# A Universal Variable-to-Fixed Length Source Code Based on Lawrence's Algorithm

Tjalling J. Tjalkens, *Member, IEEE*, and Frans M. J. Willems, *Member, IEEE*

***Abstract*—The Lawrence algorithm is a universal binary variable-to-fixed length source coding algorithm. Here, a modified version of this algorithm is introduced and its asymptotic performance is investigated. For $M$ (the segment set cardinality) large enough, it is shown that the rate $R_\theta$ as a function of the source parameter $\theta$ satisfies**

$$R_\theta \approx h(\theta) \cdot \left(1 + \frac{\log\log M}{2\log M}\right),$$

**for $0 < \theta < 1$. Here $h(\cdot)$ is the binary entropy function. In addition to this, it is proven that no codes exist that have a better asymptotic performance, thereby establishing the asymptotic optimality of our modified Lawrence code. The asymptotic bounds show that universal variable-to-fixed length codes can have a significantly lower redundancy than universal fixed-to-variable length codes with the same number of codewords.**

***Index Terms*—Universal source coding, enumerative coding, variable-to-fixed length codes, asymptotic redundancy.**

## I. Preliminaries

A BINARY memoryless information source generates a sequence of independent and identically distributed random variables $\{X_t\}_{t=1,\infty}$, each of which assumes values in the finite set $\mathscr{X} \triangleq \{0, 1\}$, called the source alphabet. Let $\theta \triangleq \Pr\{X_t = 1\} = 1 - \Pr\{X_t = 0\}$, $t = 1, 2, \cdots$. Then the entropy of the source (in bits per symbol) is equal to $h(\theta) \triangleq -\theta\log(\theta) - (1-\theta)\log(1-\theta)$. (We assume throughout this paper that $\log(\cdot)$'s have base 2 and that $\ln(\cdot)$ has base $e$.)

In what follows, we will describe a universal variable-to-fixed length coding strategy for the class of binary memoryless sources. With these codes, the (infinite length) source sequence is chopped up into sequences of variable length (segments), chosen from some finite set $\mathscr{S}$ of segments, and each segment is assigned to a code sequence of fixed length $N = \log M$, where $M$ is the number of segments in $\mathscr{S}$. (Note that we ignore the rounding of $\log M$ to an integer). This set of segments must be complete, i.e., every infinite sequence has a prefix in the segment set, since every sequence must be subdividable into segments. We also require that the set is proper, i.e., no segment in the set is a prefix of an other segment in the set. In this way, we guarantee a unique subdivision of every source sequence. It is assumed that the code alphabet $\mathscr{Y} = \{0, 1\}$. Let $L(x^*) = k$ be the length of segment $x^* = (x_1, x_2, \cdots, x_k)$. Then instead of sending the $L(x^*)$ source symbols to a receiver we can send the corresponding codeword. This codeword can be used by the receiver to reconstruct the source segment. If the code is properly chosen, the average segment length $L_\theta^{av}$ can be considerably larger than $N$, where

$$L_\theta^{av} \triangleq \sum_{x^* \in \mathscr{S}} \Pr\{X^* = x^*\} L(x^*). \tag{1}$$

Therefore, the (compression) rate $R_\theta$ of a code, which is defined as $R_\theta \triangleq N/L_\theta^{av}$, can be smaller than one. Note that a universal code can not be designed using the statistics of the source.

Tunstall [10] discovered a procedure for constructing an optimum segment set for a given memoryless source. For a fixed $N$, this construction maximizes $L_\theta^{av}$. If we form such a code for a binary source with $\theta < 0.5$, then $N + \log(\theta) \leq L_\theta^{av} h(\theta) \leq N$. A major disadvantage of a Tunstall code is that the complete code has to be stored by both the encoder and the decoder. Note that these codes are not universal.

Lawrence [5] devised a variable-to-fixed length code that is easier to implement. Only a part of Pascal's triangle must be stored by the encoder and the decoder now. An additional feature of this code is that it is universal. This code can be seen as the variable-to-fixed length counterpart of Schalkwijk's [8] "Pascal triangle" algorithm.

In this paper, we will describe a modification of this Lawrence code. Instead of using a prefix and a suffix implementation as in Schalkwijk [8] and in Lawrence [5], we compute the lexicographical indices of the segments. The lexicographical index of a segment $x^* \in \mathscr{S}$ equals the number of segments in $\mathscr{S}$ that are less than $x^*$ in a lexicographical ordering. This index can be represented using $\log M$ binary symbols. We also change the segment set of the code. Both modifications yield a more natural and simple implementation of the algorithm and reduce the redundancy of the code. In the following sections, we will be more specific about our "modified Lawrence" code.

## II. The Segment Set

As in Tunstall [10], we try to define our segment set as a set of, more or less, equiprobable sequences. However, the probability $\Pr\{x^*\} = (1-\theta)^a\theta^b$ of a sequence containing $a$ zeros and $b$ ones is unknown to the encoder and decoder.

