情報理論
第 12 回 拡大情報源と動的符号化

堀田 政二
工学部 情報工学科

符号化の基本方針

平均符号長を短くする

- 発生する確率の高い記号に短い符号を割り当てる
- 発生する確率の低い記号に長い符号を割り当てる

N 種類の記号があり，各記号の発生確率と符号長をそれぞれ p_i，L_i bit $(i = 1, ..., N)$ とする

- **情報源のエントロピー**: $H = -\sum_{i=1}^{N} p_i \log_2 p_i$ bit/記号
- **平均符号長**: $L = \sum_{i=1}^{N} p_i L_i$ bit/記号

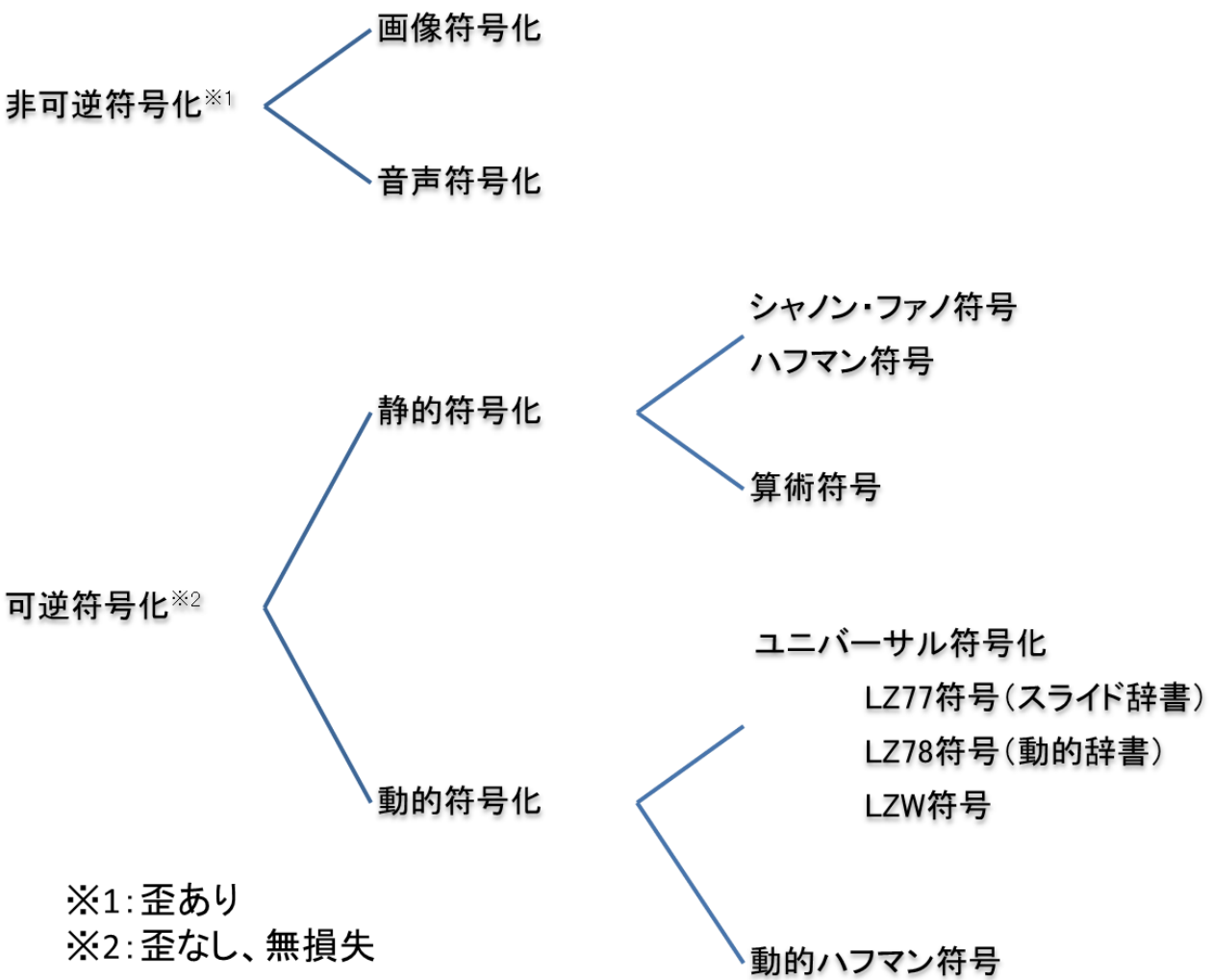
符号化の効率 e

$$e = H/L \ (0 \leq e \leq 1)$$

平均符号長 L がエントロピー H にどれだけ近いかを示す指標．
$L = H$ の時，効率が最大の 1 となる

(2)

