Low-Complexity Sequential Lossless Coding for Piecewise-Stationary Memoryless Sources

Gil I. Shamir, *Student Member, IEEE*, and Neri Merhav, *Fellow, IEEE*

Abstract— **Three strongly sequential, lossless compression schemes, one with linearly growing per-letter computational complexity, and two with fixed per-letter complexity, are presented and analyzed for memoryless sources with abruptly changing statistics. The first method, which improves on Willems' weighting approach, asymptotically achieves a lower bound on the redundancy, and hence is optimal. The second scheme achieves redundancy of $O(\log N/N)$ when the transitions in the statistics are large, and $O(\log\log N/\log N)$ otherwise. The third approach always achieves redundancy of $O(\sqrt{\log N/N})$. Obviously, the two fixed complexity approaches can be easily combined to achieve the better redundancy between the two. Simulation results support the analytical bounds derived for all the coding schemes.**

Index Terms— **Change detection, ideal code length, minimum description length, piecewise-stationary memoryless source, redundancy, segmentation, sequential coding, source block code, strongly sequential coding, transition path, universal coding, weighting.**

I. Introduction

TRADITIONAL sequential universal lossless source coding schemes are usually designed for classes of stationary sources. Not surprisingly, these schemes may perform poorly when the source is nonstationary, unless some adaptation mechanism is applied. While adaptive schemes such as the dynamic Huffman code [4], [9], [18], [25] and variations of the sliding-window Lempel–Ziv algorithm [24], [27], [28] have been developed and applied for general nonstationary sources, much less attention has been devoted to systematic, rigorous theoretical development of universal codes for *simple* classes of nonstationary sources. One example of such a class is that of memoryless sources with piecewise-fixed letter probabilities [12]–[13], [19]–[22], namely, sources for which the probability mass function (PMF) is subjected to occasional abrupt changes. This model is useful in several application areas, like compression of speech or text retrieved from several sources, edge information in images, and abrupt scene changes in video coding.

In this paper, we adopt this simple model of a *Piecewise-Stationary Memoryless Source (PSMS)*. Neither the source parameters at any stationary segment, nor the transition locations and their number are assumed to be known in advance. One can show that traditional adaptation mechanisms combined with classical compression schemes perform poorly for this class. Dynamic Huffman coding requires large block length, and thus exponentially large dictionary, in order to approach the entropy even in a stationary segment. Variations of the Lempel–Ziv (LZ) algorithm require increasing window length, which results in slow convergence to the source entropy. One may use adaptive entropy coding with respect to (w.r.t.) estimated letter probabilities across a sliding window or an exponential one, but such estimates have nondecaying variance, and thus yield poor coding performance.

This calls for a different approach. First, recall the well-known fact that, ignoring asymptotically negligible integer length constraints, the problem of sequential lossless coding (using, e.g., arithmetic coding [7], [8]), is completely equivalent to the problem of sequential probability assignment, where the length function of the code is understood as the negative logarithm of the assigned probability, i.e., the *ideal code length*. Hence, we can treat the latter problem instead of the former, and we do so from this point on.

To the best of our knowledge, universal coding for the PSMS model was first investigated by Merhav [12] (see also [13]). Merhav showed that the average universal coding redundancy over all sequences of N letters, drawn from an alphabet of r letters by almost any PSMS with a *fixed* number C of transitions between segments each of length $O(N)$, is lower-bounded by

$$R_N \geq (1-\varepsilon)\left(\frac{r-1}{2}(C+1)+C\right)\frac{\log N}{N} \tag{1}$$

where $\varepsilon > 0$ is a positive number and the base of the logarithmic function is 2. This bound was presented as a sum of two terms: The first term, henceforth referred to as *parameter redundancy* (PR), corresponds to universality w.r.t. the unknown source parameters within each stationary segment. It consists of $0.5\log N/N$ bits per symbol for each component of the parameter vector and each stationary segment (see, Rissanen [15]). The second term, henceforth referred to as *transition redundancy* (TR), corresponds to universality w.r.t. the unknown transition times from one stationary segment to

Manuscript received December 1, 1997; revised January 2, 1999. This work was supported by the Israeli Consortium of Ground Stations for Satellite Communications administered by the S. Neaman Institute for Science and Technology, and by the Israel Science Foundation administered by the Israeli Academy of Sciences and Humanities. This work was performed while G. I. Shamir was with the Department of Electrical Engineering, Technion–Israel Institute of Technology, Haifa, Israel. The material in this paper was presented in part at the IEEE International Symposium on Information Theory, Boston, MA, August 16–21, 1998. This work partially summarizes the M.Sc. research work of G. I. Shamir.

G. I. Shamir is with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA.

N. Merhav is with the Department of Electrical Engineering, Technion–Israel Institute of Technology, Technion City, Haifa 32000, Israel.

Communicated by R. Laroia, Associate Editor for Source Coding.

Publisher Item Identifier S 0018-9448(99)04355-2.

0018–9448/99$10.00 © 1999 IEEE

another. This term consists of $\log N/N$ bits per symbol for each such transition. Note that the derivation of the lower bound requires C to be fixed (see [12]) and all segments to be of length $O(N)$. If C is larger than $O(1)$, an algorithm with an upper bound of the form

$$R_N \leq (1+\varepsilon)\left(\frac{r-1}{2}(C+1)+C\right)\frac{\log(N/C)}{N} \tag{2}$$

can be obtained (refer to Theorem 1), and therefore the lower bound must be smaller or equal to this bound. Note, however, that if the transitions are not evenly spaced, that is some segments are shorter than $O(N/C)$, even smaller lower bounds can be obtained that are logarithmic with the length of the dominant longest stationary segments and linear with their number, which is smaller than C.

In [12], Merhav also demonstrated a universal compression scheme for the PSMS that achieves the lower bound of (1). This scheme employs the method of mixtures in two stages. The first-stage mixture gives Krichevskiy–Trofimov probability estimates [10] for each set of transition times, henceforth referred to as *transition path*. The second-stage mixture is performed over all possible transition paths. A *strongly sequential* version of this scheme was also obtained. That is, a scheme that sequentially updates the conditional coding probability of the next symbol given the past, independently of the future and of the horizon N. Merhav's scheme is of linearly increasing per-letter coding complexity when we assume at most a single transition. It can be generalized to any fixed number of transitions, yielding an algorithm of polynomially increasing complexity, and to an exponentially increasing complexity scheme if no assumption of a fixed number of transitions is made.