Manuscript received March 16, 1988; revised June 17, 1991. This work was presented in part at the IEEE International Symposium on Information Theory, Kobe, Japan, June 19–24, 1988; and in part at the IEEE International Symposium on Information Theory, San Diego, CA, January 14–19, 1990.

The authors are with the Faculty of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands.

IEEE Log Number 9103967.

0018-9448/92$03.00 

Because we have no knowledge about the parameter $\theta$, we assume that $\theta$ is a random variable uniformly distributed over the interval [0, 1]. In this way, we do not favor one particular value above another and this seems to be a fair choice. So the source is a composite source, see Davisson [3], instead of a memoryless one. For this composite source, the probability $Q(x^*)$ of a sequence containing $a$ zeros and $b$ ones is

$$Q(x^*) \triangleq \int_0^1 (1-\theta)^a \theta^b \, d\theta = \frac{1}{a+b+1} \binom{a+b}{b}^{-1}. \tag{2}$$

We use $Q(x^*)$ to define the segments. Given a positive integer $C$, $x^* \triangleq (x_1, x_2, \cdots, x_{L(x^*)})$ is a segment in the code, if and only if

$$Q(x^*)^{-1} \geq C \quad \text{and} \quad Q(x^{*-1})^{-1} < C, \tag{3}$$

with $x^{*-1} \triangleq (x_1, x_2, \cdots, x_{L(x^*)-1})$. Note that we can use Pascal's triangle $P(a, b) \triangleq \binom{a+b}{b}$, see Fig. 1, to determine whether or not a sequence is a segment. A new segment starts at the top of the triangle, $x_t = 0$ corresponds to a step in the $a$-direction, $x_t = 1$ to a step in the $b$-direction. Hence, 0010001 is a segment since 7*6 < 82 and 8*21 ≥ 82. We say that (5, 2) is on the segment set boundary in the triangle, i.e., (5, 2) is a boundary point. See Appendix A-1 for the exact definition of this term. In Fig. 1, we also indicate the path for 0010001.

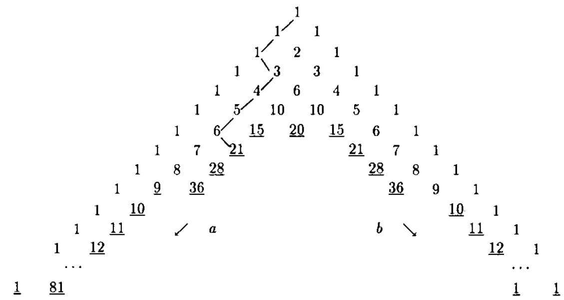


Fig. 1. Modified Lawrence code with $C = 82$. Boundary points are underlined.

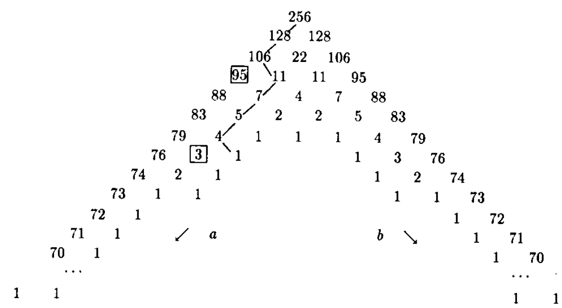


Fig. 2. Coding array for $C = 82$.

## III. The Coding Algorithm

We have seen that the segment set boundary can be determined using Pascal's triangle, but is it also possible to find the lexicographical index of a segment in a similar way? Yes, but we have to refill the triangle first. An element $M(a,b)$ of the new array must be equal to the number of distinct ways to reach a boundary point after we have seen $a$ zeros and $b$ ones. Therefore,

$$M(a, b) \triangleq 1, \qquad \text{if } (a, b) \text{ is a boundary point},$$
$$M(a, b) \triangleq M(a+1, b) + M(a, b+1), \qquad \text{if not.} \tag{4}$$

Observe that $M(0, 0)$ is equal to the total number of segments in the set. $M(0, 0)$ is a function of $C$ and for $C = 82$ this total number is 256, see Fig. 2, where we refilled the triangle of Fig. 1 from bottom to top using (4) and starting at the boundary points. In Appendix A, we show that for $C$ large enough $2C \leq M(0, 0) \leq 2C\left(1 + \ln \frac{\log M(0,0)}{2}\right)$.

To determine the lexicographical index (and the end) of a segment, the encoder uses this $M(a, b)$ array in the following way:

```
1) index := 0 a := 0 b := 0
2) WHILE M(a, b) ≠ 1 DO
      IF x(next) = 0
      THEN a := a + 1
      ELSE index := index + M(a + 1, b)
           b := b + 1.
```

This lexicographical index is sent to the decoder that reconstructs the segment as follows:

```
1) I := 0 a := 0 b := 0
2) WHILE M(a, b) ≠ 1 DO
      IF index < (I + M(a + 1, b))
      THEN x(next) := 0
           a := a + 1
      ELSE x(next) := 1
           I := I + M(a + 1, b)
           b := b + 1.
```

It will be clear that the lexicographical index of the segment 0010001 is 95 + 3 = 98, see Fig. 2. We remark that there are $M(3, 0) = 95$ segments starting with "000" and $M(6, 1) = 3$ segments starting with "0010000."