情報源符号化（圧縮）法の分類例



※1：歪あり
※2：歪なし、無損失

(3)

拡大情報源

平均符号長 L の最短限界は情報源のエントロピー H

- 各記号を一つずつ符号化 $\cdots$ 短縮化には限界
- 複数の記号をまとめて一つの新たな記号にすればもっと短縮できる可能性がある
- 等価的に 1 記号当たりの符号長を短縮

m 次拡大情報源

情報源の記号を m 個ずつまとめた情報源．$m \geq 2$ で記号をまとめることをブロック化と呼ぶ

2 次拡大情報源の記号と確率の例

記号	確率
A	0.6
B	0.25
C	0.1
D	0.05

記号	確率	記号	確率
AA	0.6 × 0.6=0.36	CA	0.06
AB	0.6 × 0.25=0.15	CB	0.025
AC	0.6 × 0.1=0.06	CC	0.01
AD	0.6 × 0.05=0.03	CD	0.005
BA	0.25 × 0.6=0.15	DA	0.03
BB	0.25 × 0.25=0.0625	DB	0.0125
BC	0.25 × 0.1=0.025	DC	0.005
BD	0.25 × 0.05=0.0125	DD	0.0025

拡大情報源のエントロピー

情報源が N 種類の記号からなる場合，m 次拡大情報源の記号は N^m 種類となる

- 例: もともと情報源記号が A,B,C,D の 4 種類の場合，2 次拡大情報源は $4^2 = 16$ 種類の記号となる (AB と BA 等は別々の記号と考える)
- 発生確率は記憶のない情報源を考えているので，ブロック化された記号の発生確率は，元の情報源記号の生起確率の積に等しい (独立なので)

【拡大情報源のエントロピー】

- エントロピー H は 1 記号当たりの平均情報量
- 情報源を m 次に拡大した場合，そのエントロピーは m 倍の mH
- これは m 個の記号当たりのエントロピー．1 記号当たりに換算すると H で不変

ハフマンブロック符号化

ブロック化した拡大情報源にハフマン符号化を適用したもの

- 拡大情報源の記号は増えるが，確率は容易に求めることができる
- ブロック化された記号と確率を並べ，通常のハフマン符号化と同様の手順で符号化

ハフマンブロック符号化で得られる m 次拡大情報源の平均符号長

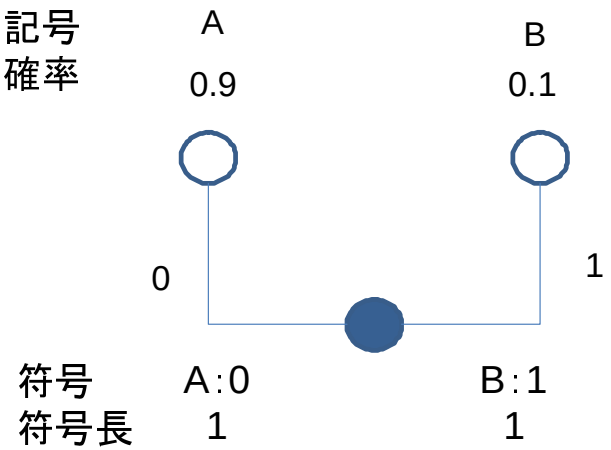
1 記号当たりの平均符号長 $L = L_m/m$ bit/記号
ここで L_m は m 個の記号当たりの符号長

