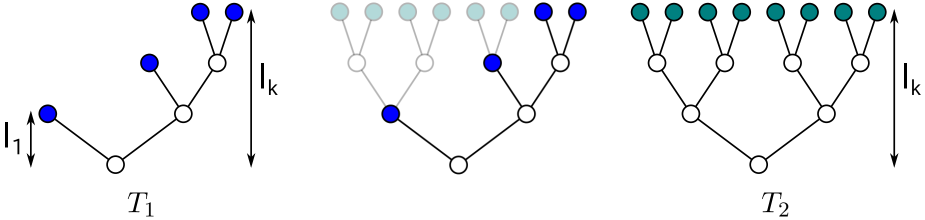
4. クラフトの不等式 (4: 1 の証明 (2))

$$[T_2\text{ の葉の個数}] = 2^{\ell_k} \tag{3}$$

$$[\text{高さ } \ell_i \text{ まで削った時に消える葉の個数}] = 2^{\ell_k-\ell_i} \tag{4}$$

$$[\text{消える葉の総数}] = \textstyle\sum_{i=1}^{k} 2^{\ell_k-\ell_i} \tag{5}$$

$$
\begin{aligned}
&[\text{消える葉の総数}] \le [T_2\text{ の葉の個数}] \\
&\iff \textstyle\sum_{i=1}^{k} 2^{\ell_k-\ell_i} \le 2^{\ell_k} \\
&\iff \textstyle\sum_{i=1}^{k} 2^{-\ell_i} \le 1 \quad (\text{両辺を } 2^{\ell_k} \text{ で割る})
\end{aligned}
$$



4. クラフトの不等式 (5: 2 の証明 (1))

$\sum_{i=1}^{k} 2^{-\ell_i} \leq 1$ かつ $\ell_1 \leq \ell_2 \leq \cdots \leq \ell_k$ を仮定する．

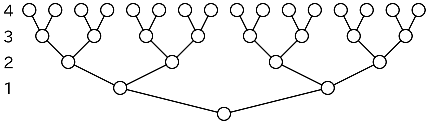
次の手順で語頭符号を構成できる．

1. 高さ ℓ_k の完全二分木を作成する
2. 高さ ℓ_1 の接点の内，一番左にあるものを符号語 $f(a_1)$ にし，子孫 (そこから上) を取り除く
3. 高さ ℓ_2 の接点の内，一番左にあるものを符号語 $f(a_2)$ にし，子孫を取り除く

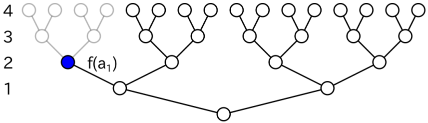
$\vdots$

4. クラフトの不等式 (6: 2 の証明 (2))

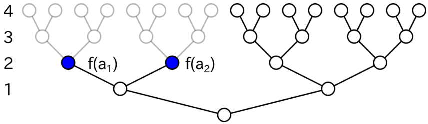
(例)：符号長 2,2,3,4 **である語頭符号**



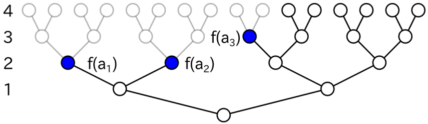
(0)



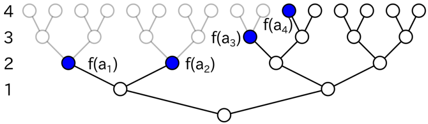
(1) $f(a_1) = 00$



(2) $f(a_2) = 01$



(3) $f(a_3) = 100$



(4) $f(a_4) = 1010$

4. クラフトの不等式 (7: 2 の証明 (3))