Three strongly sequential schemes of smaller but still increasing complexity for universal coding of PSMS's were later proposed by Willems [19]–[22]. These schemes are all based on context tree coding [23] combined with arithmetic coding. They all obtain redundancy of at least $O(\log N/N)$, but with coefficients larger than the coefficient of the lower bound in [12]. Willems implemented two-stage mixtures as described above by constructing suitable state diagrams for the second-stage mixture. The weight of a transition path in the mixture is hence obtained by state transition weights along the path. The first two schemes ([19]–[21]) take into account all transition paths. In [20] and [21], all transition paths that assume the same most recent transition time are unified into one diagram state. This results in linearly increasing per-letter computational complexity, storage complexity of $O(N\log N)$, and redundancy of $0.5(C+1)\log N/N$ beyond the lower bound. The second scheme [19], [20], groups transition paths into states according to both the last transition time and the hypothesized number of transitions thus far. The resulting diagram contains more states, each representing fewer transition paths. This leads to quadratically increasing per-letter complexity and a total redundancy of $0.5\log N/N$ beyond the bound. The third approach, proposed recently in [22], selectively eliminates states according to the time they were created. This scheme is still of increasing complexity of $O(\log N)$ and achieves per-letter redundancy of $O(\log^2 N/N)$ overall. To the best of our knowledge, no fixed complexity universal scheme has been obtained for PSMS's.

In this paper, we derive, analyze, and present simulation results of three new universal strongly sequential data compression schemes for PSMS's based on context tree coding. The first scheme achieves the lower bound on the redundancy, whereas the two other schemes provide slower decay rate of the redundancy, but have *fixed complexity* (and logarithmic bit storage complexity). The first scheme is a generalization of Merhav's scheme which uses Willems' linear transition diagram with different weights. Similarly as Willems' scheme, it is of linearly increasing per-letter complexity. Two different versions of this scheme that differ only in the transition weights are proposed. Unlike all the schemes presented in [12], [19]–[22], for which the number of states grows with time, the last two schemes have a fixed number of states, and hence can be applied for practical purposes. Although the convergence rate does not meet the bound, it is better than any existing low-complexity scheme for PSMS's.

The second scheme combines decisions based on the observed past with a reduced-state transition diagram, using different state transition weights than those used by Willems. It uses decisions to eliminate unlikely states in the diagram, thus preserving a fixed number of states. This scheme achieves *average* redundancy of $O(\log N/N)$ for large transitions if the number of stationary segments is upper-bounded by some constant S that depends on the design parameters of the scheme. Otherwise, *pointwise* redundancy (for any N-tuple) of at most $O(\log\log N/\log N)$ is obtained. In the third scheme, we partition the data sequence into smaller blocks and encode each one separately. Using the optimal block length, we achieve pointwise redundancy of $O(\sqrt{\log N/N})$. Simulations show that the true redundancies of the last two schemes are even better than the upper bounds obtained.

We can easily combine both fixed complexity schemes to obtain the better redundancy between the two for any sequence. We hence obtain a maximum upper bound on the pointwise redundancy of $O(\sqrt{\log N/N})$, which is better than known fixed-complexity schemes and, in fact, better than the currently known $O(1/\log N)$ upper bound of the Lempel–Ziv algorithm (see [11], [16]).

The outline of this paper is as follows. Section II contains notation and definitions. In Section III we generalize the analysis for the redundancy of a coding scheme. Section IV presents the first scheme of linear per-letter complexity. In Section V we describe and analyze the second scheme of decisions and weighting. Section VI presents the block partitioning scheme along with the analysis of its rate of convergence. Numerical results are presented in Section VII. Finally, in Section VIII, we present the summary and conclusions of this work.

II. Notation and Definitions

Let $\{P_\theta\}$ be a parametric family of memoryless stationary PMF's of vectors whose components take on values in a finite alphabet Σ of size r. The parameter θ designates the $(r-1)$-dimensional vector of letter probabilities. A string drawn by the source from time instant i to time instant j,

$(x_i, x_{i+1}, \cdots, x_j), j > i$, will be denoted by x_i^j. Let

$$x_1^N \triangleq x^N \triangleq (x_1, x_2, \cdots, x_n, \cdots, x_N)$$

be a string emitted from an r-ary PMF whose parameter θ takes on a particular value θ_0 from $n = 1$ to $n = t_1 - 1$; then $\theta = \theta_1$ from $n = t_1$ until $n = t_2 - 1$, and so on. Finally, from $n = t_C$ to $n = N$, θ is held at θ_C. The vectors

$$\{x_1, \cdots, x_{t_1 - 1}\}, \{x_{t_1}, \cdots, x_{t_2 - 1}\}, \cdots, \{x_{t_C}, \cdots, x_N\}$$

will be referred to as *stationary segments*, and correspondingly, $\theta_0, \theta_1, \cdots, \theta_C$ will be called the *segmental parameters*. It will be assumed that the different segments are statistically independent. The extended vector $(\theta_0, \theta_1, \cdots, \theta_C)$ will be denoted by Θ, and will be referred to as the *parameter set*. The C-dimensional vector, representing the C time instants *before* which transitions take place, $(t_1, t_2, \cdots, t_C)$, will be denoted by $\mathcal{T}$, and referred to as the *true transition path*. For convenience, we define $t_0 \triangleq 1$ and $t_{C+1} \triangleq N + 1$. We will assume that the number of transitions C is either fixed or is of lower order than the time n. Noting that C is a function of the dimension of the other parameters, the PMF of the PSMS is parameterized by the pair $\{\Theta, \mathcal{T}\}$, and defined as follows:

$$P(x^N \mid \Theta, \mathcal{T}) = \prod_{i=0}^{C} P_{\theta_i}(x_{t_i}, \cdots, x_{t_{i+1}-1}) \tag{3}$$

where the PMF of each segment is obtained by

$$\begin{aligned} P_{\theta_i}(x_{t_i}, \cdots, x_{t_{i+1}-1}) &= \prod_{n=t_i}^{t_{i+1}-1} P_{\theta_i}(x_n) \\ &= \prod_{u \in \Sigma} P_{\theta_i}(u)^{n_{t_i}^{t_{i+1}-1}(u)} \end{aligned} \tag{4}$$

where $P_{\theta_i}(x_n)$ is the probability of the letter x_n drawn by P_{θ_i}, which for simplicity will be denoted by P_i, and $n_{t_i}^{t_{i+1}-1}(u)$ denotes the number of occurrences of $u \in \Sigma$ within the ith segment.

The per-letter average entropy of a PSMS is obtained by

$$H(\Theta, \mathcal{T}) \triangleq \frac{1}{N} \sum_{i=0}^{C} (t_{i+1} - t_i) H(\theta_i) \tag{5}$$

where $H(\theta_i)$ is the entropy of the ith segment.