符号の良さは符号化の効率 $e = H/L$ で評価

ハフマンブロック符号化における例

2 種類の記号 A, B を持つ情報源に対するブロック符号化を考える

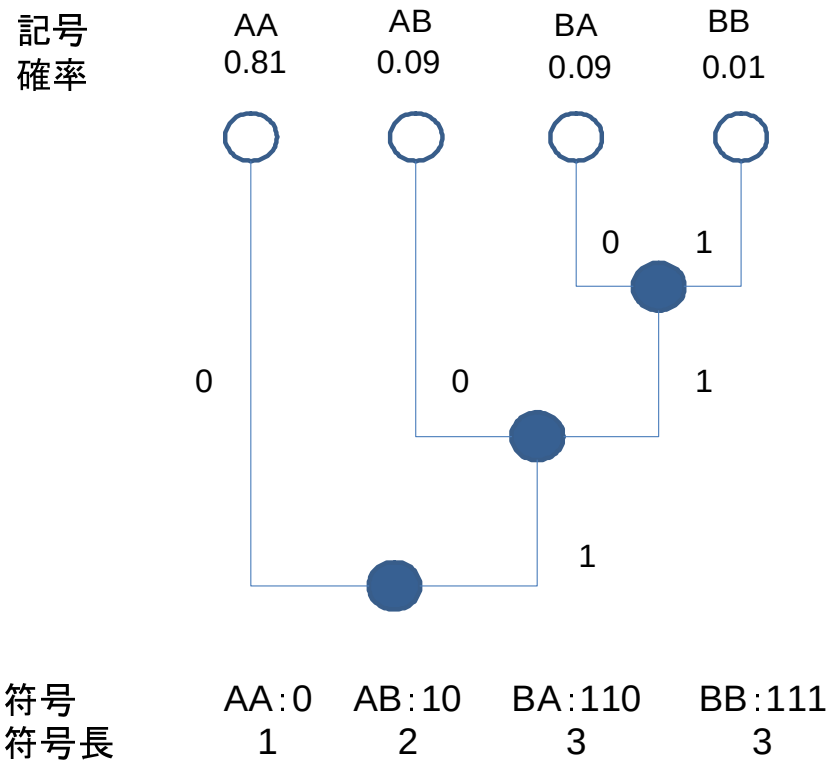
- 生起確率はそれぞれ A:0.9，B:0.1 とする
- 情報源のエントロピー
  $H = -0.9 \log_2 0.9 - 0.1 \log_2 0.1 = 0.469$ bit/記号



- 1 次の場合の平均符号長 $L_1 = 1$ bit/記号
- 符号化の効率 $e = 0.469$

(8)