- $f(a_i)$ が割り当て可能であることの証明

高さ ℓ_i で割り当て可能な接点の数が正であることを示せば良い．

$$
\begin{aligned}
&[\text{高さ } \ell_i \text{ で割り当て不能な接点数}] \\
=&\textstyle\sum_{j=1}^{i-1}[f(a_j) \text{ によって削られた高さ } \ell_i \text{ の接点数}] = \textstyle\sum_{j=1}^{i-1} 2^{\ell_i-\ell_j} \qquad (6)
\end{aligned}
$$

$$
\begin{aligned}
&[\text{高さ } \ell_i \text{ で割り当て可能な接点数}] \\
=&[\text{高さ } \ell_i \text{ にある接点の総数}] - [\text{高さ } \ell_i \text{ で割り当て不能な接点数}] \\
=&2^{\ell_i} - \textstyle\sum_{j=1}^{i-1} 2^{\ell_i-\ell_j} \qquad (\text{式 (6) より}) \\
=&2^{\ell_i}(1 - \textstyle\sum_{j=1}^{i-1} 2^{-\ell_j}) \qquad (7)
\end{aligned}
$$

$$
\begin{aligned}
&\textstyle\sum_{j=1}^{i-1} 2^{-\ell_j} < \textstyle\sum_{j=1}^{k} 2^{-\ell_j} \le 1 \qquad (\text{クラフトの不等式を利用}) \\
\iff& 1 - \textstyle\sum_{j=1}^{i-1} 2^{-\ell_j} > 0 \qquad (8)
\end{aligned}
$$

式 (7) と (8) より，

$$
[\text{高さ } \ell_i \text{ で割り当て可能な接点数}] = 2^{\ell_i}(1 - \textstyle\sum_{j=1}^{i-1} 2^{-\ell_j}) > 0 \qquad (9)
$$

4. クラフトの不等式 (8: 余談) おぼえなくてもいい

"語頭符号" ではなく "一意復号可能" な場合の条件は?
$\Rightarrow$ **"語頭符号" の場合と同じ**

(定理) クラフト・マクミランの不等式

1. **一意復号可能な 2 元符号が k 個の符号語をもち，それらの符号長が $\ell_1, \ell_2, \ldots, \ell_k$ であれば，次の不等式を満足する**

$$\sum_{i=1}^{k} 2^{-\ell_i} \leq 1. \tag{10}$$

2. **逆に (2) を満たす符号長の組 $\ell_1, \ell_2, \ldots, \ell_k$ が与えられた時に，これらの符号長を有する 2 元語頭符号を構成できる．**

注意：2 について，語頭符号ならば一意復号可能であることに注意
証明：例えば，坂庭・笠井「通信理論入門」(§3.2.2) を参照せよ．

4. クラフトの不等式 (9: 余談 2) おぼえなくてもいい

2 元符号 ($\mathcal{B} = \{0, 1\}$) でない場合は?
q 元符号 ($\mathcal{B} = \{0, 1, \ldots, q-1\}$) のクラフトの不等式は次の通り

(定理)q 元符号のクラフト・マクミランの不等式

1. 一意復号可能な q 元符号が k 個の符号語をもち，それらの符号長が $\ell_1, \ell_2, \ldots, \ell_k$ であれば，次の不等式を満足する

$$\sum_{i=1}^{k} q^{-\ell_i} \leq 1. \tag{11}$$

2. 逆に式 (11) を満たす符号長の組 $\ell_1, \ell_2, \ldots, \ell_k$ が与えられた時に，これらの符号長を有する q 元語頭符号を構成できる．

今日のまとめ (1)

定常無記憶情報源

$$P_{X_1X_2\cdots X_n}(x_1x_2\cdots x_n) = \prod_{i=1}^{k} P_X(x_i)$$

表現法

$$X : \begin{pmatrix} a_1 & a_2 & \cdots & a_k \\ p_1 & p_2 & \cdots & p_k \end{pmatrix}$$

符号の分類

- **正則** v.s. **特異符号**， **一意復号可能** v.s. **一意復号不能**，
  瞬時符号 v.s. **非瞬時符号**， **可変長符号** v.s. **固定長符号**
- **瞬時符号** = **語頭符号**

今日のまとめ (2)

符号の木の構成

語頭符号 $\Longleftrightarrow$ 符号語が全て葉 (端点) に割り当てられている

クラフトの不等式

1. 語頭符号が k 個の符号語をもち，それらの符号長が $\ell_1, \ell_2, \ldots, \ell_k$ であれば，次の不等式を満足する

$$\textstyle\sum_{i=1}^{k} 2^{-\ell_i} \leq 1.$$

2. 逆に上の不等式を満たす符号長の組 $\ell_1, \ell_2, \ldots, \ell_k$ が与えられた時に，これらの符号長を有する語頭符号を構成できる．