

# On Adaptive Strategies for an Extended Family of Golomb-type Codes

Gadiel Seroussi  
Marcelo J. Weinberger  
Computer Systems Laboratory

## Abstract

Off-centered, two-sided geometric distributions of the integers are often encountered in lossless image compression applications, as probabilistic models for prediction residuals. Based on a recent characterization of the family of optimal prefix codes for these distributions, which is an extension of the Golomb codes, we investigate adaptive strategies for their symbol-by-symbol prefix coding, as opposed to arithmetic coding. Our adaptive strategies allow for coding of prediction residuals at very low complexity. They provide a theoretical framework for the heuristic approximations frequently used when modifying the Golomb code, originally designed for one-sided geometric distributions of non-negative integers, so as to apply to the encoding of any integer.

**Index Terms:** image compression, adaptive coding, Golomb codes, geometric distribution, low complexity

# 1 Introduction

Predictive coding techniques [1] have become very widespread in lossless compression of continuous-tone images, due to their usefulness in capturing expected relations (e.g., smoothness) between adjacent pixels. It has been observed [2] that a good probabilistic model for prediction errors is given by a *two-sided geometric distribution* (TSGD) centered at zero. Namely, the probability of an integer error value  $x$  is proportional to  $\theta^{|x|}$ , where  $\theta \in (0, 1)$  is a scalar parameter that controls the two-sided exponential decay rate. We assume in the sequel that prediction errors can take on any integer value, an assumption that, in the context of exponential distributions, is well approximated in practice by the use of large symbol alphabets (e.g., 8 bits per pixel).

In applications where very low complexity is a primary concern, arithmetic coding of the TSGD may not be an affordable option. For that reason, symbol-by-symbol prefix coding of these distributions has been the subject of extensive research in conjunction with low complexity lossless image compression algorithms. This research reveals that well-tuned schemes based on symbol-wise coding can attain, for typical gray-scale images, compression ratios quite close to that of their counterparts based on arithmetic coding [3, 4].

Since the problem of completely characterizing the optimal prefix codes for the TSGD has been open until very recently (some very partial answers existed in [5]), the approach was mainly *ad hoc*. Specifically, it was based on the analogy between the targeted distribution and the *one-sided* geometric distribution (OSGD) of non-negative integers, for which it is well-known that the *Golomb codes* [6] are optimal [7]. Given a positive integer parameter  $\ell$ , the Golomb code of order  $\ell$  encodes a non-negative integer  $y$  in two parts: a *modified binary* representation of  $y \bmod \ell$ , and a *unary* representation of  $\lfloor y/\ell \rfloor$ . Thus, given  $\ell$  (which depends on the rate of decay  $\theta$ ), the optimum Huffman tree has a structure that enables simple calculation of the code word of any given source symbol, without recourse to the storage of large code tables. In the context of adaptive coding, this property is crucial as it may relax the need for a dynamic update of these tables due to possible variations in the estimated parameter  $\theta$  (see, e.g., [8]).

A popular approach [9] is to “fold” the negative error values into the positive ones. Thus, an integer is encoded by applying a Golomb code to its index in the sequence  $0, -1, +1, -2, +2, -3, +3, \dots$ . This is motivated in part by the observation that the distribution of folded indices is in some sense “close” to (but not quite) an OSGD. In addition, a further reduction in complexity is obtained by reducing the family of Golomb codes to the case where  $\ell = 2^m$  for some non-negative integer  $m$ . Choosing  $\ell$  to be a power of 2 leads to extremely simple encoding/decoding procedures: the code word for a non-negative integer  $y$  consists of the  $m$  least significant bits of  $y$ , followed by the number formed by the remaining higher order bits of  $y$ , in *unary* representation (the simplicity of the case  $\ell = 2^m$  was already noted in [6]). The length of the encoding is  $m + 1 + \lfloor y/2^m \rfloor$  bits. Even though this strategy is sub-optimal for a TSGD centered at zero for every  $\theta > 1/3$ , except for isolated singular points (see Section 2 below), it is evidenced in [9] that the average code length obtained with the best code in the reduced family (i.e., the envelope of all code length curves as functions of  $\theta$ ) is often reasonably close to the entropy. It should be noticed, however, that the average code length is computed there under the assumption that the index sequence is distributed OSGD.

A different heuristic approach is proposed in [10], where a Golomb code is applied on the

*absolute values* of the prediction residuals, and then a sign bit is appended whenever a residual is non-zero. It is evidenced that the envelope of all code length curves as functions of  $\theta$  often coincides with the curve corresponding to the Huffman code, obtained through numerical computation. In [10], an explicit formula for the average length of the proposed codes is obtained under a true TSGD assumption.

The issue of sequential adaptivity is not directly addressed in the referenced works, as the above codes are used in the context of *block coding*. The source sequence is divided into fixed length blocks, which are scanned twice in order to use the best code in the corresponding family. A few overhead bits are needed to identify the appropriate code. In schemes that target for a higher compression through *context modeling* [11, 4, 12], a *sequential* approach is mandatory, as pixels encoded in a given context are not necessarily contiguous in the image and, thus, cannot be easily blocked. With this approach, each pixel is encoded independently, and the encoder determines the code parameter based on the values observed for previous pixels. The decoder makes the same determination, after decoding the same past sequence. In LOCO-I [4], the core of the prospective lossless/near-lossless image compression standard JPEG-LS [13], the same family of codes as in [9] is used, with a very simple adaptation strategy based on the accumulated sum of the *magnitudes* of prediction errors in the past observed sequence (for each context).