## IV. The Performance

The redundancy of a code is defined as the difference between the compression rate $R_\theta$ and the source entropy $h(\theta)$. In this section, we tabulate the redundancy of our algorithm and compare it with the universal "Pascal triangle" algorithm [8] and Lawrence's algorithm [5]. We compute the redundancy of the three codes for the code sizes 256 (i.e., 8 digits codewords) and 65536 (i.e., 16 digits codewords). The results are listed in Table I.

From this table we can see that the two variable-to-fixed length algorithms (Lawrence and the modified Lawrence algorithms) outperform the fixed-to-variable length Pascal

TABLE I
THE REDUNDANCIES

| $\theta$ | Pascal Triangle | Lawrence | Modified Algorithm |
|---|---|---|---|
| Code size is $2^8$ | | | |
| 0.5 | 0.17857 | 0.25799 | 0.24574 |
| 0.1 | 0.22958 | 0.21492 | 0.17505 |
| 0.01 | 0.37748 | 0.20726 | 0.06196 |
| 0.001 | 0.42016 | 0.24226 | 0.09132 |
| 0.0001 | 0.42740 | 0.24889 | 0.09768 |
| 0.00001 | 0.42842 | 0.24986 | 0.09862 |
| Code size is $2^{16}$ | | | |
| 0.5 | 0.09943 | 0.12219 | 0.19436 |
| 0.1 | 0.13257 | 0.11983 | 0.11929 |
| 0.01 | 0.22517 | 0.04883 | 0.03849 |
| 0.001 | 0.25925 | 0.00480 | 0.00449 |
| 0.0001 | 0.26559 | 0.00329 | 0.00063 |
| 0.00001 | 0.26653 | 0.00381 | 0.00102 |

triangle algorithm, except for high entropy sources. Also we observe that for low entropy sources the modified algorithm performs better than the original Lawrence algorithm. From the table we can conclude that the variable-to-fixed length algorithms perform well for practical code sizes.

## V. The Asymptotic Performance

In the previous section, we have seen that the modified algorithm compares favorably to the other universal algorithms for small code sizes. It is also interesting to see how the rate of this modified Lawrence algorithm converges as the code size increases. An asymptotic upperbound on the rate is stated in the following theorem and its proof is given in Appendix A.

*Theorem 1:* For any $\delta > 0$ and any $0 < \theta < 1$, we have for $C > C(\theta, \delta)$ that

$$R_\theta = \frac{\log M}{L_\theta^{av}} \leq \left(1 + (1+\delta) \cdot \frac{\log\log M}{2\log M}\right) \cdot h(\theta). \quad (5)$$

It should be noted that $M$ increases when $C$ increases. In particular, see the text before (11) in Appendix A where it is shown that $M \geq 2C$.

## VI. A Lowerbound to the Compression Rate

In this section, we state a lowerbound to the compression rate achieved by any variable-to-fixed length code with $M$ codewords for (almost) all sources $\theta \in [0, 1]$. The result is summarized in the following theorem, and the proof thereof is given in Appendix B.

*Theorem 2:* For all $\delta > 0$ and any variable-to-fixed length code with a large enough number $M$ of segments we have

$$R_\theta \geq \left(1 + (1-\delta) \cdot \frac{\log\log M}{2\log M}\right) \cdot h(\theta),$$

for all $0 \leq \theta \leq 1$ except for those $\theta$ in a set $\mathscr{B}_\delta$ whose volume tends to zero as $M$ increases.

The proof of this theorem is based on Rissanen's converse for fixed-to-variable length codes for arbitrary sources [7]. We restrict ourselves to variable-to-fixed length codes for binary memoryless sources, although it is clear that the proof readily extends to arbitrary finite alphabet memoryless sources.

## VII. Conclusion

In this contribution, we showed that the modified Lawrence algorithm is universal over the class of binary memoryless sources, and in addition, that the rate converges asymptotically optimally fast to the source entropy $h(\theta)$. For the class of binary memoryless sources, the asymptotically optimal redundancy is $h(\theta) \cdot \log\log M/(2\log M)$ where $M$ is the number of codewords. When we compare this to Rissanen's redundancy for this case, which is $\log N/(2N) = \log\log M/(2\log M)$ where $M$ again denotes the number of codewords, we see that in the VF case the asymptotic redundancy is a factor $h(\theta)$ lower than in the FV case.

An earlier converse for VF codes for memoryless sources was given by Trofimov, see [4]. The lowerbound stated there showed a uniform convergence in correspondence with Davisson's [3] result that the class of memoryless sources is "minimax universal." However this bound is expressed in terms of the minimal average message length of a code (with respect to the class of sources) and we consider this a less realistic approach than our bound of Theorem 2 that relates the redundancy to the code size.