Since we assume no prior knowledge of $\{\Theta, \mathcal{T}\}$, we will not be able to assign the true probability of the sequence for a coding scheme. Instead, we will seek a universal sequential probability assignment that will implement a two-stage mixture and will serve as the basis for arithmetic coding. The probability assigned to the substring x^n by an algorithm $\mathcal{A}$ will be denoted by $Q_{\mathcal{A}}(x^n)$. To enable sequential updating of $Q_{\mathcal{A}}$, the conditional probability $Q_{\mathcal{A}}(x_n \mid x^{n-1})$ defined by [15] as

$$Q_{\mathcal{A}}(x_n \mid x^{n-1}) \triangleq Q_{\mathcal{A}}(x^n) / Q_{\mathcal{A}}(x^{n-1}) \tag{6}$$

must be well defined. Additionally, in order to enable the use of arithmetic coding, the assigned probability must satisfy the conditions described in [23]

$$Q_{\mathcal{A}}(x_1^{n-1}) = \sum_{x_n \in \Sigma} Q_{\mathcal{A}}(x_1^n), \qquad \forall x_1^{n-1} \in \Sigma^{n-1} \tag{7}$$

where the probability of the empty string is one by convention.

The first-stage mixture, implemented to obtain the assigned probability, is performed for a given transition path. The conditional probability assignment $Q(x^N \mid \mathcal{T}')$ given any transition path $\mathcal{T}' \triangleq (t_1', \cdots, t_C')$, is recursively defined using the Krichevskiy–Trofimov (KT) empirical estimates [10], that result from mixing the parameter with a Dirichlet(0.5) prior. The KT estimates will be used with relative frequency counts that are reset at every hypothesized transition. Specifically, the conditional letter probability is defined as

$$\begin{aligned} Q(X_t = u \mid x_1^{t-1}, \mathcal{T}') \triangleq \frac{n_{t_i'}^{t-1}(u) + 1/2}{(t - t_i') + r/2}, \\ t_i' \le t < t_{i+1}', \ \forall u \in \Sigma \end{aligned} \tag{8}$$

and the probability assigned to an n-tuple is given by

$$\begin{aligned} Q(x_1^n \mid \mathcal{T}') &= \prod_{i=1}^{n} Q(x_i \mid x_1^{i-1}, \mathcal{T}') \\ &= Q(x_1^{n-1} \mid \mathcal{T}') \cdot Q(x_n \mid x_1^{n-1}, \mathcal{T}') \end{aligned} \tag{9}$$

where x_1^0 represents the null string, whose probability is one by convention.

To implement the second stage mixture, the probability assigned to an N-tuple will be a weighted sum of conditional probability assignments given transition paths. Each probability assumes a different path $\mathcal{T}'$ from a set of paths $\{\mathcal{T}\}_{\mathcal{A}}$, selected by the algorithm $\mathcal{A}$. Each path will be weighted with some *weight function* $W_{\mathcal{A}}(\mathcal{T}')$

$$Q_{\mathcal{A}}(x^N) = \sum_{\mathcal{T}' \in \{\mathcal{T}\}_{\mathcal{A}}} W_{\mathcal{A}}(\mathcal{T}') Q(x^N \mid \mathcal{T}'). \tag{10}$$

The weight function must be nonnegative for all $\mathcal{T}'$ and satisfy

$$\sum_{\mathcal{T}' \in \{\mathcal{T}\}_{\mathcal{A}}} W_{\mathcal{A}}(\mathcal{T}') = 1. \tag{11}$$

The set $\{\mathcal{T}\}_{\mathcal{A}}$ contains the only transition paths that are weighed in the second stage mixture to obtain the assigned probability of scheme $\mathcal{A}$. Paths that are not contained in this set are not weighed in the mixture. A single transition path $\hat{\mathcal{T}} \in \{\mathcal{T}\}_{\mathcal{A}}$ can be chosen from $\{\mathcal{T}\}_{\mathcal{A}}$ to estimate $\mathcal{T}$. If $\mathcal{T} \in \{\mathcal{T}\}_{\mathcal{A}}$, then we can choose $\hat{\mathcal{T}} = \mathcal{T}$. This is the case in schemes like those proposed by Willems in [20], that contain all possible paths in $\{\mathcal{T}\}_{\mathcal{A}}$, including the true path $\mathcal{T}$. However, if $\mathcal{T} \notin \{\mathcal{T}\}_{\mathcal{A}}$, a different path $\hat{\mathcal{T}} \in \{\mathcal{T}\}_{\mathcal{A}}, \hat{\mathcal{T}} \neq \mathcal{T}$ must be used to estimate $\mathcal{T}$. In order to achieve good performance (i.e., small redundancy), a coding scheme $\mathcal{A}$, that implements the two stage mixture probability assignment, must construct a proper group $\{\mathcal{T}\}_{\mathcal{A}}$ that either contains $\mathcal{T}$ or at least contains a good estimate $\hat{\mathcal{T}}$ of $\mathcal{T}$, for which $Q(x^N \mid \hat{\mathcal{T}})$ and $W_{\mathcal{A}}(\hat{\mathcal{T}})$ are large. If $\mathcal{T} \notin \mathcal{T}_{\mathcal{A}}$, the choice of $\hat{\mathcal{T}}$ defines a hypothesized PSMS $\{\hat{\mathcal{C}}, \hat{\Theta}, \hat{\mathcal{T}}\}$, which is derived from the true PSMS parameters $\{\Theta, \mathcal{T}\}$. For example, if

$$\mathcal{T} = \{\alpha N + 1\}, \qquad 0 < \alpha < 1$$

i.e., a single true transition, but the estimate $\hat{\mathcal{T}} = \phi$ (i.e., no transitions) is chosen to estimate $\mathcal{T}$, then $\hat{\Theta} = \{\hat{\theta}_0\}$, where $\hat{\theta}_0 = \alpha\theta_0 + (1 - \alpha)\theta_1$. As in the example, the hypothesized parameter set $\hat{\Theta}$ is defined by the convex combinations of the true segmental parameters θ_i along each hypothesized segment, where the weights in the combination for a hypothesized segment are the relative durations of each θ_i in this segment. The *probability of an estimated PSMS*, denoted by $Q(x^N \mid \hat{\Theta}, \hat{\mathcal{T}})$, is defined similarly as in (3) w.r.t. parameter sets $\hat{\Theta}$ and $\hat{\mathcal{T}}$ instead of Θ and $\mathcal{T}$, respectively. If $\hat{\mathcal{T}} = \mathcal{T}$, then $\hat{\Theta} = \Theta$ and $Q(x^N \mid \hat{\Theta}, \hat{\mathcal{T}}) = P(x^N \mid \Theta, \mathcal{T})$.

