

THE IEICE TRANSACTIONS ON FUNDAMENTALS OF ELECTRONICS, COMMUNICATIONS AND COMPUTER SCIENCES (JAPANESE EDITION)

**IEICE** | **電子情報通信学会**  
**A** | **論文誌** 基礎・境界

DOI:10.14923/transfunj.2024JAI0001

早期公開日:2024/07/11

本PDFは、早期公開版である。本論文を引用する場合には、電子情報通信学会和文論文誌投稿のしおり(情報・システムソサイエティ)の「8.早期公開」を参照すること。

**基礎・境界ソサイエティ**

一般社団法人 **電子情報通信学会**

THE ENGINEERING SCIENCES SOCIETY

THE INSTITUTE OF ELECTRONICS, INFORMATION AND COMMUNICATION ENGINEERS

## ANS の符号化・復号アルゴリズムと平均符号長の評価

山本 博資<sup>†a)</sup> 岩田賢一<sup>††b)</sup>

## Encoding and Decoding Algorithms of ANS Variants and Evaluation of Their Average Code Length

Hirosuke YAMAMOTO<sup>†a)</sup> and Ken-ichi IWATA<sup>††b)</sup>

あらまし Jarek Duda によって提案された無歪みデータ圧縮符号化方式である ANS (Asymmetric Numeral Systems) は、算術符号より少ない算術演算量で算術符号とほぼ等しい圧縮性能を達成できる高性能な無歪みデータ圧縮法である。その性能の高さから、ANS は Facebook, Apple, Google, Dropbox, Microsoft, Pixar など、さまざまな実用的なシステムで既に広く使用されているが、ANS についてまだよく知らない研究者も多い。本稿では、各種類の ANS に対して、その符号化および復号アルゴリズムについて詳しく説明すると共に、ANS が達成できる平均符号長に関して情報理論的な評価を与えている。

キーワード ANS (Asymmetric Numeral Systems), 算術符号, 無歪みデータ圧縮符号, 平均符号長

## 1. まえがき

(注) ハフマン符号や算術符号などの従来の無歪みデータ圧縮符号では、データ系列  $s^T = s_1 s_2 \cdots s_T$  に対して、符号化も復号も  $s_t, t = 1, 2, \dots, T$  の同じ順で処理することを前提としている [2][3]。これに対して、Jarek Duda は算術符号の拡張として、符号化を  $s_t, t = T, \dots, 2, 1$  の逆方向で行い、復号を  $s_t, t = 1, 2, \dots, T$  の順方向で行う ANS (Asymmetric Numeral Systems) を提案した [4]–[14]。

データ系列  $s^T$  を算術符号化したときの符号語を  $c$  とすると、算術符号では  $s_t, t = 1, 2, \dots, T$  の順に符号化が行われるため、 $c$  の上位桁から値が決まって行く。しかし、 $s_1^T = s_1 s_2 \cdots s_t$  を符号化した時点では、その後の  $s_{t+1}^T = s_{t+1} \cdots s_T$  の値が分からないため、 $s_{t+1}^T$  の全ての可能性に対応できるように、 $c$  を含む実数値の区間（あるいは整数値の区間）を用いて符号化が行われる。これに対して ANS では、 $s_t, t = T, T-1, \dots, 1$

の順に符号化が行われるため、符号語  $c$  の下位ビットから決めることができ、その結果、ANS は単一の整数値を用いて符号化と復号を行うことができる。このことから、ANS は算術符号より少ない算術演算量で算術符号とほぼ同じ圧縮率を達成できる。この優れた特徴から、ANS は Facebook Zstandard (ZSTD) compressor, Apple LZFFSE compressor を始めとして、Google, Dropbox, Microsoft, Pixar などで使用されており [14][15][16]、最近は応用研究や関連研究の成果も多く発表されている [16]–[34]。

しかし、ANS に関する論文の多くは学術論文誌ではなく arXiv.org や会議論文として発表されているため、まだ ANS を知らない人も多い。また、これらの論文に書かれているアルゴリズムや性能解析が分りづらく、情報理論的な性能評価がほとんどなされていないため、ANS の名前を知っていても、その詳しい符号化・復号アルゴリズムや圧縮性能を知らない人も多い。本稿では、ANS の符号化・復号アルゴリズムを詳しくかつ分かりやすく解説すると共に、ANS が達成できる平均符号長に関して情報理論的な評価を与えている。

ANS には幾つかの種類 (variants) が存在するが、第 2 節で ABS (Asymmetric Binary Systems) を、第 3 節と第 4 節で rANS (range variant of ANS)、第 5 節で tANS

<sup>†</sup> 東京大学新領域創成科学研究科

The University of Tokyo, School of Frontier Sciences

<sup>††</sup> 福井大学大学院工学研究科

University of Fukui, School of Engineering

a) E-mail: hirosuke@ieee.org

b) E-mail: k-iwata@u-fukui.ac.jp

(注)：本論文は、第 12 回シャノン理論ワークショップ (STW2023) において発表した内容 [1] を改訂したものである。

(tabled variant of ANS<sup>(注1)</sup>) を取り扱う。各節ごとに、符号化関数、復号関数、符号化アルゴリズム、復号アルゴリズムを示したあと、それらの関数およびアルゴリズムでデータ系列が正しく復号できることを証明している。さらに、その符号化関数と符号化アルゴリズムでデータ系列を符号化したときに達成できるデータ 1 シンボル当りの平均符号長の期待値（簡単のため以下では単に平均符号長と呼ぶことにする）に対する情報理論的に厳密な上界を与えている。

本稿で取り扱う情報源は有限離散集合  $\mathcal{S}$  上の値を取る定常無記憶情報源とし、その確率分布を  $p = \{p(s) \mid s \in \mathcal{S}\}$  とする。その情報源から出力されたデータ系列を  $s^T = s_1 s_2 \cdots s_T$ ,  $s_t \in \mathcal{S}$ , とし、簡単のため、確率分布  $p$  と系列長  $T$  は既知であるとして、符号化および復号アルゴリズムを記載している。なお、本論文では次のような表記法を用いる<sup>(注2)</sup>。  $\lg a = \log_2 a$  とし、 $|\mathcal{A}|$  で集合  $\mathcal{A}$  の濃度（要素数）を示す。また、情報源のエントロピーを  $H(p) = \sum_{s \in \mathcal{S}} p(s) \lg(1/p(s))$  で表し、確率分布  $q = \{q(s) \mid s \in \mathcal{S}\}$  に対する相対エントロピーを  $D(p||q) = \sum_{s \in \mathcal{S}} p(s) \lg(p(s)/q(s))$  で示す。

## 2. ABS

この節では、情報源アルファベット  $\mathcal{S}$  が  $\mathcal{S} = \{0, 1\}$  の二元の場合を考え、 $p_1 = p(1)$ ,  $p_0 = p(0) = 1 - p_1$ ,  $0 < p_1 < 1$  とする。このとき ABS (Asymmetric Binary Systems)<sup>(注3)</sup> は整数値  $x$  を用いて符号化と復号を行う。ただし、 $x$  は任意の桁数を取れるものとする。

### 2.1 ABS の符号化・復号法 [4]–[6]

#### A. 符号化関数の定義

符号化関数  $x_{t-1} := C(s_t, x_t)$  を式 (1)(2) で定義する。なお、以下では  $x_{t-1}$  が  $s_t = 0, 1$  のどちらの値で求められたかを明示するときは、 $x_{t-1} = x_{t-1}^{(s_t)}$  という表記を用いている。

$$x_{t-1} = x_{t-1}^{(0)} := \left\lfloor \frac{x_t + 1}{p_0} \right\rfloor - 1 \quad \text{if } s_t = 0 \quad (1)$$

$$x_{t-1} = x_{t-1}^{(1)} := \left\lfloor \frac{x_t}{p_1} \right\rfloor \quad \text{if } s_t = 1 \quad (2)$$

#### B. 復号関数の定義

復号関数  $(s_t, x_t) := D(x_{t-1})$  を式 (3)–(5) で定義する。

$$s_t := \lceil (x_{t-1} + 1)p_1 \rceil - \lfloor x_{t-1}p_1 \rfloor \quad (3)$$

を求め、 $x_{t-1}^{(s_t)} := x_{t-1}$  とする。

$$x_t := x_{t-1}^{(0)} - \lfloor x_{t-1}^{(0)} p_1 \rfloor \quad \text{if } s_t = 0 \quad (4)$$

$$x_t := \lfloor x_{t-1}^{(1)} p_1 \rfloor \quad \text{if } s_t = 1 \quad (5)$$

#### C. 符号化アルゴリズム

- データ系列  $s^T = s_1 s_2 \cdots s_T$  をセットし、 $x_T \leftarrow 1$  と設定する。
- $t = T, \dots, 2, 1$  と逆順に、 $x_{t-1} \leftarrow C(s_t, x_t)$  を繰り返す。
- $x_0$  が  $s^T$  の符号語となる。