In a recent paper, Shtarkov [9] presented two universal VF coding schemes for $m$-ary memoryless sources. The first scheme achieves Trofimov's lowerbound. The upperbound for the redundancy for the second scheme for binary sources is twice as high as our upperbound (5).

## Acknowledgment

The authors thank the reviewer who suggested that an upperbound on the rate in a previous version might be too weak. Motivated by this remark, we indeed obtained the suggested improvement and also the converse result.

## Appendix A

## An Upper bound on the Rate of the Modified Lawrence Algorithm

This appendix consists of four subsections. Throughout these subsections, we assume that $0 < \theta < 1$ and that $\delta > 0$.

### 1. The Shape of the Boundary

For $(a, b)$ with $a = 0, 1, \cdots$ and $b = 0, 1, \cdots$, we define the segment function $f(\cdot,\cdot)$ as

$$f(a, b) \triangleq (a + b + 1)\binom{a+b}{b}. \quad (6)$$

Recall that $(a, b)$ corresponds to a sequence containing $a$ zeros and $b$ ones. A point $(a, b)$ is now said to be *internal*, if and only if $f(a, b) < C$. A point $(a, b)$ is a *boundary point*, if and only if $f(a, b) \geq C$ and at least one of the points $(a, b-1)$ and $(a-1, b)$ is internal.

Note that if $(a, b)$ is a boundary (internal) point, $(b, a)$ is a boundary (internal) point too. The set of boundary points (segments) is, therefore, symmetric. Let $S$ be the minimal value of $a + b$ when $(a, b)$ is a boundary point. Then, if $S$ is even, $(S/2, S/2)$ must be a boundary point and $(S/2, S/2 - 1)$ an internal point. Consequently $C > f(S/2, S/2 - 1) \geq S \cdot \frac{2^{S-1}}{S} = 2^{S-1}$. Likewise, for $S$ odd, we can show that $C > 2^{S-1}$. Hence,

$$S < \log 2C. \tag{7}$$

To avoid degenerate codes, we always assume that $S \geq 1$ and thus $C > 1$.

Now consider a boundary point $(a, b)$ with $a \geq b$. Then, if $b \geq 1$, the point $(a, b - 1)$ must be internal since $f(a, b - 1) \leq f(a - 1, b)$. Note that not both $(a + 1, b - 1)$ and $(a + 1, b)$ can be boundary points. On the other hand either $(a + 1, b - 1)$ or $(a + 1, b)$ must be a boundary point since $f(a + 1, b) \geq f(a, b) \geq C$.

Now, for $1 \leq b \leq S/2$, let $a_{\min}(b)$ resp. $a_{\max}(b)$ be the minimal resp. maximal value of $a$ such that $(a, b)$ is a boundary point. The consequence of this is that $(a_{\min}(b), b)$, $(a_{\min}(b) + 1, b), \cdots, (a_{\max}(b), b)$ are all (adjacent) boundary points and so is $(a_{\max}(b) + 1, b - 1)$.

If we consider a boundary point $(a, 0)$ then $(a - 1, 0)$ must be an internal point. Note that $(a + 1, 0)$ can not be a boundary point, too.

### 2. An Upper Bound for the Number of Segments

For $0 \leq b \leq S/2$, let $M(b)$ be the number of segments that contain $b$ ones. From the previous subsection it follows that

$$\begin{aligned} M(b) &= \binom{a_{\min}(b) + b}{b} \\ &\quad + \binom{a_{\min}(b) + b}{b - 1} + \cdots + \binom{a_{\max}(b) + b - 1}{b - 1} \\ &= \binom{a_{\max}(b) + b}{b}. \end{aligned} \tag{8}$$

Using symmetry we find for the total number of segments $M$ that

$$M = 2 \cdot \sum_{b=0, \lfloor S/2 \rfloor} \binom{a_{\max} + b}{b} - N(S), \tag{9}$$

where $N(S)$ is 0 for odd $S$ and $\binom{S}{S/2}$ for even $S$. Note that $(a_{\max}(b), b - 1)$ is an interior point if $2 \leq b \leq \lfloor S/2 \rfloor$. Hence,

$$\begin{aligned} C &> f(a_{\max}(b), b - 1) \\ &= (a_{\max}(b) + b)\binom{a_{\max}(b) + b - 1}{b - 1} \\ &= b\binom{a_{\max}(b) + b}{b}. \end{aligned} \tag{10}$$

From $\binom{a_{\max}(0)}{0} + \binom{a_{\max}(1)}{1} = C$ (and consequently $M \geq 2C$) for $C$ large enough, and (10), we obtain from (9) that

$$\begin{aligned} M &\leq 2C + 2 \cdot \sum_{b=2, \lfloor S/2 \rfloor} \frac{C}{b} \leq 2C + 2 \int_1^{S/2} \frac{C}{b}\, db \\ &= 2C\left(1 + \ln \frac{S}{2}\right) \\ &\leq 2C\left(1 + \ln\left(\frac{\log 2C}{2}\right)\right) \\ &\leq 2C\left(1 + \ln\left(\frac{\log M}{2}\right)\right), \end{aligned} \tag{11}$$

for $C$ large enough. The third inequality is a consequence of (7).