III. The Redundancy

The *pointwise redundancy* of scheme $\mathcal{A}$ for an N-tuple x^N, emitted by $\{\Theta, \mathcal{T}\}$, is defined as

$$R(x^N; \mathcal{A}) \triangleq R(x^N; \mathcal{A} \mid \Theta, \mathcal{T}) \triangleq \frac{1}{N} \log \frac{P(x^N \mid \Theta, \mathcal{T})}{Q_{\mathcal{A}}(x^N)} \tag{12}$$

ignoring negligible integer length constraints. For simplicity, we will omit the conditioning on the PSMS parameters. The (expected) Nth-order *redundancy* of scheme $\mathcal{A}$ is defined as

$$R_N(\mathcal{A}) \triangleq E_{\{\Theta, \mathcal{T}\}}[R(x^N; \mathcal{A})] \tag{13}$$

where $E_{\{\Theta, \mathcal{T}\}}$ denotes the expectation w.r.t. a given PSMS $\{\Theta, \mathcal{T}\}$.

The pointwise redundancy of an N-tuple for a PSMS can be expressed as

$$\begin{aligned} R(x^N; \mathcal{A}) &= \frac{1}{N} \log \frac{Q(x^N \mid \hat{\mathcal{T}})}{Q_{\mathcal{A}}(x^N)} + \frac{1}{N} \log \frac{Q(x^N \mid \hat{\Theta}, \hat{\mathcal{T}})}{Q(x^N \mid \hat{\mathcal{T}})} \\ &\quad + \frac{1}{N} \log \frac{P(x^N \mid \Theta, \mathcal{T})}{Q(x^N \mid \hat{\Theta}, \hat{\mathcal{T}})} \\ &\triangleq R_t(x^N; \mathcal{A}) + R_p(x^N; \mathcal{A}) + R_d(x^N; \mathcal{A}). \end{aligned} \tag{14}$$

Equation (14) decomposes the pointwise redundancy into three terms: R_t—*transition redundancy* (TR), R_p—*parameter redundancy* (PR), and R_d—*decision redundancy* (DR). The TR reflects universality w.r.t. the transition path. The PR is the cost of universality w.r.t. Θ for a given transition path. The DR is the additional redundancy caused by estimation error of the transition path and the segmental parameters it imposes. These terms all depend on the specific algorithm, but of course, it is the total redundancy that should be compared to the lower bound.

The PR can be upper-bounded as follows. Using the KT estimates to assign probability to a stationary segment of length m results in additional $[(r-1)/2]\log m + O(1)$ code bits for that segment [10], [17]. Therefore, using the KT estimates for each of the hypothesized segments of $\hat{\mathcal{T}}$, as done in (8) and (9) to obtain $Q(x^N \mid \hat{\mathcal{T}})$, results in

$$\begin{aligned} R_p(x^N; \mathcal{A}) &\le \frac{1}{N} \sum_{i=0}^{\hat{C}} \left[\frac{r-1}{2} \log(\hat{t}_{i+1} - \hat{t}_i) + O(1) \right] \\ &\le \frac{r-1}{2} (\hat{C}+1) \frac{\log \frac{N}{\hat{C}+1}}{N} + O\left(\frac{\hat{C}}{N} \right) \end{aligned} \tag{15}$$

where the second inequality is obtained by the Jensen inequality.

From (10) and the definition of TR in (14), we conclude that the TR depends on the weight of $\hat{\mathcal{T}}$ and can be upper-bounded by

$$\begin{aligned} R_t(x^N; \mathcal{A}) &\triangleq \frac{1}{N} \log \frac{Q(x^N \mid \hat{\mathcal{T}})}{Q_{\mathcal{A}}(x^N)} \\ &= \frac{1}{N} \log \frac{Q(x^N \mid \hat{\mathcal{T}})}{\sum_{\mathcal{T}' \in \{\mathcal{T}\}_{\mathcal{A}}} W_{\mathcal{A}}(\mathcal{T}') Q(x^N \mid \mathcal{T}')} \\ &\le -\frac{1}{N} \log W_{\mathcal{A}}(\hat{\mathcal{T}}). \end{aligned} \tag{16}$$

The inequality holds by definition of $\hat{\mathcal{T}}$.

The DR results from coding more than one true stationary segment as if it were a single stationary block. Assume the block x_{i+1}^{i+m} of length m contains data drawn by s distributions P_{u+1} to P_{u+s}, and assume this block is coded as if it were drawn by a stationary PMF Q, then Q is defined as the convex combination of the true PMF's, where each PMF P_{u+l} is weighed by its relative duration in the block. It is easy to show that the contribution to the DR of this block is upper-bounded by the entropy of the relative durations vector multiplied by m and normalized by the complete sequence length N. Since the vector consists of s components, we can bound the contribution of this block to the DR by

$$R_d(x_{i+1}^{i+m}; \mathcal{A}) \le \frac{m \log s}{N}. \tag{17}$$

The DR of an N-tuple is the sum of the contributions of all segments hypothesized by $\hat{\mathcal{T}}$.

IV. An Optimal Linear Per-Letter Complexity Scheme

The scheme presented in this section uses Willems' linear weighting scheme [20], [21] to group transition paths into states in a diagram, but with different weight functions, corresponding to the weights used by Merhav in [12]. We will denote a general linear weighting scheme by $\mathcal{L}$, and propose two different optimal versions of the new scheme, one that performs better when the number of transitions C is small, denoted by $\mathcal{L}_1$, and the other, denoted by $\mathcal{L}_2$, performs better when C grows with N. Willems' original linear scheme will be denoted by $\mathcal{W}$. Both versions of the new scheme will be shown to achieve the lower bound on the redundancy of (1), as opposed to Willems' scheme.

The idea of Willems' linear scheme is to implement the mixture method using a *linear transition diagram* that contains all possible transition paths, as illustrated in Fig. 1. This diagram reduces the exponential complexity and still enables weighting of all transition paths. A directed path along the diagram represents a transition path. Horizontal move denotes that the source remains in the same stationary segment. An upward move in the graph represents a transition of the source. A box in the diagram represents a state. State s_n at time n is defined as the time instant of the most recent transition within the period $1 \le t \le n$. In order to implement the weighting procedure, each state is assigned a weight $G(s_n, x_1^n)$ associated with the subsequence x_1^n. The weight $G(s_n, x_1^n)$ is defined as the joint probability assigned to the sequence x_1^n along with the event that the last transition before time n

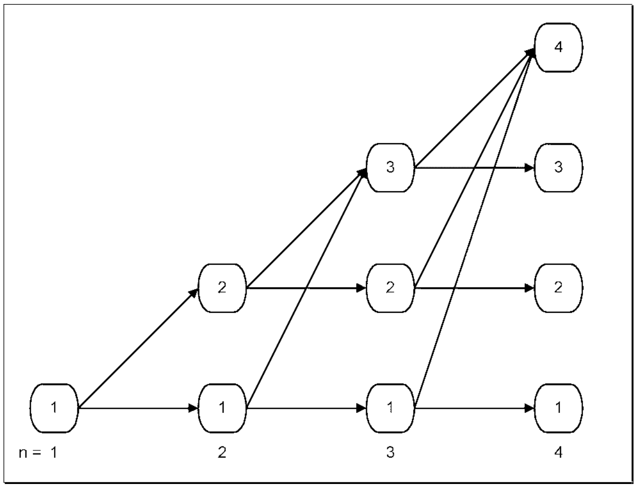

Fig. 1. Linear transition diagram at time instants 1 to 4. A number in a state box denotes the most recent transition point of the state represented by the box. The time is denoted below the graph.

occurred at $t = s_n$. The probability assigned to x_1^n is the sum of the weights of all states in the diagram at time n

$$Q_{\mathcal{L}}\big(x_1^n\big) \triangleq \sum_{s_n=1}^{n} G\big(s_n, x_1^n\big). \tag{18}$$

We note that by definition of a state the diagram will always consist of n states at time n, that form a partition of all transition paths into n disjoint sets.