#### D. 復号アルゴリズム

- 符号語  $x_0$  をセットする。
- $t = 1, 2, \dots, T$  の順に、 $(s_t, x_t) \leftarrow D(x_{t-1})$  を繰り返す。
- $s^T = s_1 s_2 \cdots s_T$  が復号系列となる。

式 (4) と式 (5) が、それぞれ式 (1) と式 (2) の逆関数になっていることを証明する。今、 $\hat{r}_t^{(0)}$  と  $\hat{r}_t^{(1)}$  を

$$\hat{r}_t^{(0)} = x_{t-1}^{(0)} + 1 - \frac{x_t + 1}{p_0} \quad (6)$$

$$\hat{r}_t^{(1)} = \frac{x_t}{p_1} - x_{t-1}^{(1)} \quad (7)$$

で定義すると、式 (1)(2) より  $0 \leq \hat{r}_t^{(0)}, \hat{r}_t^{(1)} < 1$  である。このとき、 $0 < p_1 = 1 - p_0 \leq 1 - p_0(1 - \hat{r}_t^{(0)}) < 1$  および  $0 \leq p_1 \hat{r}_t^{(1)} < p_1 < 1$  の関係が成り立つことから、 $r_t^{(0)} = 1 - p_0(1 - \hat{r}_t^{(0)})$ ,  $r_t^{(1)} = p_1 \hat{r}_t^{(1)}$  と置くと、 $r_t^{(0)}$  と  $r_t^{(1)}$  は次の関係を満たす。

$$0 \leq r_t^{(1)} < p_1 \leq r_t^{(0)} < 1 \quad (8)$$

式 (6) より、 $x_t = x_{t-1}^{(0)} - \lfloor x_{t-1}^{(0)} p_1 + \{1 - p_0(1 - \hat{r}_t^{(0)})\} \rfloor = x_{t-1}^{(0)} - \lfloor x_{t-1}^{(0)} p_1 + r_t^{(0)} \rfloor$  が成り立つ。 $x_t$  と  $x_{t-1}^{(0)}$  が整数であるため、 $\lfloor x_{t-1}^{(0)} p_1 + r_t^{(0)} \rfloor$  は整数となる。ここで、 $r_t^{(0)}$  は式 (8) を満たすことより、 $x_{t-1}^{(0)} p_1 + r_t^{(0)} = \lfloor x_{t-1}^{(0)} p_1 \rfloor$  でなければならない。ゆえに式 (4) が成り立つ。また、式 (7) より、 $x_t = x_{t-1}^{(1)} p_1 + p_1 \hat{r}_t^{(1)} = x_{t-1}^{(1)} p_1 + r_t^{(1)}$  の関係が成り立つが、 $x_t$  が整数であり  $r_t^{(1)}$  が式 (8) を満たすことより、 $x_{t-1}^{(1)} p_1 + r_t^{(1)} = \lfloor x_{t-1}^{(1)} p_1 \rfloor$  でなければなら

(注1) : tabled ANS, table-based ANS などとも呼ばれている。

(注2) : 本稿で使用する表記法は、原論文の表記法と異なっている。

(注3) : uABS (uniform ABS) と呼ばれることもある [6]。

ない。つまり、式 (5) が成り立つ。

以上の議論より、 $s_t \in \{0, 1\}$  に対して

$$[x_{t-1}^{(s_t)} p_1] = x_{t-1}^{(s_t)} p_1 + r_t^{(s_t)} \quad (9)$$

が成り立つ。このとき、式 (10) のようにして式 (3) の関係を導ける。ただし、式 (10) の 2 番目と 4 番目の等号では、それぞれ式 (9) と式 (8) を用いている。

$$\begin{aligned} & [(x_{t-1}^{(s_t)} + 1)p_1] - [x_{t-1}^{(s_t)} p_1] \\ &= [(x_{t-1}^{(s_t)} + 1)p_1 - [x_{t-1}^{(s_t)} p_1]] \\ &= [(x_{t-1}^{(s_t)} + 1)p_1 - (x_{t-1}^{(s_t)} p_1 + r_t^{(s_t)})] \\ &= [p_1 - r_t^{(s_t)}] \\ &= s_t \end{aligned} \quad (10)$$

[注意 1] 符号化  $C(s, x)$  として式 (1)(2) の代わりに式 (11)(12) を用い、復号化関数  $D(x_{t-1})$  として式 (3)–(5) の代わりに式 (13)–(15) を用いることができる [5][6].

$$x_{t-1} = x_{t-1}^{(0)} := \left\lfloor \frac{x_t}{p_0} \right\rfloor \quad \text{if } s_t = 0 \quad (11)$$

$$x_{t-1} = x_{t-1}^{(1)} := \left\lfloor \frac{x_t + 1}{p_1} \right\rfloor - 1 \quad \text{if } s_t = 1 \quad (12)$$

$$s_t := [(x_{t-1} + 1)p_1] - [x_{t-1}p_1] \quad (13)$$

$$x_t := x_{t-1}^{(0)} - [x_{t-1}^{(0)} p_1] \quad \text{if } s_t = 0 \quad (14)$$

$$x_t := [x_{t-1}^{(1)} p_1] \quad \text{if } s_t = 1 \quad (15)$$

## 2.2 ABS の平均符号長の評価

式 (4)(5) に式 (9) を代入すると、次の関係が得られる。

$$\frac{x_{t-1}^{(0)}}{x_t} = \frac{1}{p_0} \left( 1 + \frac{r_t^{(0)}}{x_t} \right) \quad (16)$$

$$\frac{x_{t-1}^{(1)}}{x_t} = \frac{1}{p_1} \left( 1 - \frac{r_t^{(1)}}{x_t} \right) \quad (17)$$

データ系列  $s^T$  の符号語  $x_0$  を  $x_0(s^T)$  で表すと、 $x_T = 1$  の場合、 $x_0(s^T)$  のビット長は  $\lg x_0(s^T) = \sum_{t=1}^T \lg(x_{t-1}^{(s_t)}/x_t)$  で与えられる<sup>(注4)</sup>。したがって、 $\lg(x_{t-1}^{(s_t)}/x_t)$  は  $s_t$  の符号化における符号長の増加分を表している。式 (8)(16)(17) より、 $x_t$  がある程度大

きくなれば、非常によい精度で、 $\lg(x_{t-1}^{(0)}/x_t) \approx -\lg p_0$ 、 $\lg(x_{t-1}^{(1)}/x_t) \approx -\lg p_1$  が成り立っていることが分かる。

簡単のため、 $0 < p_1 < 1/2 < p_0 < 1$  の場合を考える<sup>(注5)</sup>。このとき  $\eta = \min\{1/p_0, 1/(2p_1)\}$  と定義すると、 $\eta > 1$  となる。式 (16) より  $x_{t-1}^{(0)} > x_t/p_0 \geq \eta x_t$  が成り立つ。また、式 (17) において  $0 \leq r_t^{(1)} < p_1 < 1/2$  および  $x_t \geq 1$  であることより、 $x_{t-1}^{(1)} > x_t/(2p_1) \geq \eta x_t$  が成り立つ。したがって、任意の  $t, 1 \leq t \leq T$  と  $s$  に対して、 $x_{t-1}^{(s)} > \eta x_t > \eta^{T-t+1} x_T = \eta^{T-t+1}$  を満たす。この関係と式 (8)(16)(17) より、次の上界が得られる。

$$\begin{aligned} \lg \frac{x_{t-1}^{(s)}}{x_t} &< \lg \frac{1}{p_s} \left( 1 + \frac{1}{x_t} \right) = \lg \frac{1}{p_s} + \lg \left( 1 + \frac{1}{x_t} \right) \\ &\leq \lg \frac{1}{p_s} + \frac{\lg e}{x_t} < \lg \frac{1}{p_s} + \frac{\lg e}{\eta^{T-t}} \end{aligned} \quad (18)$$