### 3. An Upper Bound for the Segment Divergence

Recall that the probability that the source generates a sequence $x^*$ containing $a$ zeros and $b$ ones is $\Pr\{X^* = x^*\} = (1 - \theta)^a \theta^b$. In this subsection we will derive an upper bound for the divergence $D^*(P \| Q)$ between the actual probability distribution over the segments $P(x^*) \triangleq \Pr\{X^* = x^*\}$ and the design distribution $Q(x^*)$. For this divergence we can write

$$\begin{aligned} D^*(P \| Q) &\triangleq \sum_{x^* \in \mathcal{S}} P(x^*) \log \frac{P(x^*)}{Q(x^*)} \\ &= \sum_{(a,b) \in \mathcal{G}} \Pr\{(A, B) = (a, b)\} \\ &\quad \cdot \log \frac{(1 - \theta)^a \theta^b}{\frac{1}{a + b + 1}\binom{a + b}{b}^{-1}}, \end{aligned} \tag{12}$$

where $\mathcal{S}$ is the set of all segments in the code, $\mathcal{G}$ the set of all boundary points $(a, b)$, and $\Pr\{(A, B) = (a, b)\}$ is the probability that the source generates a segment with $a$ zeros and $b$ ones.

Now let $\theta^- \triangleq \theta/2$ and $\theta^+ \triangleq (1 + \theta)/2$. If we note that $0 < \theta^- < \theta < \theta^+ < 1$, we can define

$$\mathcal{G}^\theta \triangleq \left\{(a, b) \in \mathcal{G} : \theta^- \leq \frac{b}{a + b} \leq \theta^+\right\}. \tag{13}$$

Note that both $a$ and $b$ tend to infinity for $(a, b) \in \mathcal{G}^\theta$ when $C$ increases. This follows from the fact that for any boundary point $(a, b)$,

$$\begin{aligned} a + b &= \frac{1}{2} \log 2^{2(a+b)} \geq \frac{1}{2} \log (a + b + 1) 2^{a+b} \\ &\geq \frac{1}{2} \log (a + b + 1)\binom{a + b}{b} \geq \log \sqrt{C} \end{aligned} \tag{14}$$

and from $a \geq (1 - \theta^+)(a + b)$ and $b \geq \theta^-(a + b)$ for $(a, b) \in \mathcal{G}^\theta$. Therefore, for $C$ large enough both $a$ and $b$ will be 1 or larger for $(a, b) \in \mathcal{G}^\theta$. Using Stirling's approximation $t! = \sqrt{2\pi} \cdot t^{t+\frac{1}{2}} \exp\left(-t + \frac{\alpha}{12t}\right)$ for some $0 < \alpha < 1$ for $t > 0$ (see Abramowitz and Stegun [1]), we obtain for the argument of the log in (12) that

$$\frac{(1-\theta)^a\theta^b}{\frac{1}{a+b+1}\binom{a+b}{b}^{-1}} \le (a+b+1)\sqrt{\frac{a+b}{2\pi ab}}\exp\left(\frac{1}{12}\right)\cdot\left(\frac{(a+b)(1-\theta)}{a}\right)^a\left(\frac{(a+b)\theta}{b}\right)^b$$
$$= \sqrt{a+b}\sqrt{\frac{(a+b+1)^2}{2\pi ab}}\exp\left(\frac{1}{12}\right)\cdot\exp\left(-(a+b)d\left(\frac{b}{a+b}\|\theta\right)\right)$$
$$\le \sqrt{a+b}\sqrt{\frac{(a+b+1)^2}{2\pi ab}}\exp\left(\frac{1}{12}\right)$$
$$\le \sqrt{a+b}\sqrt{\frac{9\exp(1/6)}{8\pi\theta^-(1-\theta^+)}}$$
$$= \sqrt{a+b}\sqrt{\frac{9\exp(1/6)}{2\pi\theta(1-\theta)}} = K_\theta\sqrt{a+b}, \quad (15)$$

for $K_\theta \triangleq \sqrt{(9\exp(1/6))/(2\pi\theta(1-\theta))}$ and where $d(p\|q) \triangleq p\log\frac{p}{q} + (1-p)\log\frac{1-p}{1-q}$ (with $0 \le p \le 1$ and $0 < q < 1$) is the (nonnegative) binary divergence function. For any $(a, b) \in \mathscr{G}$, we find

$$\frac{(1-\theta)^a\theta^b}{\frac{1}{a+b+1}\binom{a+b}{b}^{-1}} \le (a+b+1) \le C. \quad (16)$$

The last inequality holds since for any point $(a, b)$, $a + b \ge 1$, on the boundary, there exists an interior point $(a', b')$ with $a' + b' = a + b - 1$ for which $(a' + b' + 1)\cdot B < C$, for some binomial coefficient $B$.