The weight of a state is recursively defined by the KT estimates and the transition rules of the diagram. The KT probability of letter x_n at state s_n is obtained, similarly as in (8), by

$$Q(x_n \mid s_n) \triangleq Q\big(x_n \,\big|\, s_n, x_1^{n-1}\big) \triangleq \frac{n_{s_n}^{n-1}(x_n) + 1/2}{(n - s_n) + r/2}. \tag{19}$$

The only two possible transitions from state s_n at time n to state s_{n+1} at time $n+1$ are the self-transition, i.e., $s_{n+1} = s_n$, and the transition to the *only* new state formed at time $n+1$ that assumes a source transition between time n and time $n+1$, i.e., $s_{n+1} = n+1$. Associated with each such transition, there is a weight $W_{\text{tr}}(s_{n+1} \mid s_n)$, where

$$W_{\text{tr}}(s_{n+1} = s_n \mid s_n) + W_{\text{tr}}(s_{n+1} = n+1 \mid s_n) = 1. \tag{20}$$

The weight of a state is, therefore, recursively updated as in (21) at the bottom of this page. It is easy to see that the weight assigned to a state by this procedure is a weighted sum of the KT estimates assigned to all transition paths $\mathcal{T}_n'$ of order n that lead to this state

$$G\big(s_n, x_1^n\big) = \sum_{\mathcal{T}_n' \to s_n} W(\mathcal{T}_n')Q\big(x_1^n \,\big|\, \mathcal{T}_n'\big) \tag{22}$$

where $\mathcal{T}_n' \to s_n$ is the set of all paths leading to s_n. The weight $W(\mathcal{T}_n')$ is the product of all the transition weights along the path representing $\mathcal{T}_n'$ in the diagram.

So far, we have described a general linear scheme as presented in [20] and [21]. However, we have not defined the actual state transition weights. The transition weights defined by Willems use the binary KT estimates of the distance from the last transition

$$W_{\text{tr}}^{\mathcal{W}}(s_{n+1} \mid s_n) \triangleq \begin{cases} \frac{1/2}{(n-s_n)+1}, & s_{n+1} = n+1 \\ \frac{(n-s_n)+1/2}{(n-s_n)+1}, & s_{n+1} = s_n. \end{cases} \tag{23}$$

The proposed scheme defines state transition weights $W_{\text{tr}}^{\mathcal{L}}(s_{n+1} \mid s_n)$, that are different from those defined above. The concept used for the weights is the same for both versions of the proposed scheme, although the transition weights themselves are defined differently for $\mathcal{L}_1$ and for $\mathcal{L}_2$. The weights for both versions are defined as follows. For a given $\varepsilon > 0$, let

$$\pi(j) \triangleq \frac{1}{j^{1+\varepsilon}}, \qquad 1 \le j < N \tag{24}$$

$$Z_n \triangleq \sum_{j=1}^{n} \pi(j) \tag{25}$$

and

$$Z_\infty = \sum_{j=1}^{\infty} \pi(j). \tag{26}$$

Now

$$W_{\text{tr}}^{\mathcal{L}_1}(s_{n+1} \mid s_n) \triangleq \begin{cases} \frac{\pi(n)}{Z_\infty - Z_{n-1}}, & s_{n+1} = n+1 \\ \frac{Z_\infty - Z_n}{Z_\infty - Z_{n-1}}, & s_{n+1} = s_n \end{cases} \tag{27}$$

and

$$W_{\text{tr}}^{\mathcal{L}_2}(s_{n+1} \mid s_n) \triangleq \begin{cases} \frac{\pi(n-s_n+1)}{Z_\infty - Z_{n-s_n}}, & s_{n+1} = n+1 \\ \frac{Z_\infty - Z_{n-s_n+1}}{Z_\infty - Z_{n-s_n}}, & s_{n+1} = s_n. \end{cases} \tag{28}$$

By both weight assignments proposed in (24)–(28), we assign to each time point n, $1 \le n < N$, a distribution over the discrete time t, $t > n$, for the probability that the next transition occurs just before time *t*. In both cases, the assigned probability of source transition at time $n+1$ is the weight $W_{\text{tr}}^{\mathcal{L}}(n+1 \mid s_n)$. For $\mathcal{L}_1$ for instance, this probability is $\pi(n)$ normalized by the infinite sum of $\pi(t)$, $t \ge n$, which is equal $Z_\infty - Z_{n-1}$. The probability assigned at time n to a transition to occur at $n+2$ is $\pi(n+1)$, normalized by the same factor, and so on, for all $t > n+2$. Hence, the probability of no transition to occur at $t = n+1$, which is assigned to the self-transition weight in (27), is the partial sum of the probabilities assigned at n for transitions to occur at all time units larger than $n+1$.

The weights of $\mathcal{L}_1$ in (27) depend on the absolute time, instead of the relative time from the last transition as in (23) and (28). This reduces the weight of a transition, thus weakening weights of transition paths with many transitions,

$$G\big(s_n, x_1^n\big) = Q(x_n \mid s_n) \cdot \begin{cases} W_{\text{tr}}(s_n = s_{n-1} \mid s_{n-1}) \cdot G\big(s_{n-1}, x_1^{n-1}\big), & s_n < n \\ \sum\limits_{j=1}^{n-1} W_{\text{tr}}(s_n = n \mid s_{n-1} = j) G\big(s_{n-1} = j, x_1^{n-1}\big), & s_n = n. \end{cases} \tag{21}$$

but strengthening weights of transition paths with a few transitions. Therefore, the use of $\mathcal{L}_1$ is justified if we assume that the number of transitions is small. If we assume that C grows with N, better performance can be achieved by $\mathcal{L}_2$. The use of absolute weights also leads to a computational improvement. First, the self-transition weight can be computed *once* using (27) for all self-transitions in (21). Furthermore, (21) for $s_n = n$ reduces, using (18), to

$$\begin{aligned} &G_{\mathcal{L}_1}\left(s_n = n, x_1^n\right) \\ &\quad = Q(x_n \mid s_n) W_{\rm tr}^{\mathcal{L}_1}(s_n = n \mid s_{n-1} < n) Q_{\mathcal{L}_1}\left(x_1^{n-1}\right) \end{aligned} \tag{29}$$

where $W_{\rm tr}^{\mathcal{L}_1}(s_n = n \mid s_{n-1} < n)$ is the same for all $s_{n-1} < n$.