In addition, a more general model than the TSGD centered at zero is considered in [4]. Although the centered TSGD is an appropriate model for memoryless image compression schemes, it has been observed [4, 12] that prediction errors in context-based schemes exhibit systematic biases, and a more appropriate model is given by an *off-centered* TSGD. This model is also useful for better capturing the two adjacent modes often observed in empirical context-dependent histograms of prediction errors. More specifically, the generalized model is given by distributions of the form

$$P_{\theta,d}(x) = p_0(\theta, d)\theta^{|x+d|}, \quad 0 < \theta < 1, \quad 0 \leq d \leq 1/2, \quad (1)$$

where  $p_0(\theta, d) = (1 - \theta)/(\theta^{1-d} + \theta^d)$  is a normalization factor, and  $d$  is a parameter that determines the “offset” of the distribution. The restriction on the range of  $d$  is justified in practice through straightforward application of appropriate translations and reflections of the real line. For example, the interval containing the “center” of the distribution can be located by using an error feedback loop [4, 12], for which the “resolution” is a unit interval (integer-valued translation). Then, the reflection  $x \rightarrow -(x + 1)$  can be used to halve the relevant range of  $d$ . The centered TSGD corresponds to  $d = 0$ , and, when  $d = 1/2$ ,  $P_{\theta,d}$  is a bi-modal distribution with equal peaks at  $-1$  and  $0$ . (The preference of  $-1$  over  $+1$  here is arbitrary).

Recently, a complete characterization of optimal prefix codes for the TSGD (1) was presented [14]. The family of optimal codes is an extension of the Golomb codes, and for some values of the parameters  $\theta$  and  $d$  the optimal codes indeed coincide with the codes used in [9], [10], and [4]. The result of [14] makes it possible to approach in a more comprehensive way the design of low complexity adaptive strategies for encoding TSGD models, in the sense that the proposed strategies can be shown to derive from provably optimal schemes. In this paper, we elaborate on the adaptive coding of TSGD models on a symbol-by-symbol basis, and we present low complexity adaptive strategies that approach the performance of the optimal codes in [14].

The remainder of this paper is organized as follows. In Section 2, we discuss the optimal codes of [14] and we derive the corresponding average code lengths. In Section 3, we present

an optimal adaptive strategy corresponding to these codes. Finally, in Section 4, we make some complexity compromises which lead to very simple, yet efficient adaptive strategies for the family of TSGD models. We compare the performance of the proposed codes with the achievable optimum, in terms of average code length.

## 2 Optimal prefix codes for TSGD's

Throughout this section the values of the parameters  $\theta$  and  $d$  in (1) are assumed known, with  $0 < \theta < 1$  and  $0 \leq d \leq 1/2$ . The optimal prefix codes for TSGD's are described in [14] in terms of a partition of the parameter space  $(\theta, d)$ , such that each class corresponds to a different code. We omit here a detailed description of this partition, as only the following properties are relevant to our study:

- a) Each class in the partition is associated with one of four different types of codes, termed *Type I*, *II*, *III*, and *IV*, respectively.
- b) For each type of code, the classes are indexed with a positive integer  $\ell$ , which is given by an explicit many-to-one function  $(\theta, d) \rightarrow \ell(\theta, d)$ . For fixed  $d$ , this function is non-decreasing in  $\theta$ .
- c) The various types of codes alternate in the  $(\theta, d)$  plane as shown in Figure 1 for  $\ell = 1, 2$ , where the  $\theta_i(\ell, d)$  denote boundary curves between classes.

For any integer  $K > 0$ , let  $G_K$  denote the *Golomb* code [6] of order  $K$ . Also, for any integer  $x$ , define

$$M(x) = \begin{cases} 2x & x \geq 0, \\ 2|x| - 1 & x < 0. \end{cases} \quad (2)$$

**Theorem 1** [14] *Let  $x$  denote an integer-valued random variable distributed according to (1), and let  $\ell = \ell(\theta, d)$ . Then, an optimal prefix code for  $x$  is constructed by proceeding in the different classes as follows:*

**(Type I)** *Encode  $x$  using  $G_{2\ell-1}(M(x))$ .*

**(Type III)** *Encode  $x$  using  $G_{2\ell}(M(x))$ .*

**(Type II)** *Encode  $|x|$  using the code described below, and append a sign bit whenever  $x \neq 0$ . Let  $r$  be the integer satisfying  $2^{r-1} \leq \ell < 2^r$ , let  $s = 2^r - \ell$ , and let  $s' = s \bmod 2^{r-1}$ , where  $a \bmod b$  denotes the least nonnegative residue of  $a$  modulo  $b$ . Define*

$$z(x) = \begin{cases} s' & x = 0, \\ 0 & x = s', \\ x & \text{otherwise.} \end{cases} \quad (3)$$