To get an upper bound for the divergence $D^*(P\|Q)$ in (12), we first consider

$$\Pr\{(A, B) \notin \mathscr{G}^\theta\} = \Pr\left\{(A, B) \in \mathscr{G}: \frac{B}{A+B} < \theta^-\right\} + \Pr\left\{(A, B) \in \mathscr{G}: \frac{B}{A+B} > \theta^+\right\}$$
$$= \Pr\left\{\sum_{i=0,\lceil l^-\theta^-\rceil-1}(1-\theta)^{l^- - i}\theta^i\right\} + \Pr\left\{\sum_{i=\lceil l^+\theta^+\rceil, l^+}(1-\theta)^{l^+ - i}\theta^i\right\}$$
$$\le \exp(-l^- d(\theta\|\theta^-)) + \exp(-l^+ d(\theta\|\theta^+))$$
$$\le 2\exp(-\log(\sqrt{C})\cdot\Delta) = 2C^{-\Delta/2}, \quad (17)$$

where $l^-$ resp. $l^+$ is the value of $a + b$ of the boundary point $(a, b) \in \mathscr{G}^\theta$ for which $b/(a+b)$ is minimal resp. maximal and $\Delta = \min(d(\theta\|\theta^-), d(\theta\|\theta^+))$. The second equality follows from inspection of the shape of the boundary, the first inequality from Chernoff's bound, the last one from inequality (14).

We now combine (15)–(17) and obtain for the divergence in (12)

$$D^*(P\|Q) \le \sum_{(a,b)\in\mathscr{G}^\theta}\Pr\{(A, B) = (a, b)\}\log(K_\theta\sqrt{a+b}) + \Pr\{(A, B) \notin \mathscr{G}^\theta\}\log C$$
$$\le \sum_{(a,b)\in\mathscr{G}}\Pr\{(A, B) = (a, b)\}\log(\sqrt{a+b}) + K_\theta + 2C^{-\Delta/2}\log C$$
$$\le \frac{1}{2}\log(L_\theta^{av}) + K_\theta + \delta, \quad (18)$$

for $C$ large enough.

### 4. An Upper Bound on the Rate

First recall (from Section A-2) that $M \ge 2C$ for $C$ large enough. Therefore, $M$ tends to infinity when $C$ increases. We shall use this fact several times in this subsection.

Taking logarithms in (11) and noting that $1 + \ln(\log(M)/2) \le \log\log M$ for $C$ large enough we obtain

$$\log C \ge \log M - \log\log\log M - 1. \quad (19)$$

From Massey's "leaf-node" theorem [6], it follows that $H_\theta^{\text{segm}} = L_\theta^{av}\cdot h(\theta)$, where $H_\theta^{\text{segm}}$ is the segment entropy. Hence, with $\log Q(x^*) \le C^{-1}$ for all segments $x^*$, we have

$$D^*(P\|Q) = \sum_{x^*\in\mathscr{S}} P(x^*)\log\frac{1}{Q(x^*)} - H_\theta^{\text{segm}} \ge \log C - L_\theta^{av}\cdot h(\theta). \quad (20)$$

Substituting the upper bound for the divergence (18) in (20), we find that

$$\log C \le L_\theta^{av}\cdot h(\theta) + \frac{1}{2}\log L_\theta^{av} + K_\theta + \delta, \quad (21)$$

for $C$ large enough. Combining (19) and (21) yields, again for $C$ large enough, that

$$L_\theta^{av}\cdot h(\theta) \ge \log M - \frac{1}{2}\log L_\theta^{av} - \log\log\log M - K_\theta - \delta - 1$$
$$\ge \log M - \frac{1}{2}\log\log M - \log\log\log M - K'_\theta, \quad (22)$$

where $K'_\theta \triangleq K_\theta - \frac{1}{2}\log h(\theta) + 1 + \delta$. The second inequality follows from $\log M \ge H_\theta^{\text{segm}} = L_\theta^{av}\cdot h(\theta)$. For the rate of our modified Lawrence, code we finally obtain

$$R_\theta = \frac{\log M}{L_\theta^{av}} \le \frac{\log M}{\log M - \frac{1}{2}\log\log M - \log\log\log M - K'_\theta}\cdot h(\theta)$$
$$= \frac{1}{1 - \frac{\log\log M}{2\log M} - \frac{\log\log\log M}{\log M} - \frac{K'_\theta}{\log M}}\cdot h(\theta)$$
$$\le \left(1 + (1+\delta)\cdot\frac{\log\log M}{2\log M}\right)\cdot h(\theta), \quad (23)$$

for $C$ large enough. This proves Theorem 1.

## Appendix B

## The Proof of the Converse

The converse presented here, can be regarded as an adaptation of Rissanen's converse for fixed-to-variable length codes for arbitrary sources [7]. We restrict ourselves to variable-to-fixed length codes for binary memoryless sources. For the source parameter $\theta$ we assume that $0 \le \theta \le 1$.