The weights proposed for $\mathcal{L}_1$ and for $\mathcal{L}_2$ appear to be more computationally demanding than Willems' weights. However, it is straightforward to see that the transition weights to the new state $s_{n+1} = n+1$ of $\mathcal{L}_1$ have similar behavior to ε/n, and those of $\mathcal{L}_2$ behave similarly as $\varepsilon/(n-s_n+1)$. Hence, although the analysis and understanding of the scheme are simpler with the original definitions, the weights of both versions of the scheme can be replaced by the simpler weights.

The TR results from two factors: the transition weights at transition points, and the cumulative weight assigned to all self-transitions along the true path $\mathcal{T}$. The weights of transitions to new states must decay as $O\left(1/n\right)$ as in $\mathcal{L}_1$, or as $O\left(1/(n-s_n)\right)$ as in $\mathcal{L}_2$ and $\mathcal{W}$. This results in additional TR of $(\log t_i)/N$ in the first case and of $[\log\left(t_i - t_{i-1}\right)]/N$ in the second for transition t_i. However, additional TR is obtained from the self-transitions. Therefore, they must be designed large enough to ensure that the cumulative weight assigned to all self-transitions results in still negligible contribution to the TR. This is not the case for the weights of $\mathcal{W}$ defined in (23), that are "generous" to new transitions at the expense of the self-transitions, and therefore do not achieve the PSMS bound for $C < O\left(N\right)$. The pointwise redundancy of Willems' weights is bounded by

$$R(x^N; \mathcal{W}) \leq \left[\frac{r-1}{2}(C+1) + \frac{3C+1}{2}\right] \frac{\log(N/C)}{N} + O\left(\frac{C}{N}\right). \tag{30}$$

The bound of (30) is obtained from the analysis in [20]. The quadratical per-letter complexity scheme, proposed in [20], does not achieve the bound either. However, both versions of the new scheme, on the other hand, achieve the lower bound. This is stated in the following theorem.

Theorem 1: The redundancies of both versions of the linear weighting scheme with state transition weights as in (27) and (28), are upper-bounded by

$$\begin{aligned} R(x^N; \mathcal{L}_1) &\leq \left[\frac{r-1}{2}(C+1) + C + \varepsilon\right] \frac{\log N}{N} + O\left(\frac{C}{N}\right) \\ R(x^N; \mathcal{L}_2) &\leq \left[\frac{r-1}{2}(C+1) + C + (C+1)\varepsilon\right] \\ &\qquad \cdot \frac{\log\left(N/C\right)}{N} + O\left(\frac{C}{N}\right) \end{aligned} \tag{31}$$

respectively, for every N-tuple drawn by any PSMS with C transitions, for all $\varepsilon > 0$.

If we let ε decay *slowly* with time, such that

$$\varepsilon_j \triangleq \frac{\log\left(\log\left(2j\right)\right)^k}{\log j}, \qquad k > 1$$

and

$$\pi(j) \triangleq j^{-(1+\varepsilon_j)} = \frac{1}{j(\log\left(2j\right))^k} \tag{32}$$

it can be shown that

$$\begin{aligned} R(x^N; \mathcal{L}_1) &\leq \left[\frac{r-1}{2}(C+1) + C\right] \frac{\log N}{N} \\ &\qquad + O\left(\frac{C \log\log N}{N}\right) \\ R(x^N; \mathcal{L}_2) &\leq \left[\frac{r-1}{2}(C+1) + C\right] \frac{\log(N/C)}{N} \\ &\qquad + O\left(\frac{C \log\log(N/C)}{N}\right) \end{aligned} \tag{33}$$

and thus the lower bound is asymptotically achieved by both versions of the scheme.

Theorem 1 makes no prior assumptions on the number of transitions C. However, it derives the expected two conclusions: If $C = O(1)$, the weights of $\mathcal{L}_1$ are optimal since they are the least "generous" to new transitions. On the other hand, if C is expected to be larger than $O(1)$, the weights of $\mathcal{L}_2$ that are more "generous" to new transitions should be used. We also note that the weights of (23) are merely a special case of scheme $\mathcal{L}_2$ with $\varepsilon = 0.5$. We conclude this section with the proof of Theorem 1. When we derive an upper bound on the TR of $\mathcal{L}_2$, we will demonstrate where the weights of (23) fail to achieve the bound.

Proof of Theorem 1: It is straightforward that this coding scheme satisfies (10) (see, e.g., [20]), and weighs probabilities assigned given *all* possible transition paths. Therefore, the probability assigned to the true path $\mathcal{T}$ is always contained in the mixture, and so we can choose $\hat{\mathcal{T}} = \mathcal{T}$, resulting in $R_d = 0$, and $\hat{C} = C$. The PR is upper-bounded by (15), with $\hat{C} = C$, and thus attains the first term of the lower bound, as expressed by the first term of the dominant expression of (31). Note that to obtain the first term of the bound on the redundancy we use the relations $\log\left(N/(C+1)\right) \leq \log\left(N/C\right) \leq \log N$. The first inequality is used for $\mathcal{L}_2$, and both are used for $\mathcal{L}_1$. It is now sufficient to show that the second term of the lower bound is attained by the TR as well for both versions of the scheme. To do so, we will show that

$$\begin{aligned} R_t(x^N; \mathcal{L}_1) &\leq \frac{1}{N}[(C+\varepsilon)\log N + \log(1+\varepsilon) - C\log\varepsilon] \\ R_t(x^N; \mathcal{L}_2) &\leq \frac{1}{N}\left[C \log\frac{N}{C} + (C+1)\varepsilon\log\frac{N}{C+1}\right. \\ &\qquad \left. + \, (C+1)\log\frac{1+\varepsilon}{\varepsilon} + \log\varepsilon\right]. \end{aligned} \tag{34}$$

We begin with the TR of $\mathcal{L}_1$. We first use (27) to express $W_{\mathcal{L}_1}(\mathcal{T})$, next we identify which factors result from self-transitions and which from transitions to new states, and then upper- and lower-bound the partial sum $Z_\infty - Z_n$, $\forall n: 0 \leq n < N$, by approximating the sum by an integral. Finally, we use these bounds on the cumulative weight function $W_{\mathcal{L}_1}(\mathcal{T})$

to upper-bound the TR by $-(1/N)\log W_{\mathcal{L}_1}(\mathcal{T})$ as in (16). The same procedure is then used to bound $R_t(x^N;\mathcal{L}_2)$.