$x_T = 1$  の場合の平均符号長  $L$  は、式 (18) を用いて次のように評価できる。

$$\begin{aligned} L &= \frac{1}{T} \sum_{s^T \in \mathcal{S}^T} p(s^T) \lg x_0(s^T) \\ &= \frac{1}{T} \sum_{s^T \in \mathcal{S}^T} p(s^T) \sum_{t=1}^T \lg \frac{x_{t-1}^{(s_t)}}{x_t} \\ &= \frac{1}{T} \sum_{t=1}^T \sum_{s_t \in \{0,1\}} p(s_t) \lg \frac{x_{t-1}^{(s_t)}}{x_t} \\ &< \frac{1}{T} \sum_{t=1}^T \sum_{s \in \{0,1\}} p_s \left( \lg \frac{1}{p_s} + \frac{\lg e}{\eta^{T-t}} \right) \\ &= \sum_{s \in \{0,1\}} p_s \lg \frac{1}{p_s} + \frac{\lg e}{T} \sum_{t=1}^T \eta^{t-T} \\ &< H(p) + \frac{\lg e}{T} \frac{\eta}{\eta-1} \end{aligned} \quad (19)$$

ここで  $H(p)$  は情報源のエントロピーである。可変長符号化に対する情報源符号化定理 [2, 定理 5.3.1] より  $L \geq H(p)$  であることと式 (19) より、 $T \rightarrow \infty$  のとき  $L \rightarrow H(p)$  である。

## 3. rANS

前節の ABS では 2 元情報源を対象としていたが、一般の多元情報源を対象とする場合は ANS (Asymmetric Numeral Systems) と呼ばれる。また、前節の ABS の

(注4) :  $x_0(s^T)$  のビット長は MSB の 1 を除けば  $\lfloor \lg x_0(s^T) \rfloor$  ビットになるが、表記を簡単にするために、実数値の  $\lg x_0(s^T)$  で評価している。以後同様である。

(注5) :  $p_0 = p_1 = 1/2$  のときは圧縮できないので考える必要がない。 $p_0 < p_1$  の場合は、式 (11)(12) を用いればよい。

符号化や復号では実数値  $p_0, p_1$  を用いていた。これに対して整数値演算のみを用いて、算術符号の Range Coding のように整数値区間に基づいて符号化や復号を行う ANS を rANS (range variant of ANS) という。

この節では整数  $x$  は任意の桁数まで大きくできるものとする。なお、 $x$  を適切なビットサイズに制限する場合は、次節以降で取り扱う。

### 3.1 rANS の符号化・復号法 [6]

$N = \sum_{s \in \mathcal{S}} N_s$  を満たし、 $N_s/N \approx p(s)$  となるように、整数  $\{N, N_s, s \in \mathcal{S}\}$  を決める。また、情報源シンボルに任意に与えた全順序を  $<$  で示し、 $d_s$  を次式で定義する。

$$d_s = \sum_{\hat{s} < s} N_{\hat{s}} \quad (20)$$

#### A. 符号化関数の定義

符号化関数  $x_{t-1} := C(s_t, x_t)$  を次式で定義する<sup>(注6)</sup>。

$$x_{t-1} := N \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor + d_{s_t} + \text{mod}(x_t, N_{s_t}) \quad (21)$$

#### B. 復号関数の定義

復号関数  $(s_t, x_t) := D(x_{t-1})$  を次式で定義する。

$$s_t := \min \left\{ s : \text{mod}(x_{t-1}, N) < \sum_{i \leq s} N_i \right\} \quad (22)$$

$$x_t := N_{s_t} \left\lfloor \frac{x_{t-1}}{N} \right\rfloor + \text{mod}(x_{t-1}, N) - d_{s_t} \quad (23)$$

#### C. 符号化アルゴリズム

- データ系列  $s^T = s_1 s_2 \cdots s_T$  をセットし、 $x_T \leftarrow 1$  と設定する。
- $t = T, \dots, 2, 1$  と逆順に、 $x_{t-1} \leftarrow C(s_t, x_t)$  を繰り返す。
- $x_0$  が  $s^T$  の符号語となる。

#### D. 復号アルゴリズム

- 符号語  $x_0$  をセットする。
- $t = 1, 2, \dots, T$  の順に、 $(s_t, x_t) \leftarrow D(x_{t-1})$  を繰り返す。
- $s^T = s_1 s_2 \cdots s_T$  が復号系列となる。

式 (21) に対して式 (22)(23) の関係が成り立つことを示す。まず、式 (20) と  $\text{mod}(x_{t-1}, N_{s_t}) < N_{s_t}$  より、

(注6)：剰余演算  $a \bmod b$  を、Duda の論文 [6] と同様に、 $\text{mod}(a, b)$  で表現する。

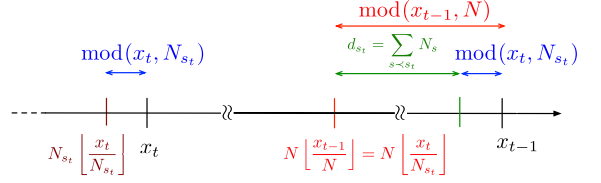


図1 rANS の符号化と復号の関係

次の関係が成り立つ。

$$d_{s_t} + \text{mod}(x_t, N_{s_t}) < \sum_{s < s_t} N_s + N_{s_t} \leq N \quad (24)$$

つまり、式 (21) 右辺の第 2 項と第 3 項の和は  $N$  より小さい。したがって、式 (21) の両辺を  $N$  で割り、floor 関数を取ると次の関係が成り立つ。

$$\left\lfloor \frac{x_{t-1}}{N} \right\rfloor = \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor \quad (25)$$

式 (21)(24)(25) を用いると、次の関係が導かれる。

$$\begin{aligned} \text{mod}(x_{t-1}, N) &= x_{t-1} - N \left\lfloor \frac{x_{t-1}}{N} \right\rfloor \\ &= x_{t-1} - N \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor \\ &= d_{s_t} + \text{mod}(x_t, N_{s_t}) \end{aligned} \quad (26)$$

$$< \sum_{s < s_t} N_s + N_{s_t} \quad (27)$$

式 (27) より、 $s_t$  が式 (22) で復号できることが分かる。また、式 (23) は、式 (25)(26) を用いて次のように導くことができる。

$$\begin{aligned} x_t &= N_{s_t} \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor + \text{mod}(x_t, N_{s_t}) \\ &= N_{s_t} \left\lfloor \frac{x_{t-1}}{N} \right\rfloor + \text{mod}(x_t, N_{s_t}) \\ &= N_{s_t} \left\lfloor \frac{x_{t-1}}{N} \right\rfloor + \text{mod}(x_{t-1}, N) - d_{s_t} \end{aligned} \quad (28)$$

なお、数直線上の整数値  $x_t$  と  $x_{t-1}$  に対する式 (21)(25) (26)(28) の関係を図示すると、図 1 のような関係となっている。

### 3.2 rANS の平均符号長の評価

情報源の確率分布  $p = \{p(s) \mid s \in \mathcal{S}\}$  と、 $q(s) = N_s/N$  によって定まる確率分布  $q = \{q(s) \mid s \in \mathcal{S}\}$  を用いて、rANS の平均符号長を評価する。

式 (25) より、次の関係が成り立つ。

$$\begin{aligned}\frac{x_{t-1}}{N} &\approx \frac{x_t}{N_{s_t}} \\ \frac{x_{t-1}}{x_t} &\approx \frac{N}{N_{s_t}} = \frac{1}{q(s_t)}\end{aligned}\quad (29)$$

式 (29) の  $\approx$  が等号で成り立つ場合を考える。このとき、 $x_T = 1$  としてデータ系列  $s^T = s_1 s_2 \cdots s_T$  を式 (21) を用いて符号化したときの符号語  $x_0$  を  $x_0(s^T)$  で表すと、 $x_0(s^T)$  のビット長は次式で与えられる。

$$\begin{aligned}\lg x_0(s^T) &= \lg \prod_{t=1}^T \frac{x_{t-1}}{x_t} + \lg x_T \\ &= \sum_{t=1}^T \lg \frac{x_{t-1}}{x_t} \\ &= \sum_{t=1}^T \lg \frac{1}{q(s_t)}\end{aligned}$$