Let $\epsilon > 0$. Fix a $\gamma$ such that $0 < \gamma < 1$. Consider a variable-to-fixed length code with a (proper and complete) segment set $\mathscr{S}$ with $M$ segments. For such a code, we define the probability

$$A_\theta \triangleq \Pr\{x^* \in \mathscr{S}: L(x^*) < L^{\min}\}, \tag{24}$$

where $L^{\min} \triangleq \lceil \gamma \log M \rceil$. Note that $A_\theta$ depends on $\theta$ as is indicated by its subscript. For the entropy of the segments in terms of $A_\theta$, we find that

$$\begin{aligned} H_\theta^{\text{segm}} &\le h(A_\theta) + A_\theta \log 2^{L^{\min}-1} + (1 - A_\theta)\log M \\ &\le 1 + (1 - A_\theta(1-\gamma))\log M. \end{aligned} \tag{25}$$

From Massey's "leaf-node" theorem [6] it follows that $H_\theta^{\text{segm}} = L_\theta^{av} \cdot h(\theta)$. Combining this with our the upper bound for $H_\theta^{\text{segm}}$, we find for the rate of our code for given $\theta$ that

$$\begin{aligned} R_\theta &= \frac{\log M}{L_\theta^{av}} = \frac{\log M}{H_\theta^{\text{segm}}} \cdot h(\theta) \\ &\ge \frac{\log M}{1 + (1 - A_\theta(1-\gamma))\log M} \cdot h(\theta) \\ &\ge \left(1 + A_\theta(1-\gamma) - \frac{1}{\log M}\right) \cdot h(\theta). \end{aligned} \tag{26}$$

Note that this lower bound for $R_\theta$ holds also for $h(\theta) = 0$. Now we easily arrive at our first implication:

$$\begin{aligned} &A_\theta(1-\gamma) \ge \frac{\log\log M}{2\log M} \Rightarrow \\ &\qquad R_\theta \ge \left(1 + \frac{\log\log M}{2\log M} - \frac{1}{\log M}\right) \cdot h(\theta). \end{aligned} \tag{27}$$

Next we introduce the set $\mathscr{X}_\theta$ of segments that have a prefix of length $L^{\min}$ which is "$\theta$-typical," i.e.,

$$\mathscr{X}_\theta \triangleq \left\{ x^* \in \mathscr{S}: L(x^*) \ge L^{\min} \cdot \left| \frac{1}{L^{\min}} \cdot \sum_{i=1}^{L^{\min}} x_i - \theta \right| \le \frac{c}{\sqrt{L^{\min}}} \right\}, \tag{28}$$

where $c > 0$ is to be specified later. Note that for $P_\theta \triangleq \Pr\{x^* \in \mathscr{X}_\theta\}$, by the union-bound and Chebyshev's inequality, we may conclude that

$$\begin{aligned} P_\theta &\ge 1 - \Pr\{x^* \in \mathscr{S}: L(x^*) < L^{\min}\} \\ &\quad - \Pr\left\{ x^* \in \mathscr{S}: \left| \frac{1}{L^{\min}} \cdot \sum_{i=1}^{L^{\min}} x_i - \theta \right| > \frac{c}{\sqrt{L^{\min}}} \right\} \\ &\ge 1 - A_\theta - \frac{\theta(1-\theta)}{c^2} \ge 1 - A_\theta - \frac{1}{4c^2}. \end{aligned} \tag{29}$$

Let $M_\theta$ be the number of segments in $\mathscr{X}_\theta$, then from the "log-sum" inequality (Csiszár and Körner [2]), we obtain that

$$T_\theta \triangleq \sum_{x^* \in \mathscr{X}_\theta} \Pr\{x^*\} \log \frac{\Pr\{x^*\}}{1/M} \ge P_\theta \log \frac{P_\Theta}{M_\theta / M}. \tag{30}$$

Furthermore, from Massey's leaf-node theorem, the log-sum inequality and the basic inequality $\ln t < t - 1$ it follows that

$$\begin{aligned} \log M &= L_\theta^{av} \cdot h(\theta) + \sum_{x^* \in \mathscr{S}} \Pr\{x^*\} \log \frac{\Pr\{x^*\}}{1/M} \\ &\ge L_\theta^{av} \cdot h(\theta) + T_\theta + (1 - P_\theta) \log \frac{1 - P_\theta}{1 - M_\theta/M} \\ &\ge L_\theta^{av} \cdot h(\theta) + T_\theta - \log e. \end{aligned} \tag{31}$$

This combined with $L_\theta^{av} \cdot h(\theta) = H_\theta^{\text{segm}} \le \log M$, leads for $M \ge 2^{\exp \frac{2}{1-\epsilon}}$, to our second implication:

$$\begin{aligned} &T_\theta \ge (1-\epsilon) \cdot \frac{\log\log M}{2} \Rightarrow \\ &\qquad R_\theta \ge \left(1 + (1-\epsilon) \cdot \frac{\log\log M}{2\log M} - \frac{\log e}{\log M}\right) \cdot h(\theta). \end{aligned} \tag{32}$$

