Towards Lossy Lempel-Ziv: Natural Type Selection

Ram Zamir and Kenneth Rose

Dept. of Elect. & Comp. Eng., University of California, Santa Barbara, CA 93106

Abstract — **We present a sequential lossy compression algorithm, almost all of whose dictionary strings are asymptotically of the same type as the rate-distortion achieving reproduction distribution, and whose coding rate can approach the rate distortion bound.**

I. Wild Growing Tree Codes

In the Lempel-Ziv algorithm [1], a tree-code is sequentially grown by matching strings from a source sequence with existing paths in a tree, and extending the selected paths. We assume a version of LZ78 algorithm where all possible extensions of the selected path by one symbol are added to the dictionary. Thus the dictionary is a complete prefix code and internal (nonleaf) nodes need not be retained. A natural lossy variant of LZ78 is obtained by replacing perfect matching with lossy D-matching: a source string $\mathbf{x} \in \mathcal{X}^n$ is encoded into a path (or leaf) $\mathbf{y} \in \mathcal{Y}^n$ of the tree only if $d(\mathbf{x},\mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^n d(x_i, y_i) \leq D$, where $\mathcal{Y}$ is the (finite) reconstruction alphabet, and d is an additive distortion measure. There is always at least one D-match, and if there are more we select one of them *at random*. Note that this fixed-distortion variable-rate algorithm reduces to the ordinary LZ algorithm for $D = 0$.

A basic property of the lossless LZ tree-code is that it favors growing strings (paths) of the source type. We make the surprising (and exciting) observation that the lossy tree-code favors growing branches of the type Q^* corresponding to the reproduction distribution that achieves the rate-distortion function of the source. Given this fact, and the optimality of non-universal tree-codes [2], we are led to believe that a sequentially grown lossy tree code is asymptotically optimal. At the current stage of the research, however, we are able to show our claims only with respect to a modification of the above lossy LZ algorithm, and we further restrict the discussion to discrete memoryless sources.

II. A Key Quantity

We define the quantity

$$I_m(P\|Q, D) = \min_{W \in \mathcal{W}_{P,Q,D}} I_W(X;Y) \ , \qquad (1)$$

where $\mathcal{W}_{P,Q,D}$ is the set of joint distribution over $\mathcal{X} \times \mathcal{Y}$ whose projections (marginals) on $\mathcal{X}$ and $\mathcal{Y}$ are P and Q, respectively, and whose distortion is $\leq D$. If $\mathcal{W}_{P,Q,D}$ is empty, we define $I_m = \infty$. Note that for $X \sim P$ we get $I_m(P\|Q,D) \geq R_X(D)$, with equality iff $Q = Q^*$.

The quantity I_m plays an important role in lossy string matching and governs the "D-match exponent":

$$-\frac{1}{n}\log \Pr(d(\mathbf{X},\mathbf{y}) \leq D) = I_m(P\|Q_{\mathbf{y}}, D) + O(\log n/n) \ , \quad (2)$$

where $\mathbf{X} \in \mathcal{X}^n$ is uniformly distributed over the type class $T(P)$, and $Q_{\mathbf{y}}$ is the type of $\mathbf{y} \in \mathcal{Y}^n$. From this property it follows that when a source $X \sim P^*$ is encoded via string matching with an independent database $Y \sim Q$ (e.g., in [3] $\mathcal{X} = \mathcal{Y}$ and $Q = P^*$), the coding rate is given by $\min_{Q'}[\mathcal{D}(Q'\|Q) + I_m(P^*\|Q', D)]$.

III. A Constrained Growing Tree Code

Let $0 \leq R_u \leq \log|\mathcal{Y}|$ be a prespecified target rate. The modified algorithm grows the tree in a sequence of stages. At the n-th stage the depth of the tree is n, and we seek to D-match source strings of length n. Associated with each leaf are a probability for its selection, and a counter. For every incoming source n-string, the leaves of the tree are scanned at a random order according to their probabilities, and each candidate codeword is randomly permuted[1]. The index of the first D-matching leaf is transmitted, and its counter is incremented. If no D-match is found for a source string, it is transmitted uncompressed. Once $\lceil 2^{nR_u} \rceil$ source n-strings have been transmitted, the $n+1$st layer of the tree is grown by branching from each of the leaves whose counter value is positive into $|\mathcal{Y}|$ new leaves. The probability associated with each new leaf is chosen to be proportional to the counter value of its parent leaf, while its own counter is set to zero, and the tree is ready to encode $(n+1)$-strings.

We first derive a recursive formula for the probability $P_n(Q)$ of D-matching a leaf of type $Q \in \mathcal{Q}_n$ at stage n, where $\mathcal{Q}_n$ denotes the set of types over $\mathcal{Y}^n$. For source $X \sim P^*$ and large n,

$$P_n(Q) \doteq \frac{w_n(Q) \cdot 2^{-nI_m(P^*\|Q,D)}}{\sum_{Q' \in \mathcal{Q}_n} w_n(Q') \cdot 2^{-nI_m(P^*\|Q',D)}} \ , \qquad (3)$$

where $w_n(Q) = \frac{1}{|\mathcal{Y}|}\sum_{Q' \in \mathcal{A}(Q)} P_{n-1}(Q')$, and where $\mathcal{A}(Q) \subset \mathcal{Q}_{n-1}$ denotes the set of parent types of type Q. Observe that $P_n(Q)$ collapses on the optimal reproduction distribution $Q = Q^*$ as $n \to \infty$.

The following theorem implies that the coding rate of the algorithm asymptotically approaches the target rate R_u as long as it is greater than the rate-distortion function of the source:

Theorem *If $R_u > R_X(D)$, then*

$$\lim_{n\to\infty} \Pr[D\textit{-match at stage } n] = 1 \ . \qquad (4)$$

References

[1] J. Ziv and A. Lempel. Compression of individual sequences via variable rate coding. IT-24:530–536, Sept. 1978.

[2] F. Jelinek. Tree encoding of memoryless time-discrete sources with a fidelity criterion. IT-15:584–590, 1969.

[3] Y. Steinberg and M. Gutman. An algorithm for source coding subject to a fidelity criterion, based on string matching. *IEEE Trans. Information Theory*, IT-39:877–886, May 1993.

[0] This work was supported in part by the NSF under grant no. NCR-9314335, the UC MICRO program, DSP Group, Echo Speech, Moseley Associates, National Semiconductor, Qualcomm, Rockwell International, and Texas Instruments.

[1] This (pseudo-) random permutation eases our analysis, but is apparently unnecessary.