したがって、平均符号長  $L$  は次式で与えられる。

$$\begin{aligned}L &= \frac{1}{T} \sum_{s^T \in \mathcal{S}^T} p(s^T) \lg x_0(s^T) \\ &= \frac{1}{T} \sum_{s^T \in \mathcal{S}^T} p(s^T) \sum_{t=1}^T \lg \frac{1}{q(s_t)} \\ &= \frac{1}{T} \sum_{t=1}^T \sum_{s_t \in \mathcal{S}} p(s_t) \lg \frac{1}{q(s_t)} \\ &= \sum_{s \in \mathcal{S}} p(s) \lg \frac{1}{q(s)} \\ &= H(p) + D(p||q)\end{aligned}\quad (30)$$

ここで、 $H(p)$  と  $D(p||q)$  は情報源のエントロピーと相対エントロピーである。

次に、式 (29) の近似の程度を見積もる。式 (25) より

$$\frac{x_{t-1}}{N} - 1 < \frac{x_t}{N_{s_t}}, \quad \frac{x_{t-1}}{N} > \frac{x_t}{N_{s_t}} - 1$$

の関係が成り立つことから、 $x_{t-1}/x_t$  は次の上界と下界を持つ。

$$\frac{N}{N_{s_t}} - \frac{N}{x_t} < \frac{x_{t-1}}{x_t} < \frac{N}{N_{s_t}} + \frac{N}{x_t}\quad (31)$$

この式より、 $x_t$  が次式を満たすとき、式 (29) は非常によい近似になっている。

$$x_t \gg N\quad (32)$$

式 (31) の誤差による符号長への影響を評価する。式

(31) の右側の不等式より、次の関係が成り立つ。

$$\begin{aligned}\lg \frac{x_{t-1}}{x_t} &< \lg \left( \frac{N}{N_{s_t}} + \frac{N}{x_t} \right) \\ &= \lg \frac{N}{N_{s_t}} + \lg \left( 1 + \frac{N_{s_t}}{x_t} \right)\end{aligned}\quad (33)$$

$$\begin{aligned}&< \lg \frac{1}{q(s_t)} + \lg \left( 1 + \frac{N}{x_t} \right) \\ &< \lg \frac{1}{q(s_t)} + (\lg e) \frac{N}{x_t}\end{aligned}\quad (34)$$

式 (34) の第 2 項が、式 (29) の近似誤差による符号長のロスの上界を表している。 $s^T$  全体での符号長のロスを  $l_{\text{loss}}(s^T)$  で表すと、その上界は次式となる。

$$l_{\text{loss}}(s^T) < (N \lg e) \sum_{t=1}^T \frac{1}{x_t}$$

$\eta = \left( \min_{s \in \mathcal{S}} \frac{N}{N_s} \right) - \frac{N}{A} > 1$  を満たすように  $A$  を選び、 $x_T = A$  とすると、式 (31) の左側の不等式と  $A = x_T < x_{T-1} < \cdots < x_{t-1}$  より、 $x_{t-1} > \eta^{T-t+1} A$  となる。したがって、 $x_T = 1$  の代わりに  $x_T = A$  を用いると、その増加分  $\lg A$  を合わせて  $l_{\text{loss}}(s^T)$  は次の上界を持つ。

$$\begin{aligned}l_{\text{loss}}(s^T) &< \lg A + (N \lg e) \sum_{t=1}^T \frac{1}{\eta^{T-t} A} \\ &< \lg A + \frac{N \lg e}{A} \frac{\eta}{\eta - 1}\end{aligned}\quad (35)$$

式 (30)(35) より、 $x_T = A$  を用いたときの平均符号長  $L$  は次式を満たす。

$$L < H(p) + D(p||q) + \frac{1}{T} \left( \lg A + \frac{N \lg e}{A} \frac{\eta}{\eta - 1} \right)\quad (36)$$

式 (36) より、 $T \rightarrow \infty$  のとき  $L \rightarrow H(p) + D(p||q)$  である。

#### 4. 有限桁演算を用いる rANS

前節で説明した rANS では、 $T$  が大きいとき  $t$  がゼロに近づくに連れて  $x_t$  が非常に大きな整数値になり、実用的でなくなる欠点がある。これに対して、 $T$  がどのように大きくても有限桁の  $x_t$  で符号化・復号が行えるようにしたものを、ストリーム符号化・復号 (stream coding/decoding) という。

この節では、前節の rANS をストリーム型に修正し

た Townsend [28] の符号化・復号アルゴリズムを紹介し、その平均符号長を理論的に評価する。なお、この節では  $R$  を整数とし、 $N = \sum_{s \in S} N_s = 2^R$  を満たすように  $N$  と  $N_s$  が決められているものとする。

#### 4.1 有限桁演算を用いる rANS の符号化・復号法 [6]

式 (21)–(23) より、 $N = 2^R$  の場合の符号化関数と復号関数は次のように定義される。

##### A. 符号化関数の定義

符号化関数  $x_{t-1} := C(s_t, x_t)$  を次式で定義する。

$$x_{t-1} := 2^R \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor + d_{s_t} + \text{mod}(x_t, N_{s_t}) \quad (37)$$

##### B. 復号関数の定義

復号関数  $(s_t, x_t) := D(x_{t-1})$  を次式で定義する。

$$s_t := \min \left\{ s : \text{mod}(x_{t-1}, 2^R) < \sum_{i \leq s} N_i \right\}$$

$$x_t := N_{s_t} \left\lfloor \frac{x_{t-1}}{2^R} \right\rfloor + \text{mod}(x_{t-1}, 2^R) - d_{s_t}$$

このとき、式 (37) で定まる  $x_{t-1}$  に次の制限を加えることを考える。

$$2^{r_a - r_b} \leq x_{t-1} < 2^{r_a} \quad (38)$$

ただし、 $r_a$  と  $r_b$  は  $r_a - r_b > R$  を満たす整数とする。このとき、式 (37)(38) より、次の関係が成り立つ。

$$2^{r_a - r_b} \leq 2^R \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor + d_{s_t} + \text{mod}(x_t, N_{s_t}) < 2^{r_a}$$

$$2^{r_a - r_b - R} \leq \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor + \frac{d_{s_t} + \text{mod}(x_t, N_{s_t})}{2^R} < 2^{r_a - R}$$

$$2^{r_a - r_b - R} \leq \left\lfloor \frac{x_t}{N_{s_t}} \right\rfloor < 2^{r_a - R} \quad (39)$$

$$2^{r_a - r_b - R} \leq \frac{x_t}{N_{s_t}} < 2^{r_a - R}$$

$$N_{s_t} 2^{r_a - r_b - R} \leq x_t < N_{s_t} 2^{r_a - R} \quad (40)$$

式 (39) は、式 (24) より  $(d_{s_t} + \text{mod}(x_t, N_{s_t}))/2^R < 1$  であることから成り立つ。

式 (37) の符号化で得られる  $x_{t-1}$  が式 (38) を満たすためには、 $x_t$  は式 (40) を満たしていなければならない。そこで、 $x_t$  が式 (40) を満たしていないときは、 $x_t$  の下位ビットをスタックに Push したのち、その下位ビットを  $x_t$  から取り除くことで、 $x_t$  が式 (40) を満たすようにする。その後、式 (37) の符号化を行う。逆に

復号時は、式 (38) を満たしている  $x_{t-1}$  から復号される  $x_t$  は式 (40) の範囲となる。そこで、スタックから  $x_t$  の下位ビットを Pop して  $x_t$  の下位ビットに追加することで、 $x_t$  が  $2^{r_a - r_b} \leq x_t < 2^{r_a}$  を満たすようにする。その後、 $s_{t+1}$  の復号のステップに進む。

符号化・復号と Push-Pop の関係を図 2 に示す。また、Push と Pop の操作は以下ようになる。ただし、式 (41)(42) における  $u$  はスタックであり、stack-push ( $u, \tilde{u}$ ) はスタック  $u$  に  $\tilde{u}$  を Push することを示し、stack-pop ( $u$ ) はスタック  $u$  の上部から適切な長さ分の  $u_{\text{top}}$  を Pop することを示している。