*Encode  $|x|$  using  $G_\ell(z(|x|))$ .*

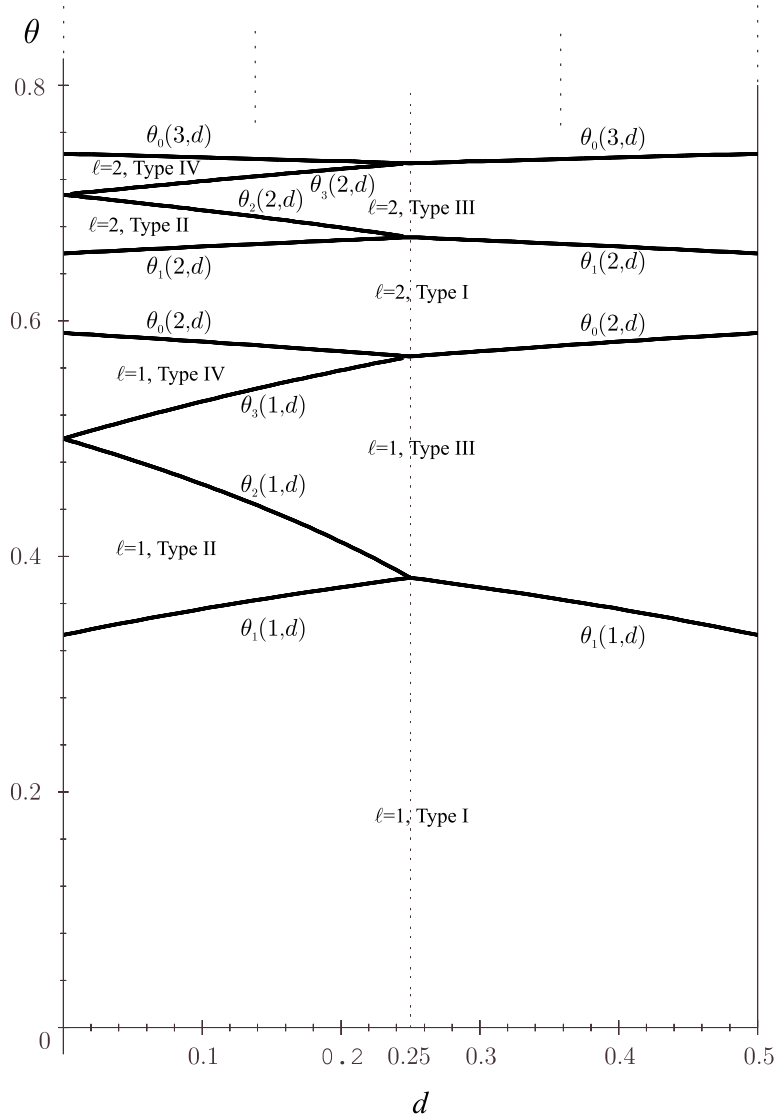


Figure 1: Optimal prefix code classes

**(Type IV)** Define  $s$  as in Type II, encode  $|x|$  using  $C(|x|)$  defined below, and append a sign bit whenever  $x \neq 0$ .

$$C(|x|) = \begin{cases} G_\ell(|x| - 1) & |x| > s, \\ G_\ell(|x|) & 1 \leq |x| < s, \\ G_\ell(0)0 & x = 0, \\ G_\ell(0)1 & |x| = s. \end{cases} \quad (4)$$

The codes of types I and III are *asymmetric*, in that they assign different code lengths to  $x$  and  $-x$  for some values of  $x$ . In contrast, the codes of types II and IV are symmetric.

As stated in Section 1, the mapping (2) was first employed in [9] to apply Golomb codes to alphabets that include both positive and negative numbers. Theorem 1 shows that this strategy (which was also used in [4] and always produces asymmetric codes) is optimal for values of  $(\theta, d)$  for which the optimal code is of type I or III, but is not so for classes that yield

optimal codes of type II or IV. Moreover, the codes in the minimal complexity sub-family used in both [9] and [4], for which the code parameter  $\ell$  is a power of 2, are *never* optimal in the centered case ( $d = 0$ ) considered in [9], except for the case  $\ell = 1$  (Type I) and for isolated singular points for which codes of Type III are optimal (see Figure 1). On the other hand, the codes in the sub-family of [10] (which are always symmetric) can *only* be optimal when  $\ell$  is a power of 2 (Type II), as only in this case  $z(x) = x$  for every non-negative integer  $x$  in (3) (namely,  $s' = 0$ ). Of course, the sub-optimality of its individual codes does not necessarily rule out a sub-family as a good approximation of the optimal family of Theorem 1.

Lemma 1 below summarizes the average code lengths of all the codes described in Theorem 1. In order to state Lemma 1, it will be convenient to re-parameterize the family of off-centered TSGD models (1) as

$$P_{\theta,\rho}(x) = \begin{cases} (1-\rho)(1-\theta)\theta^x & x = 0, 1, 2, \dots \\ \rho(1-\theta)\theta^{-x-1} & x = -1, -2, \dots \end{cases} \quad (5)$$

The real parameter  $\rho$  designates the probability that a random variable drawn according to (5) would be negative. It can be seen that when  $\rho$  is restricted to the range  $[\theta/(1+\theta), 1/2]$  and  $d \in [0, 1/2]$ , the model classes given in (1) and (5) coincide, with

$$\rho = \frac{\theta^{1-d}}{\theta^d + \theta^{1-d}}. \quad (6)$$

In the framework of adaptive coding, we will see in Section 3 that the representation (5) is also more convenient in terms of the estimation of its parameters.