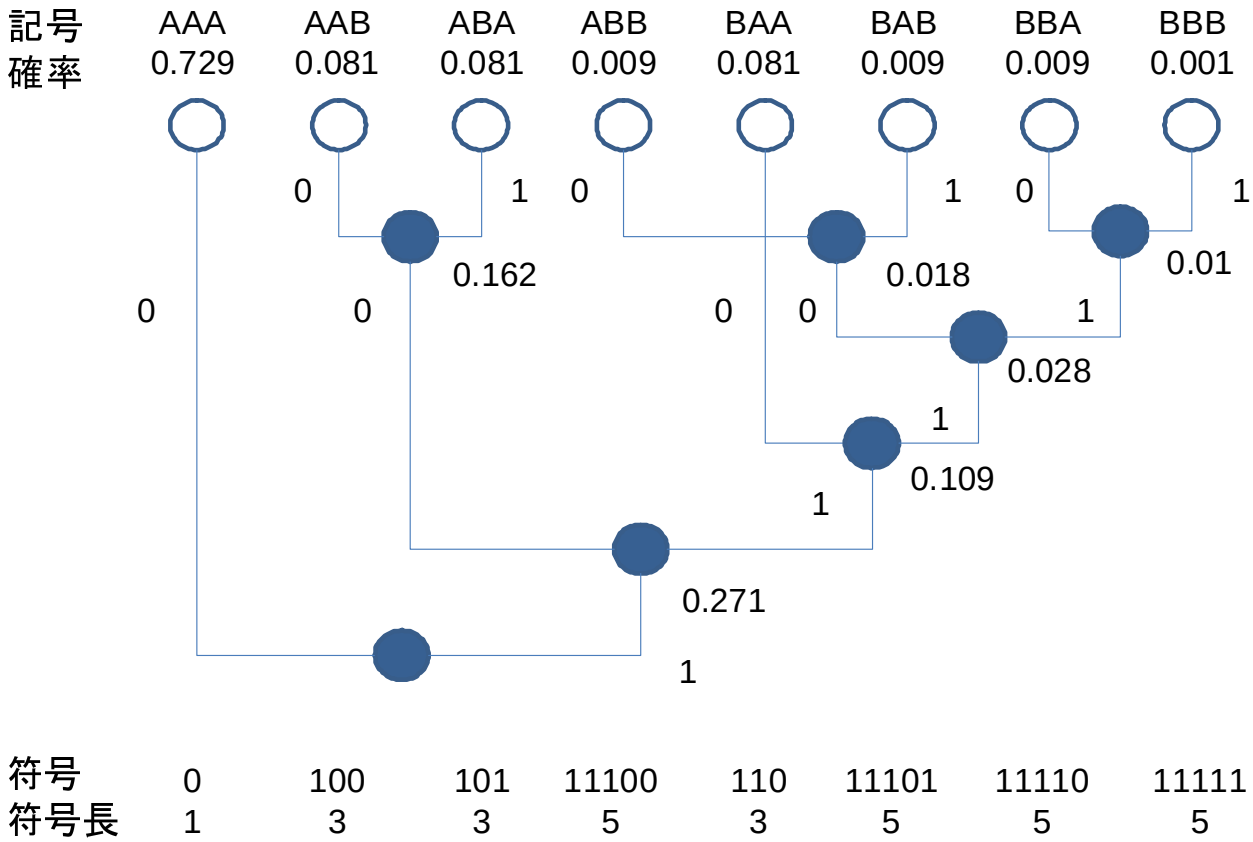
2 次拡大情報源の符号化



- **記号は** AA, AB, BA, BB **の** 4 **種類**
- **平均符号長**:
  $L_2 = (0.81 + 0.09 \times 2 + 0.09 \times 3 + 0.01 \times 3)/2 = 0.645$ bit/**記号**
- **符号化の効率**: $e = 0.727$

(9)

3 次拡大情報源の符号化



- **記号は** AAA, AAB,...,BBB **の** 8 **種類**
- **平均符号長**: $L_3 = 0.533$ bit/**記号**
- **符号化の効率**: $e = 0.880$

(10)

シャノンの第 1 基本定理

拡大情報源のように，符号化を工夫すれば平均符号長 L を情報源の 1 記号あたりのエントロピー H にいくらでも近づけることができる．次式が 0 に近い任意の正の数 ϵ について成り立つ:

$$H \leq L < H + \epsilon$$

m 次拡大情報源の m 個の記号当たりのエントロピーを $H_m = mH$，平均符号長を $L_m = mL$ とする．どのような m についても $H_m \leq L_m < H_m + 1$ が成り立つので，各項を m で割れば $H \leq L < H + 1/m$ が得られる．m を大きくしていくと $\lim_{m\to\infty}(1/m) = \epsilon$ となり定理が得られる．

(11)

シャノンの第 1 基本定理の意義

- 別名，情報源符号化定理，雑音の無い場合の符号化定理
- 通信路符号化に出てくる雑音がある場合の符号化定理と対比させた呼び名
- 情報源符号化定理は，情報源がどのような確率分布でも，平均符号長をいくらでもエントロピーに近づける符号化法が存在することを保証している
- 情報源の拡大次数を大きくすれば効率は上がるが，符号化と復号化の手順は複雑になる

テキストデータの圧縮

ハフマン符号化でテキストを符号化するには，全文を一度走査して，全記号の発生確率を求めた後，記号を符号化し，もう一度，全文を走査して符号に変換しなくてはならない．すなわち，二回走査が必要

ユニバーサル符号化

記号の発生確率を求めないで，通信文の最初から符号化する手法

代表的なもの: ジフ (Ziv) とランペル (Lempel) による LZ 符号化

- 同じ文字が繰り返し出現することに着目
- その文字列自体を送る代わりに繰り返しの情報を伝送
- 繰り返される文字列は単語として辞書に登録
- 同じ単語が現れる頻度が高いほど効率良く圧縮可能

スライド辞書法 (LZ77 符号)，動的辞書法 (LZ78 符号)

スライド辞書法 (LZ77 符号)