##### $x_t$ に対する Push 操作

While  $x_t \geq N_{s_t} 2^{r_a - R}$  :

$$u \leftarrow \text{stack-push}(u, \text{mod}(x_t, 2^{r_b})) \quad (41)$$

$$x_t \leftarrow \left\lfloor \frac{x_t}{2^{r_b}} \right\rfloor$$

##### $x_t$ に対する Pop 操作

While  $x_t < 2^{r_a - r_b}$  :

$$u_{\text{top}} \leftarrow \text{stack-pop}(u) \quad (42)$$

$$x_t \leftarrow 2^{r_b} x_t + u_{\text{top}}$$

上記の Push 操作において式 (41) の Push が行われる回数は、 $s_t$  と  $x_t$  の値に依存する。 $2^{r_a - r_b} \leq x_t < 2^{r_a}$  である  $x_t$  が、最初から  $x_t < N_{s_t} 2^{r_a - R}$  を満たす場合は (つまり  $2^{-r_b} < p(s_t) \approx q(s_t) = N_{s_t}/N = N_{s_t} 2^{-R}$  で、 $2^{r_a - r_b} \leq x_t < N_{s_t} 2^{r_a - R}$  となる場合は) Push 回数は 0 回となる。逆に、 $2^{-r_b} > p(s_t) \approx q(s_t) = N_{s_t}/N = N_{s_t} 2^{-R}$  で  $N_{s_t} 2^{r_a - R} \leq x_t 2^{-r_b} < 2^{r_a - r_b}$  の場合は、2 回以上 Push される。なお、 $x_{t-1}$  に対して式 (32) の関係を満たすためには、式 (38) より  $r_a, r_b, R$  は次の関係を満たさなければならない。

$$2^{r_a - r_b} \gg 2^R \quad (43)$$

以上より、符号化アルゴリズムと復号アルゴリズムは下記ようになる。

##### C. 符号化アルゴリズム

- a. データ系列  $s^T = s_1 s_2 \cdots s_T$  をセットし、 $x_T \leftarrow 2^{r_a - r_b}$  に設定する。
- b.  $t = T, \dots, 2, 1$  と逆順に下記の (i)(ii) を繰り返す。
  - (i)  $x_t$  に対する Push 操作。
  - (ii)  $x_{t-1} \leftarrow C(s_t, x_t)$ 。
- c.  $x_0$  およびスタック  $u$  が  $s^T$  の符号語となる。

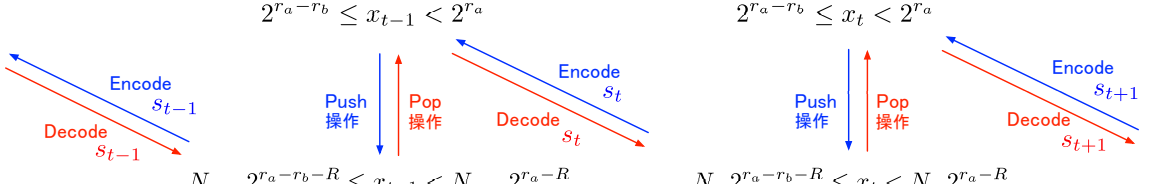


図2 rANS のストリーム符号化・復号における Push 操作と Pop 操作の関係

#### D. 復号アルゴリズム

- a. 符号語 ( $x_0$  とスタック  $u$ ) をセットする.
- b.  $t = 1, 2, \dots, T$  の順に下記の (iii)(iv) を繰り返す.
  - (iii)  $(s_t, x_t) \leftarrow D(x_{t-1})$ .
  - (iv)  $x_t$  に対する Pop 操作.
- c.  $s^T = s_1 s_2 \dots s_T$  が復号系列となる.

[注意2] 同様の手法を第2節の ABS に用いれば、ABS を有限桁演算のストリーム符号化・復号をすることができる。

##### 4.2 平均符号長の評価

時刻  $t$  で  $s_t$  を符号化したときの符号長の増加量  $l(s_t)$  は、 $l(s_t) = \lg x_{t-1} - \lg x_t = \lg(x_{t-1}/x_t)$  で与えられるため、式 (33) の上界が使える。式 (33) の  $x_t$  に式 (40) の左側の不等式を用いると、 $l(s_t)$  は次の上界を満たす。

$$\begin{aligned} l(s_t) &= \lg \frac{x_{t-1}}{x_t} < \lg \frac{N}{N_{s_t}} + \lg \left( 1 + \frac{N_{s_t}}{x_t} \right) \\ &\leq \lg \frac{N}{N_{s_t}} + \lg \left( 1 + \frac{1}{2^{r_a-r_b-R}} \right) \\ &< \lg \frac{1}{q(s_t)} + \frac{\lg e}{2^{r_a-r_b-R}} \end{aligned}$$

したがって、 $x_T = 2^{r_a-r_b}$  に設定して、 $s^T$  を符号化したときの符号語のトータルビット長  $l(s^T)$  は次式で与えられる。

$$\begin{aligned} l(s^T) &= \sum_{t=1}^T l(s_t) + \lg x_T \\ &< \sum_{t=1}^T \left( \lg \frac{1}{q(s_t)} + \frac{\lg e}{2^{r_a-r_b-R}} \right) + r_a - r_b \\ &= \sum_{t=1}^T \lg \frac{1}{q(s_t)} + \left( \frac{T \lg e}{2^{r_a-r_b-R}} + r_a - r_b \right) \quad (44) \end{aligned}$$

式 (44) より、平均符号長  $L = (1/T) \sum p(s^T) l(s^T)$  は式 (30) と同様の計算により、次式で与えられる。

$$L < H(p) + D(p||q) + \frac{\lg e}{2^{r_a-r_b-R}} + \frac{r_a - r_b}{T} \quad (45)$$

$r_a, r_b, R$  が式 (43) を満たし、かつ  $T \gg r_a - r_b$  のとき、式 (45) の右辺第3項と第4項は十分に小さくなる。

## 5. tANS

本節では tANS (tabled variant of ANS) を取り扱う。tANS は単に ANS と呼ばれることも多い [13] [14]。tANS は  $|S| \geq 2$  の情報源に対して、rANS と同様に整数演算のみで符号化と復号が行える。前節で取り扱った有限桁演算を用いる rANS では、データ系列  $s^T$  が  $x_0$  およびスタック  $u$  に符号化されるが、 $s_t$  ごとに符号語が決まるわけではない。これに対して、tANS では  $s_t$  の符号語が  $s_t$  と  $x_t$  の値から一意に決まるように工夫されている。その結果、tANS は後に述べるように算術演算無しで全て表 (table) を用いて符号化と復号が行える特徴がある。

前節と同様にある自然数  $R$  に対して  $N = 2^R$  とし、 $N = \sum_{s \in S} N_s$  を満たすように、 $\{N_s, s \in S\}$  が決められているものとする。tANS では、以下のように定義された整数の状態集合  $\mathcal{X}, \mathcal{X}_s, \mathcal{Y}$  を用いて符号化と復号が行われる。

1.  $\mathcal{X}$ : 符号化および復号で使用する状態集合。  
 $\mathcal{X} = \{N, N+1, \dots, 2N-1\}$ ,  $N = |\mathcal{X}|$  とする。
2.  $\mathcal{X}_s$ :  $s \in S$  に対応する状態集合。  $N_s = |\mathcal{X}_s|$  とし、 $\mathcal{X}_s \cap \mathcal{X}_{s'} = \emptyset$  for  $s \neq s'$ ,  $\mathcal{X} = \bigcup_{s \in S} \mathcal{X}_s$  を満たす任意に定めた集合。  $N = \sum_{s \in S} N_s$ .
3.  $\mathcal{Y}_s$ :  $s \in S$  に対応するもう一つの状態集合。  
 $\mathcal{Y}_s = \{N_s, N_s+1, \dots, 2N_s-1\}$ ,  $N_s = |\mathcal{Y}_s|$  とする。

各  $s \in S$  に対して  $N_s = |\mathcal{X}_s| = |\mathcal{Y}_s|$  の関係から、 $x \in \mathcal{X}_s$  と  $y \in \mathcal{Y}_s$  は一対一に対応付けることができる。さらに、 $\mathcal{X}_s$  が上記2の条件を満たしていることから、 $s \in S$  と  $y \in \mathcal{Y}_s$  のペア  $(s, y)$  は  $x \in \mathcal{X}$  と一対一に対応付けることができる。その一対一対応を関数  $\tilde{C}$  と  $\tilde{D}$  を用いて表す<sup>(注7)</sup>。

(注7): 前節までに使用した符号化関数  $x_{t-1} := C(s_t, x_t)$  および復号関



## A. 符号化関数の定義

各  $s \in \mathcal{S}$  に対して, 全単射な関数  $\tilde{C}(s, \cdot) : \mathcal{Y}_s \rightarrow \mathcal{X}_s$ . 各  $s \in \mathcal{S}$  に対して  $y \in \mathcal{Y}_s$  のとき,  $x := \tilde{C}(s, y) \in \mathcal{X}_s$  である.