The weight of $\mathcal{T}$ is obtained by

$$\begin{aligned} W_{\mathcal{L}_1}(\mathcal{T}) &= \prod_{i=0}^{C}\Bigg[W_{\rm tr}\big(s_{t_i}=t_i \,\big|\, s_{t_i-1}=t_{i-1}\big) \\ &\qquad \cdot \prod_{j=t_i+1}^{t_{i+1}-1} W_{\rm tr}(s_j=t_i \mid s_{j-1}=t_i)\Bigg] \\ &= \left[\prod_{n=1}^{N-1} W_{\rm tr}(s_{n+1}=s_n \mid s_n)\right] \\ &\qquad \cdot \left[\prod_{i=1}^{C} \frac{W_{\rm tr}\big(s_{t_i}=t_i \,\big|\, t_{i-1}\big)}{W_{\rm tr}\big(s_{t_i}=t_{i-1} \,\big|\, t_{i-1}\big)}\right] \\ &= \left[\prod_{n=1}^{N-1}\frac{Z_\infty - Z_n}{Z_\infty - Z_{n-1}}\right]\cdot\left[\prod_{i=1}^{C}\frac{\pi(t_i-1)}{Z_\infty - Z_{t_i-1}}\right] \\ &= \frac{Z_\infty - Z_{N-1}}{Z_\infty}\cdot\prod_{i=1}^{C}\frac{\pi(t_i-1)}{Z_\infty - Z_{t_i-1}}. \end{aligned} \tag{35}$$

We define $W_{\rm tr}(s_1=1 \mid s_0=t_{-1}) \triangleq 1$. The first equality is obtained by taking the transition weights along $\mathcal{T}$. The second equality is obtained by multiplication and division by the self-transition weights at points of true source transitions. We use the telescopic property of the first product to obtain the last equality. The first term represents the TR of the self-transitions and is of $O(N^{-\varepsilon})$, while the second product term, which can be lower-bounded by $O(N^{-C})$, represents the TR of the C true transitions.

Approximating the infinite sum by an integral, we bound the partial sum $Z_\infty - Z_n$ by

$$\frac{1}{\varepsilon(n+1)^{\varepsilon}} \le Z_\infty - Z_n \le \frac{n+1+\varepsilon}{\varepsilon(n+1)^{1+\varepsilon}}, \qquad \forall n\ge 0. \tag{36}$$

We note that $Z_\infty = Z_\infty - Z_0$, and thus $1/\varepsilon \le Z_\infty \le (1+\varepsilon)/\varepsilon$. Finally, we use the last bounds to upper-bound the TR by the upper bound of (16). We bound each term separately, and then sum all the terms to obtain the TR upper bound. The contribution of self-transitions is bounded by

$$\begin{aligned} -\log\frac{Z_\infty - Z_{N-1}}{Z_\infty} &\le \log(\varepsilon N^{\varepsilon}) + \log\frac{1+\varepsilon}{\varepsilon} \\ &= \varepsilon\log N + \log(1+\varepsilon). \end{aligned} \tag{37}$$

The contribution of a single transition is bounded by

$$\begin{aligned} -\log\frac{\pi(t_i-1)}{Z_\infty - Z_{t_i-1}} &\le \log\frac{(t_i+\varepsilon)(t_i-1)^{1+\varepsilon}}{\varepsilon t_i^{1+\varepsilon}} \\ &\le \log N - \log\varepsilon. \end{aligned} \tag{38}$$

The second inequality is obtained by taking N as an upper bound on $t_i+\varepsilon$. Summing up the last two inequalities, we conclude that

$$\begin{aligned} R_t(x^N;\mathcal{L}_1) &\le -\frac{1}{N}\log W_{\mathcal{L}_1}(\mathcal{T}) \\ &\le \frac{1}{N}[(C+\varepsilon)\log N + \log(1+\varepsilon) - C\log\varepsilon] \end{aligned} \tag{39}$$

proving the first inequality of (34).

To bound the TR of $\mathcal{L}_2$, we must represent $W_{\mathcal{L}_2}(\mathcal{T})$ a little differently from the representation of $W_{\mathcal{L}_1}(\mathcal{T})$ in (35). Let us define $W_{\rm tr}(t_{C+1}|s_N=t_C) \triangleq 1$, (where we recall that $t_{C+1} \triangleq N+1$), then

$$\begin{aligned} W_{\mathcal{L}_2}(\mathcal{T}) &= \prod_{i=0}^{C}\left[\prod_{n=t_i}^{t_{i+1}-2} W_{\rm tr}(s_{n+1}=t_i \mid s_n=t_i)\right] \\ &\qquad \cdot W_{\rm tr}\big(s_{t_{i+1}}=t_{i+1} \mid s_{t_{i+1}-1}=t_i\big) \\ &= \left[\prod_{i=0}^{C-1}\frac{Z_\infty - Z_{t_{i+1}-t_i-1}}{Z_\infty}\cdot\frac{\pi(t_{i+1}-t_i)}{Z_\infty - Z_{t_{i+1}-t_i-1}}\right] \\ &\qquad \cdot\frac{Z_\infty - Z_{N-t_C}}{Z_\infty} \\ &= \left[\prod_{i=0}^{C-1}\frac{\pi(t_{i+1}-t_i)}{Z_\infty}\right]\cdot\frac{Z_\infty - Z_{N-t_C}}{Z_\infty}. \end{aligned} \tag{40}$$

The second equality is obtained by substituting the transition weights, and by the telescopic product of self-transitions inside a stationary segment. The inner product in the first line of (40) is the contribution of self-transitions, while the outer term constitutes the contribution of true transition points to the TR. It is easy to show, using the Stirling formula, that if we assign $W_{\rm tr}^{\mathcal{W}}$ in (40) instead of $W_{\rm tr}^{\mathcal{L}_2}$, the contribution to the TR of self-transitions in segment i will be of $0.5\log(t_{i+1}-t_i)/N$, which is not negligible, while each true transition will still result in TR of $\log(t_i - t_{i-1})/N$. This is the reason that the weights used in $\mathcal{W}$ do not achieve the lower bound.

The last equality of (40) consists of $C+1$ terms, each representing the contribution of a segment, which consists of the self-transitions in the segment and the true transition out of the segment to another segment. This, of course, excludes the last segment, for which only self-transitions occur. The contributions to the TR of each of the two general terms in (40) are upper-bounded by

$$-\log\frac{\pi(t_{i+1}-t_i)}{Z_\infty} \le (1+\varepsilon)\log(t_{i+1}-t_i) + \log\frac{1+\varepsilon}{\varepsilon} \tag{41}$$

$$-\log\frac{Z_\infty - Z_{N-t_C}}{Z_\infty} \le \varepsilon\log(N+1-t_C) + \log\varepsilon + \log\frac{1+\varepsilon}{\varepsilon}. \tag{42}$$

Summing up all the terms, normalizing by N, and using the Jensen inequality w.r.t. the segment lengths in the logarithmic expressions, we conclude the proof of (34), and the proof of Theorem 1.