From the definition,

$$\mathscr{B}_\epsilon \triangleq \left\{ \theta: A_\theta(1-\gamma) < \frac{\log\log M}{2\log M} \wedge T_\theta < (1-\epsilon) \cdot \frac{\log\log M}{2} \right\}. \tag{33}$$

and the implications (27) and (32) it follows that for $m$ large enough

$$\theta \notin \mathscr{B}_\epsilon \Rightarrow R_\theta \ge \left(1 + (1 - 2\epsilon) \cdot \frac{\log\log M}{2\log M}\right) \cdot h(\theta). \tag{34}$$

For $\theta \in \mathscr{B}_\epsilon$ and $M$ large enough, however, the rate of our code may not satisfy the inequality in (34). In this converse it is our objective to show that the "volume" $V$ of the set $\mathscr{B}_\epsilon$, i.e., the set containing all $\theta$'s for which our code has a "small enough" redundancy, can be made arbitrarily small by increasing $M$. Therefore, let $N$ be the maximal number of disjoint intervals $I_\theta \triangleq [\theta - c/\sqrt{L^{\min}}, \theta + c/\sqrt{L^{\min}}]$ that can be constructed with centerpoints $\theta \in \mathscr{B}_\epsilon$. Let $\mathscr{C}$ be the set of all centerpoints. The corresponding intervals may not cover $\mathscr{B}_\epsilon$ completely, but they do if we double their sizes. To see this, suppose that some $\hat{\theta} \in \mathscr{B}_\epsilon$ remained uncovered after the size of the intervals had been doubled. Then, the distance between $\hat{\theta}$ and any centerpoint would be at least $2c/\sqrt{L^{\min}}$. But then the original number of intervals would not have been maximal! We can, therefore, bound the volume $V$ by

$$V \le 4N \cdot \frac{c}{\sqrt{L^{\min}}}. \tag{35}$$

To find an upper bound for $N$, consider a $\theta \in \mathscr{B}_\epsilon$. Then from inequality (30) and the definition of $\mathscr{B}_\epsilon$ in (33), we may conclude that

$$-\log \frac{M_\theta}{M} < \left( \frac{1-\epsilon}{P_\theta} - \frac{2\log P_\theta}{\log\log M} \right) \cdot \frac{\log\log M}{2}, \tag{36}$$

while from (29) and definition (33), it follows that

$$P_\theta > 1 - \frac{1}{1-\gamma} \cdot \frac{\log\log M}{2\log M} - \frac{1}{4c^2}. \tag{37}$$

By choosing $c = (\sqrt{\epsilon})^{-1}$ we can guarantee that $P_\theta > 1 - \epsilon/2$ for $M$ large enough. Substituting this in (36), we obtain that for $M$ large enough there must exist an $\alpha < 1$ such that

$$M_\theta > M \cdot (\log M)^{-\alpha/2}. \tag{38}$$

The intervals $I_\theta$ that correspond to all centerpoints in $\mathscr{C}$ are disjoint, therefore the sets $\mathscr{X}_\theta$ corresponding to all these centerpoints must also be disjoint. Consequently

$$M \geq \sum_{\theta \in \mathscr{C}} M_\theta > NM \cdot (\log M)^{-\alpha/2}. \tag{39}$$

Combining (38) and (39) yields

$$N < (\log M)^{\alpha/2}. \tag{40}$$

If we substitute this bound for $N$ in (35), set $c = (\sqrt{\epsilon})^{-1}$ and note that $L^{\min} \geq \gamma \log M$, we obtain that

$$V < \frac{4}{\sqrt{\gamma\epsilon}} \cdot (\log M)^{(\alpha-1)/2}. \tag{41}$$

Since $\alpha < 1$, we get that $V \downarrow 0$ for $M \to \infty$. Taking $\epsilon = \delta/2$ finally proves the converse as stated in Theorem 2. □

## References


[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*. New York: Dover, 1970.

[2] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Budapest, Hungary: Akadémiai Kiadó, 1981.

[3] L. D. Davisson, "Universal noiseless coding," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 783–795, Nov. 1973.

[4] R. E. Krichevsky and V. K. Trofimov, "The performance of universal encoding," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 199–207, Mar. 1981.

[5] J. C. Lawrence, "A new universal coding scheme for the binary memoryless source," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 466–472, July 1977.

[6] J. L. Massey, "The entropy of a rooted tree with probabilities," presented at the *IEEE Int. Symp. Inform. Theory*, St. Jovite, Canada, Sept. 26–30, 1983.

[7] J. Rissanen, "Universal coding, information, prediction, and estimation," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 629–636, July 1984.

[8] J. P. M. Schalkwijk, "An algorithm for source coding," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 395–399, May 1972.

[9] Yu. M. Shtarkov, "The variable-to-block universal encoding of memoryless sources," presented at the *Fourth Joint Swedish-Soviet Int. Workshop Inform. Theory*, Aug 1989, Gotland, Sweden.

[10] B. P. Tunstall, "Synthesis of noiseless compression codes," Ph.D. dissert., Georgia Inst. of Technol., Atlanta, GA, Sept. 1967.