## B. 復号関数の定義

各  $x \in \mathcal{X}_s$  に対して, 全単射な関数  $\tilde{D} : \mathcal{X}_s \rightarrow \{s\} \times \mathcal{Y}_s$ .  $x \in \mathcal{X}_s \subset \mathcal{X}$  のとき,  $(s, y) := \tilde{D}(x) \in \{s\} \times \mathcal{Y}_s$  である.

### 5.1 tANS の符号化・復号法 [6][13][14]

時刻  $t$  における状態, データシンボル, 符号語, 符号長をそれぞれ  $x_t, s_t, b_t, k_t$  で表現すると, 符号化は  $(x_t, s_t)$  から  $(b_t, x_{t-1})$  を求める処理であり, 復号は  $(x_{t-1}, b_t)$  から  $(s_t, x_t)$  を求める処理である. tANS では, これらの処理は  $(y_{t-1}, k_t)$  を経由して行うようになっている [6][13]. ただし, 関数  $\tilde{C}$  と  $\tilde{D}$  は与えられているものとする.

## C. 符号化アルゴリズム

- データ系列  $s^T = s_1 s_2 \cdots s_T$  をセットし,  $x_T \in \mathcal{X}$  を 1 つ選ぶ.
- $t = T, \dots, 2, 1$  と逆順に下記の演算を繰り返す.

$$k_t \leftarrow \left\lfloor \lg \frac{x_t}{N_{s_t}} \right\rfloor \quad (46)$$

$$b_t \leftarrow \text{mod}(x_t, 2^{k_t}) \quad (47)$$

$$y_{t-1} \leftarrow \left\lfloor \frac{x_t}{2^{k_t}} \right\rfloor \quad (48)$$

$$x_{t-1} \leftarrow \tilde{C}(s_t, y_{t-1}) \quad (49)$$

- $x_0 b_1 b_2 \cdots b_T$  が  $s^T$  の符号語系列となる.

## D. 復号アルゴリズム

- 符号語系列  $x_0 b_1 b_2 \cdots b_T$  から,  $x_0$  と  $\mathbf{b} \leftarrow b_1 b_2 \cdots b_T$  をセットする.
- $t = 1, 2, \dots, T$  の順に下記の演算を繰り返す.

$$(s_t, y_{t-1}) \leftarrow \tilde{D}(x_{t-1})$$

$$k_t \leftarrow R - \lfloor \lg y_{t-1} \rfloor \quad (50)$$

$$b_t \leftarrow \mathbf{b} \text{ の先頭 } k_t \text{ ビット} \quad (51)$$

$$\mathbf{b} \leftarrow \mathbf{b} \text{ の先頭の } k_t \text{ ビットを取り除いたもの}$$

$$x_t \leftarrow 2^{k_t} y_{t-1} + b_t \quad (52)$$

- $s^T = s_1 s_2 \cdots s_T$  が復号系列となる.

数  $(s_t, x_t) := D(x_{t-1})$  とは意味合いが異なる関数であるため,  $(\tilde{C}, \tilde{D})$  という表記を用いている.

符号語系列に含まれる  $x_0$  は, 例えば固定長符号を用いると  $\lg N = R$  ビットで表現できる. 式 (47) より,  $k_t$  は符号語  $b_t$  のビット長を表している. 復号時には, 式 (50) で  $k_t$  が求まるため, 式 (51) で符号語系列  $\mathbf{b}$  から  $b_t$  を切り出すことができる.

[注意 3] 符号化・復号アルゴリズムで使われている演算  $\lfloor \lg a \rfloor$  の値は,  $a$  の整数部分の MSB の 1 を除いた桁数であるため, 対数計算を行わずに容易に求めることができる.

[注意 4] 式 (47) は式 (41) に対応した操作であるが, 式 (41) では  $s_t$  や  $x_t$  の値に関係なく  $2^{b_t}$  で剰余を取っていたのに対し, 式 (47) では  $s_t$  と  $x_t$  の値によって定まる  $2^{k_t}$  で剰余を取ることで,  $s_t$  ごとに符号語  $b_t$  が定まるように工夫されている.

まず, 式 (48) で求まる  $y_{t-1}$  が  $y_{t-1} \in \mathcal{Y}_{s_t}$  を満たす (つまり,  $N_{s_t} \leq y_{t-1} < 2N_{s_t}$  を満たす) ことを示す. 式 (46) より, 以下の関係が成り立つ.

$$2^{k_t} \leq \frac{x_t}{N_{s_t}} < 2^{k_t+1}$$

$$N_{s_t} \leq \frac{x_t}{2^{k_t}} < 2N_{s_t}$$

$$N_{s_t} \leq y_{t-1} = \left\lfloor \frac{x_t}{2^{k_t}} \right\rfloor < 2N_{s_t} \quad (53)$$

式 (53) は  $N_{s_t}$  が整数であることによる. よって,  $y_{t-1} \in \mathcal{Y}_{s_t}$  を満たすため, 式 (49) で関数  $\tilde{C}(s_t, y_{t-1})$  が使用でき,  $x_{t-1} = \tilde{C}(s_t, y_{t-1}) \in \mathcal{X}_{s_t}$  となる.

$b_t$  と  $y_{t-1}$  が式 (47)(48) で与えられることより,  $x_t$  は式 (52) で求められる. 次に  $k_t$  が式 (50) で求まることを示す.  $x_t \in \mathcal{X}$  (つまり  $N \leq x_t < 2N$ ) と式 (52) および  $N = 2^R$  の関係から以下の不等式が成り立つ.

$$2^R \leq 2^{k_t} y_{t-1} + b_t < 2^{R+1}$$

$$2^{R-k_t} \leq y_{t-1} + \frac{b_t}{2^{k_t}} < 2^{R-k_t+1}$$

$$2^{R-k_t} \leq y_{t-1} < 2^{R-k_t+1} \quad (54)$$

$$R - k_t \leq \lg y_{t-1} < R - k_t + 1$$

よって, 式 (50) で  $k_t$  が求まる. なお, 式 (54) は,  $2^{R-k_t}$  と  $y_{t-1}$  が整数であり, 式 (47) より  $0 \leq b_t/2^{k_t} < 1$  であることによる.

[注意 5] 上記では状態数  $N = 2^R$  の場合を取り扱っているが, tANS は  $N$  が 2 のべき乗でない一般の整数の場合に対しても使用することができる [6]. ただし, 一般の  $N$  の場合は,  $k_t = \lfloor \lg(N/y_{t-1}) \rfloor$  と  $k_t = \lceil \lg(N/y_{t-1}) \rceil - 1$

の場合があるため、復号では  $N \leq 2^{k_t} y_{t-1} + b_t < 2N$  を満たす方の  $k_t$  を使用する必要がある。

[注意 6]  $x_t = x, N \leq x < 2N$  と  $s_t = s, s \in \mathcal{S}$  の全ての場合に対して、式 (46)–(49) および (50)–(52) を予め計算しておき、それらの結果を表 (table) として保持しておけば、毎回算術計算を行うことなく符号化と復号を行うことができる。そのことから、tANS (tabled-variant ANS) と呼ばれている。

## 5.2 tANS の平均符号長の評価

符号化の初期状態  $x_T \in X$  の選び方や、符号語に含まれる  $x_0$  のビット長  $R$  による平均符号長の増加は、データ系列長  $T$  が長くなるに連れてゼロに収束する。簡単のため、この節ではそれらを見捨て、定常状態における平均符号長を求めることにする。

$x \in X$  に対する定常確率を  $Q(x)$  で表し、 $q(s) = N_s/N$  とする。最初に、式 (46) を用いて平均符号長  $L$  を評価すると、次のようになる。

$$\begin{aligned} L &= \sum_{s \in \mathcal{S}} \sum_{x \in X} p(s) Q(x) \left\lceil \lg \frac{x}{N_s} \right\rceil \\ &\leq \sum_{s \in \mathcal{S}} \sum_{x \in X} p(s) Q(x) \lg \frac{x}{N_s} \\ &= \sum_{s \in \mathcal{S}} \sum_{x \in X} p(s) Q(x) \left( \lg \frac{N}{N_s} + \lg \frac{x}{N} \right) \\ &= \sum_{s \in \mathcal{S}} p(s) \lg \frac{N}{N_s} + \sum_{x \in X} Q(x) \lg \frac{x}{N} \quad (55) \end{aligned}$$

$$\begin{aligned} &\leq^* \sum_{s \in \mathcal{S}} p(s) \lg \frac{1}{q(s)} + \lg \frac{\sum_{x \in X} Q(x)x}{N} \\ &= H(p) + D(p||q) + \lg \frac{E[X]}{N} \quad (56) \end{aligned}$$