二種類のバッファメモリを用意

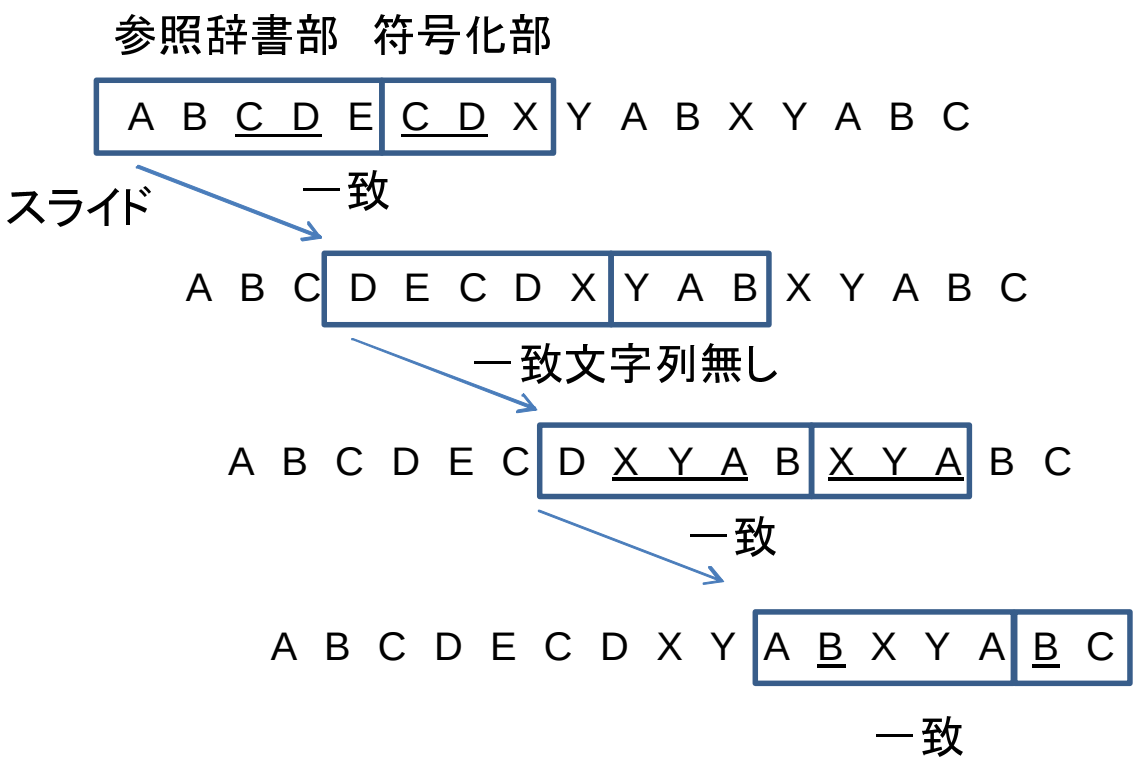
- 参照辞書部: ある文字 (記号) の列を入れるバッファ
- 符号化部: 符号化部の文字列が参照辞書部に存在する文字列に一致すれば，何個前の文字列の何個まで一致するかを符号化

【特徴】

- 参照辞書部のバッファの長さが短いと圧縮率が低下
- 最長の一致文字列を探索するには時間がかかる
- ただし，単純なアルゴリズムであり，特別な辞書は不要

スライド辞書法の例

入力記号列:A B C D E C D X Y A B X Y A B C



出力記号列(1):A B C D E

出力記号列(2):(3,2)X

出力記号列(3): Y A B

出力記号列(4): (4,3)

出力記号列(5): (4,1)C

出力記号列:A B C D E (3,2) X Y A B (4,3) (4,1) C

動的辞書法 (LZ78 符号)

スライド辞書法の欠点

- 参照辞書部に含まれない文字列は置き換えられない
- 最長の一致文字列を検索するのに時間がかかる

【動的辞書法】

- 別に辞書を用意してその辞書と照合
- 文を調べながら単語の辞書を追加・更新していく方法

【辞書】

- 送信側で完成した辞書の情報を，圧縮した元のデータに付け加えて送信
- 辞書が次第に成長していくが，辞書がツリー状に構成できるため時間はかからない

動的辞書の例

入力記号列:A B C D E C D X Y A B X Y A B C

出力と登録の手順

入力	出力	登録
A	(0,A)	#1:A
B	(0,B)	#2:B
C	(0,C)	#3:C
D	(0,D)	#4:D
E	(0,E)	#5:E
CD	(3,D)	#6:CD
X	(0,X)	#7:X
Y	(0,Y)	#8:Y
AB	(1,B)	#9:AB
XY	(7,Y)	#10:XY
ABC	(9,C)	#11:ABC

辞書の内容 (#n:登録番号)

#1	#2	#3	#4	#5	#7	#8
A | #9 AB | #11 ABC	B	C | #6 CD	D	E	X | #10 XY	Y

(n, 記号):nは辞書の登録番号 #n
0は辞書への新規登録

出力記号列:(0,A) (0,B)(0,C)(0,D) (0,E) (3,D) (0,X) (0,Y) (1,B) (7,Y) (9,C)

問題

- 【12.1】次の情報源

$$S = \begin{pmatrix} S_1 & S_2 \\ 1/3 & 2/3 \end{pmatrix}$$

のエントロピーは $\mathcal{H}(1/3) = 0.918$ である．この情報源の 2 次拡大情報源 S^2，3 次拡大情報源 S^3 を作成し，それらに対してハフマン符号化を求め，それぞれの平均符号長と符号化の効率を求めよ．