The values  $\ell$ ,  $r$ ,  $s$ , and  $s'$  in Lemma 1 are defined as in Theorem 1. Also, logarithms are taken to base 2.

**Lemma 1** *For all  $(\theta, \rho)$ , the per-symbol average lengths of the codes defined for the various classes in Theorem 1 are as follows.*

**(Type I)**

$$\bar{L}_1(\theta, \rho) = 1 + \lfloor \log(2\ell - 1) \rfloor + \frac{\theta^{s'}}{1 - \theta^{2\ell-1}}(1 - P_{\theta,\rho}(0) + \theta^\ell). \quad (7)$$

**(Type III)**

$$\bar{L}_3(\theta, \rho) = 1 + \lfloor \log(2\ell) \rfloor + \frac{\theta^s}{1 - \theta^\ell}. \quad (8)$$

**(Type II)**

$$\bar{L}_2(\theta, \rho) = 1 + \lfloor \log \ell \rfloor + (1 - P_{\theta,\rho}(0))\theta^{s'} \left( 1 + \frac{\theta^{\ell-1}}{1 - \theta^\ell} \right). \quad (9)$$

**(Type IV)**

$$\bar{L}_4(\theta, \rho) = 2 + \lfloor \log \ell \rfloor + (1 - P_{\theta,\rho}(0))\theta^{s-1} \left( 1 + \frac{\theta^{\ell+1}}{1 - \theta^\ell} \right). \quad (10)$$

Notice that these average code lengths apply to all values  $(\theta, \rho)$ , and not just to the class for which a code is optimal. Also, the dependence of  $\bar{L}_1$ ,  $\bar{L}_2$ , and  $\bar{L}_4$  on  $\rho$  is given through  $P_{\theta,\rho}(0) = (1-\rho)(1-\theta)$ . The average code length  $\bar{L}_3$ , on the other hand, is independent of  $\rho$ . A special case of (9), in which  $\rho = \theta/(1+\theta)$  (centered TSGD) and  $\ell$  is a power of 2 (so that  $z(x) = x$  in (3)), was derived in [10].

### 3 Adaptive coding of TSGD's

In this section, we consider the case where the value of the parameters  $\theta$  and  $\rho$  (or  $d$ ) is unknown *a priori*, in the framework of symbol-by-symbol coding. Even though, in general, adaptive strategies are easier to implement with arithmetic codes, the construction of [14] suggests a reasonable alternative for low complexity adaptive coding of TSGD models: Given a sequential maximum-likelihood (ML) estimate of  $\theta$  and  $\rho$  based on the sequence of source symbols  $x^t = x_1, x_2, \dots, x_t$  processed so far, determine the corresponding value  $\ell(\theta, d)$  of the index  $\ell$  and the code-type for the class, and use the associated code to encode the next symbol  $x_{t+1}$ .

To obtain ML estimates of  $\theta$  and  $\rho$ , we define the indicator function  $u(x)$ , which takes the value 1 whenever the integer  $x$  is negative, and 0 otherwise. For a sequence of integers  $x^t$ , let

$$S_t = \sum_{i=1}^t (|x_i| - u(x_i)) \quad (11)$$

and

$$N_t = \sum_{i=1}^t u(x_i) \quad (12)$$

namely,  $N_t$  is the total number of negative samples in  $x^t$ , and  $S_t + N_t$  is the accumulated sum of absolute values. It can be readily shown that  $S_t$  and  $N_t$  serve as sufficient statistics for the distribution (5). The ML estimator  $\tilde{\rho}_t$  of  $\rho$  at time  $t$  is given by  $N_t/t$ , while the ML estimator of  $\theta$  is

$$\tilde{\theta}_t = S_t / (S_t + t). \quad (13)$$

Thus, it is convenient to replace the parameter  $\theta$  by a new parameter

$$S \triangleq \theta / (1 - \theta), \quad (14)$$

for which the ML estimate is simply  $S_t/t$ . In case  $\tilde{\rho}_t > 1/2$ , we apply the transform  $x \rightarrow -(x + 1)$  to the source symbol  $x_{t+1}$ , and conceptually to the past sequence  $x^t$ . The accumulated statistics  $S_t$  are invariant under this transform, while  $N_t$  undergoes  $N_t \rightarrow t - N_t$ , so that  $\tilde{\rho}_t \in [0, 1/2]$ . Also, if

$$N_t/t < S_t/(2S_t + t) \quad (15)$$

then the ML estimate of  $\tilde{\rho}_t$  in the allowed parameter range corresponds to a centered TSGD, namely

$$\tilde{\rho}_t = S_t / (2S_t + t). \quad (16)$$

Even though arithmetic coding is avoided, both the class determination for the estimated pair  $(\theta, \rho)$  and the encoding procedure may still be too complex in some applications. In these cases, a complexity compromise can be achieved by reducing the family of codes under consideration. Next, we present a specific family reduction, for which in Section 4 we derive both optimal and low complexity, further simplified, adaptation strategies. We should point out that this sub-family represents a specific optimality-complexity trade-off, and that similar derivations could be possible for other sub-families. For example, a further simplified sub-family, resulting from a more drastic reduction, is considered in [14], leading to an easier one-dimensional analysis.