ここで、 $\leq^*$  は  $\lg$  関数に対するイェンセンの不等式による。また、 $E[X]$  は  $x$  の定常状態における期待値である。 $N \leq x \leq 2N-1$  より、緩い上界として  $\lg(E[X]/N) < 1$  が成り立つ。

前節までの ABS や rANS と異なり、tANS では  $s_t$  1 シンボルごとに式 (46) で定まる  $k_t$  ビットの符号語が出力される。そのため、 $s^T = s_1 s_2 \cdots s_T$  全体に対して一度に 1 つの符号語を割り当てる場合に比べてロスが生じる。以下では、tANS の条件 ( $N \leq x_{t-1} < 2N, N_{s_t} \leq y_{t-1} < 2N_{s_t}$ ) の下で、このロスの大きさを見積もる。

$x_{t-1}$  の代わりに、 $\tilde{x}_{t-1} := (N/N_{s_t})y_{t-1}$  で定義される  $\tilde{x}_{t-1}$  を用いると

$$N \leq \tilde{x}_{t-1} = \frac{N}{N_{s_t}} y_{t-1} < 2N \quad (57)$$

を満たすが、 $\tilde{x}_{t-1}$  は一般に整数値にならない。そのため、tANS では関数  $\tilde{C}$  を用いて、 $y_{t-1}$  を  $N \leq x_{t-1} < 2N$  を満たす整数  $x_{t-1}$  に写像している。

$s^T$  を一度に符号化する場合に対応して、各  $s_t$  の符号化ごとに実数値の符号長を許す理想的な場合を考える。このときの符号長  $l(s_t)$  は、rANS の場合と同様に、 $l(s_t) = \lg(x_{t-1}/y_{t-1})$  で与えられる。 $\tilde{x}_{t-1}$  を用いる場合は、式 (57) より  $l(s_t) = \lg(\tilde{x}_{t-1}/y_{t-1}) = \lg(N/N_{s_t}) = \lg 1/q(s_t)$  となり、 $s_t$  の値のみに依存し、 $y_{t-1}$  の値に依存しない。したがって、この場合の平均符号長を  $\tilde{L}^*$  で表すと、 $\tilde{L}^*$  は次式で与えられる。

$$\begin{aligned} \tilde{L}^* &= \sum_{s \in \mathcal{S}} p(s) \lg \frac{N}{N_s} \\ &= H(p) + D(p||q) \end{aligned}$$

しかし、 $x_{t-1} = \tilde{C}(s, y_{t-1})$  の場合は、 $x_{t-1}/y_{t-1}$  の値が  $N/N_{s_t}$  から少しずれるため、 $\tilde{L}^*$  より平均符号長が大きくなる。 $x_{t-1} = \tilde{C}(s_t, y_{t-1})$  の場合の実数値の符号長は  $l(s_t) = \lg(\tilde{C}(s_t, y_{t-1})/y_{t-1})$  で与えられ、 $s_t$  と  $y_{t-1}$  の両方に依存するため、その平均符号長  $L^*$  は次式で与えられる。

$$\begin{aligned} L^* &= \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) \lg \frac{\tilde{C}(s, y)}{y} \\ &= \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) \left( \lg \frac{N}{N_s} + \lg \frac{\tilde{C}(s, y) N_s}{N y} \right) \\ &=^*2 \sum_{s \in \mathcal{S}} p(s) \lg \frac{N}{N_s} + \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) \lg \frac{\tilde{C}(s, y)}{N} \\ &\quad - \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) \lg \frac{y}{N_s} \\ &=^*3 \sum_{s \in \mathcal{S}} p(s) \lg \frac{N}{N_s} + \sum_{x \in X} Q(x) \lg \frac{x}{N} \\ &\quad - \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) \lg \frac{y}{N_s} \quad (58) \end{aligned}$$

ここで、 $=^*2$  は  $\sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) = p(s)$  の関係による。また、 $=^*3$  は、 $(s, y)$  と  $x = \tilde{C}(s, y)$  が一対一対応であることによる。

式 (55) と式 (58) より、 $L^*$  に対する tANS の平均符号長  $L$  のロスは次の上界を持つ。

$$\begin{aligned}
L - L^* &\leq \sum_{s \in S} \sum_{y \in \mathcal{Y}_s} Q(\tilde{C}(s, y)) \lg \frac{y}{N_s} \\
&\leq^* \sum_{s \in S} p(s) \lg \frac{\sum_{y \in \mathcal{Y}_s} \frac{Q(\tilde{C}(s, y))}{p(s)} y}{N_s} \\
&= \sum_{s \in S} p(s) \lg \frac{E[Y|s]}{N_s} \quad (59)
\end{aligned}$$

ここで、 $E[Y|s]$  は  $s_t = s \in S$  の条件下での  $y_{t-1}$  の期待値である。式 (53) の関係より式 (59) において  $E[Y|s]/N_s < 2$  であることから、緩い上界として  $L - L^* < 1$  が成り立つ。

式 (56) は、任意の  $\tilde{C}$  に対して成り立つ上界であるが、 $\tilde{C}$  を適切に選ぶことにより、平均符号長  $L$  をより小さくすることができる。性能の良い  $\tilde{C}$  の構成法は、文献 [13] [14] [27] などて提案されている。tANS の理想的な定常確率<sup>(注8)</sup>は

$$Q^*(x) = \lg \frac{x+1}{x}$$

で与えられるが [22] [25] [26]、文献 [27] ではこの  $Q^*(x)$  に基づく  $\tilde{C}$  の構成法が示されている。

なお、ある定数  $\alpha \geq 0$  に対して  $Q(x)$  が全ての  $x \in \mathcal{X}$  で  $Q(x) \leq Q^*(x) + (\alpha/N^2)$  を満たすとき、適切な  $\tilde{C}$  を用いれば平均符号長  $L$  は次式を満たす [34]。

$$L \leq H(p) + D(p||q) + \frac{\alpha}{N}$$

さらに、 $q(s) = N/N_s = p(s)$  の場合に対して、定常確率  $Q(x)$  が  $Q(x) = Q^*(x) + O(1/N^2)$  を満たす  $\tilde{C}$  の構成法が文献 [26] [34] で与えられている。

## 6. ま と め

本稿では、ANS の各種類に対する符号化アルゴリズムと復号アルゴリズムを詳しく紹介すると共に、それらの平均符号長の上界を導出した。

本稿では復号時に情報源の確率分布  $p = \{p(s)\}$  が既知であるとしたが、 $p$  が未知の場合は、符号語に  $\{p(s)\}$  または  $\{N_s\}$  の情報を含める必要がある。あるいは、符号語に  $\{p(s)\}$  や  $\{N_s\}$  の情報を含めずに、動的ハフマン符号 [35] のように頻度分布を利用することが考えられる。 $s_t^1 = s_1 s_2 \cdots s_t$  の頻度分布に基づいて推定した確率分布  $\hat{p}_t = \{\hat{p}_t(s)\}$  を使って  $s_{t+1}$  を復号する場合は、符号化でも  $s^T$  から確率分布

$\hat{p}_t, t = 1, 2, \dots, T$  を先に求めて、 $t = T, \dots, 2, 1$  と逆順に  $\hat{p}_t$  を使って符号化する必要がある。また本稿では、符号化を降順 ( $s_t, t = T, T-1, \dots, 1$ ) に行い、復号を昇順 ( $s_t, t = 1, 2, \dots, T$ ) で行うとして説明したが、符号化を昇順、復号を降順にすることもできる。

ANS は、算術符号の Range Code と同様に整数の算術演算を用いて符号化および復号を行うように作られている。しかし、算術演算を用いない場合でも、符号化と復号を互いに逆順で行うことで高性能な圧縮を行うことが可能である [32]–[34]。また、ANS を簡易暗号や暗号用の乱数発生器として利用する研究もなされている [8]–[10] [12]。

**謝辞** 本研究は JSPS 科研費 24K07487 の助成を受けたものである。

## 文 献

- [1] 山本博資, 岩田賢一, “ANS の符号化・復号アルゴリズムとその圧縮性能の評価,” 第 12 回シャノン理論ワークショップ予稿集 (STW2023), pp.36-43, Oct. 2023.
- [2] 山本博資, 古賀弘樹, 有村光晴, 岩本貢 (共訳), 情報理論 基礎と広がり, 共立出版, 2012.
- [3] D. Salomon and G. Motta, *Handbook of data compression*, 5th Ed., Springer, 2010.
- [4] J. Duda, “Optimal encoding on discrete lattice with translational invariant constraints using statistical algorithms,” arXiv:0710.3861v5, Nov. 2, 2008.
- [5] J. Duda, “Asymmetric numeral systems,” arXiv:0902.0271v5, May 21, 2009.
- [6] J. Duda, “Asymmetric numeral systems: entropy coding combining speed of Huffman coding with compression rate of arithmetic coding,” arXiv:1311.2540v2, Jan. 6, 2014.
- [7] J. Duda, K. Tahboub, et al., “The use of asymmetric numeral systems as an accurate replacement of Huffman coding,” Proc. of 2015 Picture Coding Symposium (PCS2015), pp. 65–69, May, 2015.
- [8] J. Duda and M. Niemic, “Lightweight compression with encryption based on asymmetric numeral systems,” arXiv:1612.04662v1, Dec. 14, 2016.
- [9] S. Camtepe, J. Duda, et al., “Compcrypt-lightweight ANS-based compression and encryption,” IEEE Trans. on Inform. Foren. and Security, vol. 16, pp. 3859–3873, July, 2021.
- [10] S. Camtepe, J. Duda, et al., “ANS–based compression and encryption with 128–bit security,” Int. Journal of Inform. Security, vol. 21, pp.1051–1067, July, 2022.
- [11] J. Duda, “Encoding of probability distributions for asymmetric numeral systems,” arXiv:2106.06438v4, July 4, 2022.
- [12] J. Pieprzyk, M. Pawlowski, et al., “Pseudorandom bit generation with asymmetric numeral systems,” Cryptology ePrint Archive, Paper 2022/005, 2022
- [13] J. Pieprzyk, J. Duda, et al., “Compression optimality of asymmetric numeral systems,” arXiv:2209.02228v1, Sep 6, 2022.
- [14] J. Pieprzyk, J. Duda, et al., “The compression optimality of asym-

(注8)：定常確率  $Q^*(x)$  が実現できたときは最適になる。

- metric numeral systems," *Entropy*, vol. 25, no. 4, article no. 672, April, 2023.
- [15] Wikipedia, "Asymmetric numeral systems," [https://en.wikipedia.org/wiki/Asymmetric\\_numeral\\_systems](https://en.wikipedia.org/wiki/Asymmetric_numeral_systems).
- [16] P. A. Hsieh, J.-L. Wu, "A review of the asymmetric numeral system and its applications to digital images," *Entropy*, vol. 24, no. 3, 375, March, 2022.
- [17] A. Moffat and M. Petri, "ANS-based index compression," *Proc. of 2017 ACM on Conf. on Information and Knowledge Management (CIKM'17)*, pp. 677-686, Nov., 2017.
- [18] A. Moffat and M. Petri, "Index compression using byte-aligned ANS coding and two-dimensional contexts," *Proc. of the 7th ACM Int. Conf. on Web Search and Data Mining (WSDN'18)* pp. 405-413, Feb., 2018.
- [19] A. Moffat and M. Petri, "Large-alphabet semi-static entropy coding via asymmetric numeral systems," *ACM Trans. on Inform. Systems*, vol. 38, Issue 4, pp. 1-33, July, 2020.
- [20] H. Fujisaki, "Invariant measure for the subshifts associated with the asymmetric binary systems," *Proc. of 2018 Int. Sym. of Inform. Theory and its Applications (ISITA2018)*, pp. 675-679, Oct., 2018.
- [21] H. Fujisaki, "On topological entropies of the subshifts associated with the stream version of asymmetric binary systems," 第 46 回情報理論とその応用シンポジウム (SITA2023), 5.1.3, pp. 338-343, Nov., 2023.
- [22] H. Yokoo, "On the stationary distribution of asymmetric binary systems," *Proc. of 2016 IEEE Int. Sym. of Information Theory (ISIT2016)*, pp.11-15, July, 2016.
- [23] 井上レオナルド, 横尾英俊, "エントロピー符号化法 ANS のための確率近似法の多元情報源での評価," 電子情報通信学会 技術報告, IT2018-93, March, 2019.
- [24] X. Qi and H. Yokoo, "A new variation of asymmetric numeral systems," 電子情報通信学会 技術報告, IT2018-94, March, 2019.
- [25] D. Dubé and H. Yokoo, "Fast construction of almost optimal symbol distributions for asymmetric numeral systems," *Proc. of 2019 IEEE Int. Sym. of Information Theory (ISIT2019)*, pp. 1682-1686, July, 2019.
- [26] H. Yokoo and D. Dubé, "Asymptotic optimality of asymmetric numeral systems," 第 42 回情報理論とその応用シンポジウム 予稿集 (SITA2019), 4.2.3, pp. 289-294, Nov., 2019.
- [27] 横尾英俊, "最適性の十分条件に基づくストリーム版 ANS の再構成," 第 46 回情報理論とその応用シンポジウム (SITA2023), 5.1.2, pp. 332-337, Nov. 2023.
- [28] J. Townsend, "A tutorial on the range variant of asymmetric numeral systems," arXiv:2001.09186v3, Oct. 7, 2020.
- [29] I. Blanes, M. Hernández-Cabronero, et al., "Redundancy and optimization of tANS entropy encoders," *IEEE Trans. on Multimedia*, vo. 23, pp.4341-4350, Nov., 2021.
- [30] T. Strutz, "Rescaling of symbol counts for adaptive rANS coding," *Proc. of 2023 31th European Signal Processing Conf. (EUSIPCO2023)*, pp. 585-589, Sep., 2023.
- [31] M. Li, Y. Liu, and N. Wang, "A novel ANS coding with low computational complexity," 2023 IEEE/CIC Int. Conf. on Comm. in China (ICCC2023), pp. 1-6, Sep., 2023.
- [32] 山本博資, 岩田賢一, "無歪みデータ圧縮のための非対称符号化復号化方式," 電子情報通信学会技術報告, IT2023-17, pp.17-22, Aug., 2023.
- [33] H. Yamamoto and K. Iwata, "An asymmetric encoding-decoding scheme for lossless data compression," *Proc. of 2024 IEEE Int. Sym. of Information Theory (ISIT2024)*, July, 2024 (to appear).
- [34] 山本博資, 岩田賢一, "非対称符号化復号化方式の漸近最適性について," 第 46 回情報理論とその応用シンポジウム (SITA2023), 5.1.1, pp. 326-331, Nov., 2023.
- [35] R. G. Gallager, "Variations on a Theme by Huffman," *IEEE Trans. on Infor. Theory*, vol. IT-24, no. 6, pp.668-674, Nov., 1978.

(xxxx 年 xx 月 xx 日受付)

**山本博資** (名誉員：フェロー)

1975 静岡大・工・電気卒。1980 東大大学院博士課程了。工博。徳島大・工・助教授，電通大・助教授，東大・工／情報理工／新領域 教授，早稲田大・基幹理工・客員教授などを経て，現在，東大・名誉教授，中央大・客員研究員（機構教授）。情報理論全般の研究に従事。平 21 年度論文賞，平 29 年度業績賞，令 2 年度功績賞受賞。IEEE 会員 (Life-Fellow)。

**岩田 賢一** (正員：シニア会員)

1993 和歌山大・教育卒，1995 北陸先端大大学院情報科学研究科博士前期課程了。徳島大助手，松山東雲女子大講師，電通大助手，山形大助教授，准教授を経て，現在，福井大学 工学系部門 教授。博士 (工学)。情報理論の研究に従事。IEEE 会員。

**Abstract** Asymmetric Numeral Systems (ANS) proposed by Jarek Duda are high-performance distortionless data compression schemes that can achieve almost the same compression performance as arithmetic codes with less arithmetic operations than arithmetic coding. The ANS is widely used in various practical systems like Facebook, Apple, Google, Dropbox, Microsoft, and Pixar, due to their high performance, but many researchers still lack much knowledge about the ANS. This paper thoroughly explains the encoding and decoding algorithms of the ANS, and theoretically analyzes the average code length achievable by the ANS.

**Key words** ANS (Asymmetric Numeral Systems), arithmetic code, distortionless data-compression code, average code length