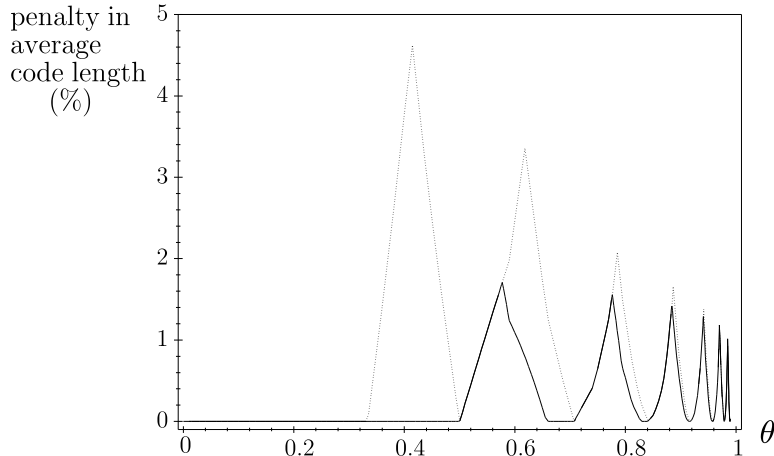


Figure 2: Code length penalty for  $\mathcal{C}$  and  $\mathcal{C}_{I,III}$  at  $d = 0$

First, following [9] and [4], we simplify the encoding procedure by limiting the sub-family under consideration to codes for which the associated Golomb parameter is a power of 2 (thus, only one code, with  $\ell = 1$ , is of Type I). This limitation also yields simpler adaptation formulas, as will be shown in Section 4. An additional complexity reduction stems from the fact that the mapping (3) used for codes of Type II becomes  $z(x) = x$ . Second, in order to avoid the implementation of the mapping (4) used for codes of Type IV, which would yield intricate adaptation procedures, we omit considering these codes. Specifically, let  $\mathcal{C}_{I,III}$  denote the family of (asymmetric) codes of types I and III for which  $\ell$  is a power of 2 (this is the family considered in [9]), and let  $\mathcal{C}_{II}$  denote the analogous family of (symmetric) codes of Type II (as discussed above, this is the subset of optimal codes in the family considered in [10]). Furthermore, let  $\mathcal{C}$  denote the union of  $\mathcal{C}_{I,III}$  and  $\mathcal{C}_{II}$ . We will derive both optimal and minimal complexity adaptation strategies, based on the sufficient statistics  $S_t$  and  $N_t$ , for  $\mathcal{C}$ . Moreover, our strategies will not be limited to the centered TSGD case.

The coding performance of the richer sub-family  $\mathcal{C}$  is illustrated in Figure 2, which shows the compression loss due to the restriction from the full family of optimal codes of Theorem 1 to  $\mathcal{C}$ , for a centered TSGD. It also shows, in the dotted line, the compression loss due to using  $\mathcal{C}_{I,III}$  alone, as in [9] and [4]. It can be seen that using only asymmetric codes can cause compression losses close to 4.7% (for  $\theta = \sqrt{2} - 1$ , i.e.,  $S = \sqrt{2}$ ).

## 4 Low complexity adaptive coding of TSGD's

In this section, we restrict our study on adaptive coding to the codes in  $\mathcal{C}$ . We neglect second-order effects on the code length, disregarding the vanishing adaptation cost due to the convergence of the estimates to the true parameter values (“model-learning cost”). In other words, it is assumed that the ML estimates  $S_t/t$  and  $N_t/t$  at time  $t$  already converged to the parameters  $S$  and  $\rho$ , respectively. First, we derive the optimal decision regions to choose among the codes in the family. By the above discussion, these regions are described in Theorem 2 below in terms of  $S$  and  $\rho$ , although we also use  $\theta$  wherever convenient. This theorem follows from Lemma 1.



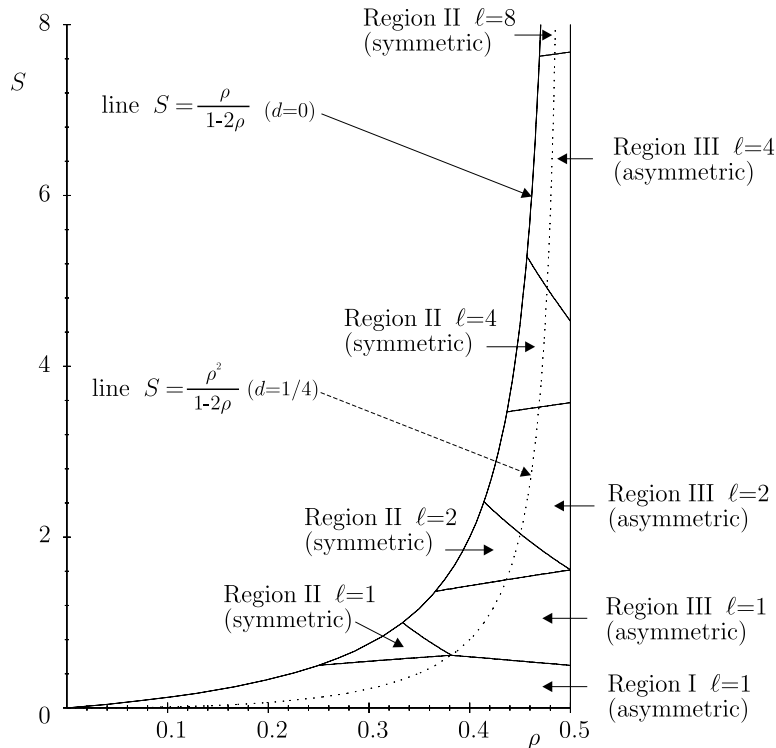


Figure 3: Optimal adaptation regions for  $\mathcal{C}$

**Theorem 2** *The optimal codes in  $\mathcal{C}$  alternate in the  $(S, \rho)$  plane as depicted in Figure 3, with boundaries between regions (labeled according to the corresponding code-type) given as follows:*

- (i) *The boundary between regions I and III, indexed by  $\ell = 1$ , is  $\rho = 1 - S$ .*
- (ii) *The boundary between regions I and II, indexed by  $\ell = 1$ , is  $\rho = S^2$ .*
- (iii) *The boundary between Region III, indexed by  $\ell$ , and Region II, indexed by  $2\ell$ , for every  $\ell > 0$  ( $\ell$  a power of 2), is*

$$\rho = -S \frac{\theta^{2\ell} + \theta^{\ell-1} - 1}{\theta^{2\ell} - \theta^{2\ell-1} - 1}, \quad (17)$$

*which is an increasing function of  $S$  (or  $\theta$ ).*

- (iv) *The boundary between regions II and III, both indexed by  $\ell$ , for every  $\ell > 0$  ( $\ell$  a power of 2), is*

$$\frac{1 - \rho}{\rho} = \frac{\theta^{\ell-1}}{1 - \theta^\ell}, \quad (18)$$

*which is a decreasing function of  $S$  (or  $\theta$ ).*

The exact determination of the regions defined by Theorem 2 is still computationally expensive. However, Lemma 2 below states some properties of the boundary curves which suggest that these curves can be easily approximated with low complexity computations. Lemma 2 involves the constant  $\phi \triangleq (\sqrt{5} - 1)/2$  (inverse of the golden ratio), which also appears in [3] in connection with Golomb codes.

**Lemma 2** (i) The boundary (17) intersects the curve  $\rho = S/(2S + 1)$  (i.e., the centered distribution) at points where the value of  $S$  is

$$S_1(\ell) = \frac{-\ell}{\ln \phi} - \epsilon_1(\ell), \quad (19)$$

where  $\epsilon_1$  is a decreasing function of  $\ell$ , that ranges between  $-\ln^{-1} \phi - (\sqrt{3}-1)^{-1} \approx 0.71$  ( $\ell = 1$ ) and  $(1/2) - (1 - \phi)/(2\sqrt{5} \ln \phi) \approx 0.68$  ( $\ell \rightarrow \infty$ ).

(ii) The boundary (17) intersects the curve  $\rho = 1/2$  (i.e., the maximum value of  $\rho$  in the relevant range) at points where the value  $S_2(\ell)$  of  $S$  satisfies

$$S_1(\ell) < S_2(\ell) < S_1(\ell) + 1/(3.8\ell). \quad (20)$$

(iii) The boundary (18) intersects the curve  $\rho = S/(2S + 1)$  at points where the value of  $\theta$  is  $2^{-1/\ell}$ . The corresponding value of  $S$  is

$$S_3(\ell) = \frac{\ell}{\ln 2} - \frac{1}{2} + \epsilon_3(\ell), \quad (21)$$

where  $\epsilon_3$  is a decreasing function of  $\ell$ , that ranges between  $3/2 - 1/\ln 2 \approx 0.06$  ( $\ell = 1$ ) and 0 ( $\ell \rightarrow \infty$ ).

(iv) For every  $\ell > 1$ , the boundary (18) intersects the curve  $\rho = 1/2$  at points where the value of  $S$  is

$$S_4(\ell) = \frac{\ell - 1}{\ln 2} + \epsilon_4(\ell), \quad (22)$$

where  $\epsilon_4$  is an increasing function of  $\ell$ , that ranges between  $1 + \phi - (1/\ln 2) \approx 0.18$  ( $\ell = 2$ ) and  $(1/(\ln 2) - 1)/2 \approx 0.22$  ( $\ell \rightarrow \infty$ ).

## Corollaries from Lemma 2

- a. Since  $-\ln^{-1} \phi \approx 2.078$ , (19) implies that  $S_1(\ell) + 1/2$  is within 4% of  $2\ell$  (a power of 2) for every  $\ell > 1$ . By (20), this also holds for  $S_2(\ell) + 1/2$ . By the monotonicity of the boundaries (17), it follows that we can approximate these boundaries by the lines  $S = 2\ell - 1/2$  for every  $\ell > 1$ , independently of  $\rho$ . The fact that the boundary  $S$  between approximate regions is such that  $S + 1/2$  is a power of 2, results in a very simple adaptation.
- b. Since  $-\ln^{-1} 2 \approx 1.44$ , (21) implies that  $S_3(\ell) + 1/2$  is also within 4% of  $3\ell/2$  for every  $\ell > 0$ . Although the difference is larger for  $S_4(\ell)$ , the loss caused by approximating these boundaries by the lines  $S = (3\ell - 1)/2$  (thus using a code of Type II in cases where the estimated parameter values indicate that a code of Type III is optimum) can be reasonably bounded for every  $\ell > 2$ . Specifically, it can be shown that the maximum loss due to the use of a code of Type II where a code of Type III would have been appropriate, occurs when  $\rho = 1/2$  and  $S$  takes the maximum value in the mismatch region, namely  $S = (3\ell - 1)/2$  in this case. It can further be shown that in this case the maximum loss  $\Delta L_{max}$  is upper-bounded as

$$\Delta L_{max} = \frac{1}{3\ell - 1} \left( \frac{1}{[2/(3\ell + 1) + 1]^\ell - 1} - \frac{1}{1 + 2/(3\ell - 1)} \right) < \frac{0.21}{3\ell - 1}. \quad (23)$$

On the other hand, the average length of the optimum prefix code in Region III, index  $\ell$ , can be shown to be at least  $1 + \log(2\ell) + \phi$  bits (as  $\ell$  is a power of 2,  $s = \ell$  in (8), and it can be shown that  $\theta^\ell > \phi^2$ ). Thus, the worst case relative loss occurs for  $\ell = 4$  and it is less than 0.41%. In fact, a similar line of reasoning can be employed to show that, for  $\ell > 2$ , the worst case loss derived from using  $\mathcal{C}_{II}$  *only*, is always smaller than the worst case loss corresponding to  $\mathcal{C}_{I,III}$ .

The result of the above corollaries is that for  $\ell > 2$  the boundaries between regions can be approximated by the lines  $S = 2\ell - (1/2)$  and  $S = (3\ell - 1)/2$ , independent of  $\rho$ , which are very easily handled as  $\ell$  is a power of 2. This holds in the first case for  $\ell = 2$  as well. The remaining boundaries also admit reasonable linear approximations, which were designed under the low complexity constraint that every multiplication by a constant can be performed with no more than two shifts and one addition/subtraction. This leads to the following simplified adaptive coding strategy, which assumes that  $x_{t+1}$  is encoded with  $N_t < t/2$  (otherwise, the symbol  $-(x_{t+1} + 1)$  is encoded).

### Encoding procedure

Let  $A \triangleq S_t + t/2$ .

**If**  $A > 4t$ ,

Let  $m$  be the least integer such that  $2^{m+1}t \geq A$  ( $m \geq 2$ ), and let  $\ell = 2^m$ .

**If**  $A \leq 3t2^{m-1}$ , use Type II with index  $\ell$ . **Else**, use Type III with index  $\ell$ .

**Else**, let  $B \triangleq S_t - t$ .

**If**  $12B > 63t - 112N_t$ , use Type III with  $\ell = 2$ .

**Else if**  $16B > 5(6N_t - t)$ , use Type II with  $\ell = 2$ .

**Else if**  $3B > 8(t - 3N_t)$  and  $B > -N_t$ , use Type III with  $\ell = 1$ .

**Else if**  $9(S_t + B) > 16N_t - 4t$ , use Type II with  $\ell = 1$ .

**Else**, use Type I with  $\ell = 1$ .

Update: **If**  $x_{t+1} < 0$ ,  $N_{t+1} = N_t + 1$  and  $S_{t+1} = S_t + |x_{t+1}| - 1$ .

**Else**,  $N_{t+1} = N_t$  and  $S_{t+1} = S_t + |x_{t+1}|$ .

**End.**

The additional code length penalty resulting from the approximate adaptation strategy is negligible compared to the original penalty caused by the restriction in code family. This is illustrated in Figure 4, which shows the overall code length penalty (relative to the optimal prefix code) due to both code family restriction and approximate adaptation, for three representative values of  $d$ . As expected from (23), the maximum additional penalty due to approximate adaptation is about 0.41%, as seen in the “spike” at  $\theta = 11/13 \approx 0.85$ ,  $d = 0.5$ . Overall, the code family  $\mathcal{C}$ , with approximate code region determination as per the above procedure, is estimated to be within 1.8% of the average code length for the optimal prefix code family, with exact code region determination. While this estimate has not been formally proven, we expect it to follow from techniques similar to the ones used in Lemma 2 and its corollaries.

Notice that when the TSGD is assumed to be centered (line  $S = \rho/(1 - 2\rho)$ ), the above encoding procedure is particularly simple, as the boundary points between regions II and III are  $S = 2\ell - (1/2)$ ,  $\ell > 1$ , and  $S = (3\ell - 1)/2$ ,  $\ell > 0$ . In addition, the remaining boundaries are located at  $S = 1/(\sqrt{3} - 1) \approx 1 + 3/8$  (Region III,  $\ell = 1$ , and Region II,  $\ell = 2$ ), and

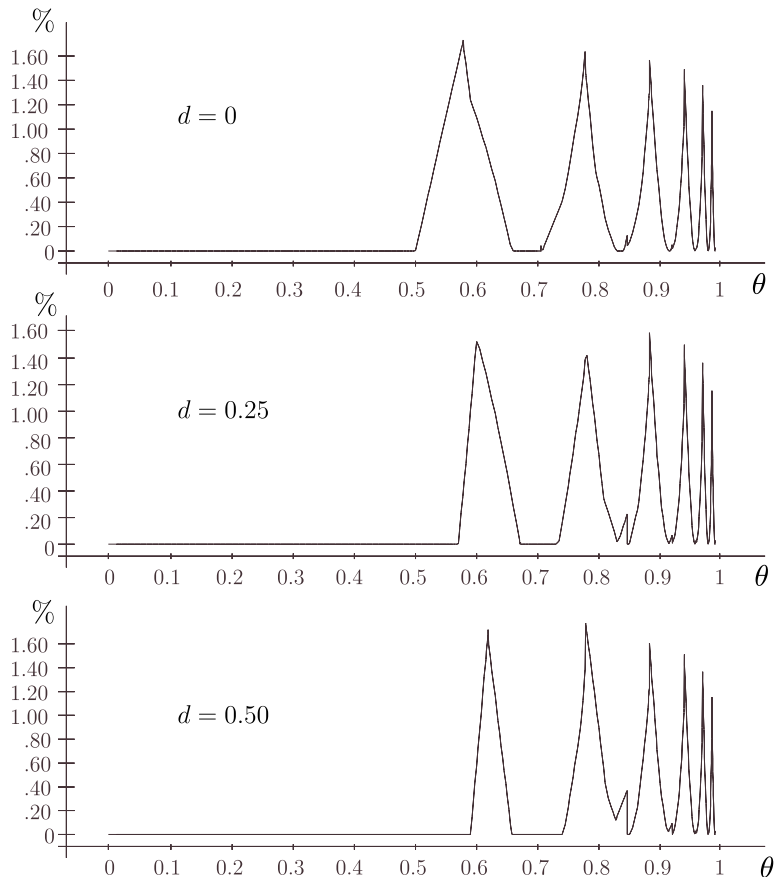


Figure 4: Code length penalty for  $\mathcal{C}$  with approximate adaptation strategy

$S = 1/2$  (regions I and II,  $\ell = 1$ ). However, even though  $S_t/t$  can be used as an estimate of  $S$  as in the general case, it should be noticed that in this case it *is not* the ML estimator. A decision rule based on ML estimation should rather depend on  $S_t + N_t$ , as the accumulated sum of absolute values serves as sufficient statistics in the centered case. This rule can be derived by similar methods.

The next natural step would be to add codes of Type IV to the adaptive strategy, with  $\ell$  a power of 2. This would entail higher complexity, and in any case the coding penalty will still remain at least 1.2%.

## References

- [1] A. Netravali and J. O. Limb, "Picture coding: A review," *Proc. IEEE*, vol. 68, pp. 366–406, 1980.
- [2] J. O'Neal, "Predictive quantizing differential pulse code modulation for the transmission of television signals," *Bell Syst. Tech. J.*, vol. 45, pp. 689–722, May 1966.
- [3] P. G. Howard and J. S. Vitter, "Fast and efficient lossless image compression," in *Proc. of the 1993 Data Compression Conference*, (Snowbird, Utah, USA), pp. 351–360, Mar. 1993.
- [4] M. J. Weinberger, G. Seroussi, and G. Sapiro, "LOCO-I: A low complexity, context-based, lossless image compression algorithm," in *Proc. of the 1996 Data Compression Conference*, (Snowbird, Utah, USA), pp. 140–149, Mar. 1996.
- [5] J. Abrahams, "Huffman-type codes for infinite source distributions," *J. Franklin Inst.*, vol. 331B, no. 3, pp. 265–271, 1994.
- [6] S. W. Golomb, "Run-length encodings," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 399–401, July 1966.
- [7] R. Gallager and D. V. Voorhis, "Optimal source codes for geometrically distributed integer alphabets," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 228–230, Mar. 1975.
- [8] D. E. Knuth, "Dynamic Huffman coding," *J. Algorithms*, vol. 6, pp. 163–180, 1985.
- [9] R. F. Rice, "Some practical universal noiseless coding techniques - parts I and III," Tech. Rep. JPL-79-22 and JPL-91-3, Jet Propulsion Laboratory, Pasadena, CA, Mar. 1979 and Nov. 1991.
- [10] K.-M. Cheung and P. Smyth, "A high-speed distortionless predictive image compression scheme," in *Proc. of the 1990 Int'l Symposium on Information Theory and its Applications*, (Honolulu, Hawaii, USA), pp. 467–470, Nov. 1990.
- [11] M. J. Weinberger, J. Rissanen, and R. Arps, "Applications of universal context modeling to lossless compression of gray-scale images," *IEEE Trans. Image Processing*, vol. 5, pp. 575–586, Apr. 1996.
- [12] X. Wu, "An algorithmic study on lossless image compression," in *Proc. of the 1996 Data Compression Conference*, (Snowbird, Utah, USA), pp. 150–159, Mar. 1996.
- [13] ISO/IEC JTC 1/SC 29/WG 1, "Lossless/near-lossless compression, working document WD14495," 1996.
- [14] N. Merhav, G. Seroussi, and M. J. Weinberger, "Modeling and low-complexity adaptive coding for image prediction residuals," in *Proc. of the 1996 Int'l Conference on Image Processing*, vol. II, (Lausanne, Switzerland), pp. 353–356, Sept. 1996.