V. A Decision Weighting Scheme

In this section we show that there exists a fixed complexity scheme, based on the transition diagram of the linear per-letter complexity scheme, that achieves vanishing redundancy. The redundancy is of the order of the lower bound when the transitions are large. We refer to the new scheme as the *decision weighting scheme* (DW), and denote it by $\mathcal{D}$. This scheme uses a *data-dependent reduced-state transition diagram*. It eliminates transition paths with low likelihood, and it does not create a new state every time instant.

The scheme produces new states every $k \ge 1$ time instants instead of every instant, in order to reduce the diagram's

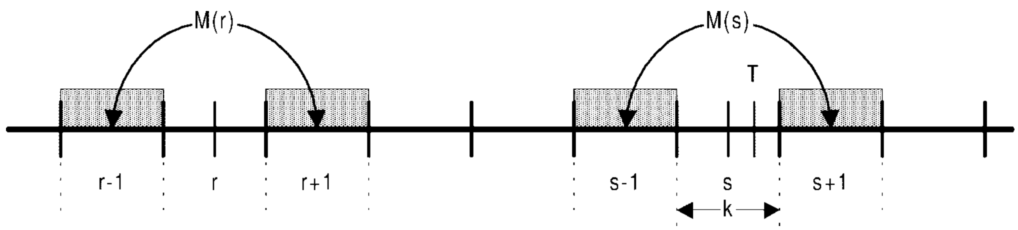

Fig. 2. Block partitioning for DW. The solid line represents the data sequence that is divided into blocks of length k. True transition occurs at T and can be only estimated at the block midpoint by state s with likelihood $M(s)$, obtained by the empirical data of the two neighboring blocks $s-1$ and $s+1$. Another estimated transition is within block r, estimated at its midpoint, with likelihood $M(r)$.

growth rate. This forms a partition of the data into N/k nonoverlapping blocks of length k, and for each block only one state is created. The parameter k is a design parameter that will be referred to as the *block length*. In order to keep the number of *surviving* states (and the computational complexity) fixed, we assign to each state s a *metric* $M(s)$ that determines the likelihood of a transition within the block represented by s. States with low metric values are eliminated. The number of surviving states with high metrics S is the second design parameter of the algorithm. By definition of the transition diagram, the set of surviving states defines a set of surviving transition paths, and a transition path that leads to an eliminated state is said to be eliminated and not to exist in the diagram.

The state number s represents the block number in which the most recent transition is assumed to have occurred. A state s, $s>1$, that estimates transition at any time within the block it represents, is created at the block midpoint. The first state $s=1$ is naturally created at the first time instant.

The metric of a state s is defined as follows. Let $n_s(u)$ be the number of occurrences of the letter u in block s. The empirical per-letter entropy of the block is given by

$$H(s) = -\sum_{u\in\Sigma} \frac{n_s(u)}{k}\log\frac{n_s(u)}{k}. \qquad (43)$$

The empirical entropy of the concatenation of blocks r and s is given by

$$H(r,s) = -\sum_{u\in\Sigma} \frac{n_r(u)+n_s(u)}{2k}\log\frac{n_r(u)+n_s(u)}{2k}. \qquad (44)$$

Now, $M(s)$ is defined by

$$M(1) \triangleq \infty \qquad (45)$$

$$M(s) \triangleq H(s-1,s+1) - 0.5H(s-1) - 0.5H(s+1), \quad \forall s>1. \qquad (46)$$

The quantity $M(s)$ measures the "distance" between the empirical distributions of blocks $s-1$ and $s+1$. If $M(s)$ is large then it is likely that a change has occurred between the two blocks inside block s. It is well known that $M(s)$ serves as asymptotically optimal statistics for testing whether or not two sequences emerged from the same source [6], [26]. Hence, a state s that is created at the midpoint of a block s with large $M(s)$ is likely to represent transition in a surviving transition path in the diagram. Fig. 2 illustrates the partitioning mechanism. The metric $M(s)$ is nonnegative for all $s>1$, even if transition has not occurred. This can cause elimination of $s=1$ if its metric had been defined smaller, even when other transitions have not occurred. Since state $s=1$ always represents a transition, we thus define its metric to be infinite.

The probability assignment scheme can be described by the state diagram shown in Fig. 3 for $k=4$ and $S=3$. The diagram begins when all S large metric states already exist, i.e., in steady state. The boxes in the diagram denote the states, and the numbers in the boxes the block numbers of the most recent transitions assumed by the states. As in the linear scheme, each state is assigned a weight $G(s_n, x_1^n)$ associated with the subsequence x_1^n. The weight of a state is recursively defined by the KT estimates and the transition rules shown in the diagram. The KT probability of letter x_n at state s_n is obtained by

$$Q(x_n \mid s_n) \triangleq Q\big(x_n \mid s_n, x_1^{n-1}\big) \triangleq \frac{n_{\tau_n}^{n-1}(x_n)+1/2}{(n-\tau_n)+r/2} \qquad (47)$$

where the time τ_n is the first time instant after the last transition assumed by s_n, which is defined by

$$\tau_n \triangleq \begin{cases} 1, & s_n = 1\\ \lfloor (s_n-0.5)k \rfloor + 1, & s_n > 1. \end{cases} \qquad (48)$$

The transition rules and update procedures at each time point are defined for three different cases as follows:

1) At a block midpoint $n = \lfloor (m-0.5)k \rfloor + 1; m > 1$ fixed, a new state m is created. The state weights are recursively updated almost as in the linear scheme by (49) at the bottom of this page, where the transition weights are given by

$$W_{\mathrm{tr}}(s_{n+1}|s_n=j) \triangleq \begin{cases} \frac{\pi(m-j)}{Z_\infty - Z_{m-j-1}}, & s_{n+1}=m\\ \frac{Z_\infty - Z_{m-j}}{Z_\infty - Z_{m-j-1}}, & s_{n+1}=j. \end{cases} \qquad (50)$$

$$G\big(s_n, x_1^n\big) = Q(x_n \mid s_n)\cdot \begin{cases} W_{\mathrm{tr}}(s_n = s_{n-1} \mid s_{n-1})\cdot G\big(s_{n-1}, x_1^{n-1}\big), & s_n < m\\ \sum_{j=1}^{m-1} W_{\mathrm{tr}}(s_n = m \mid j) G\big(s_{n-1}=j, x_1^{n-1}\big), & s_n = m. \end{cases} \qquad (49)$$

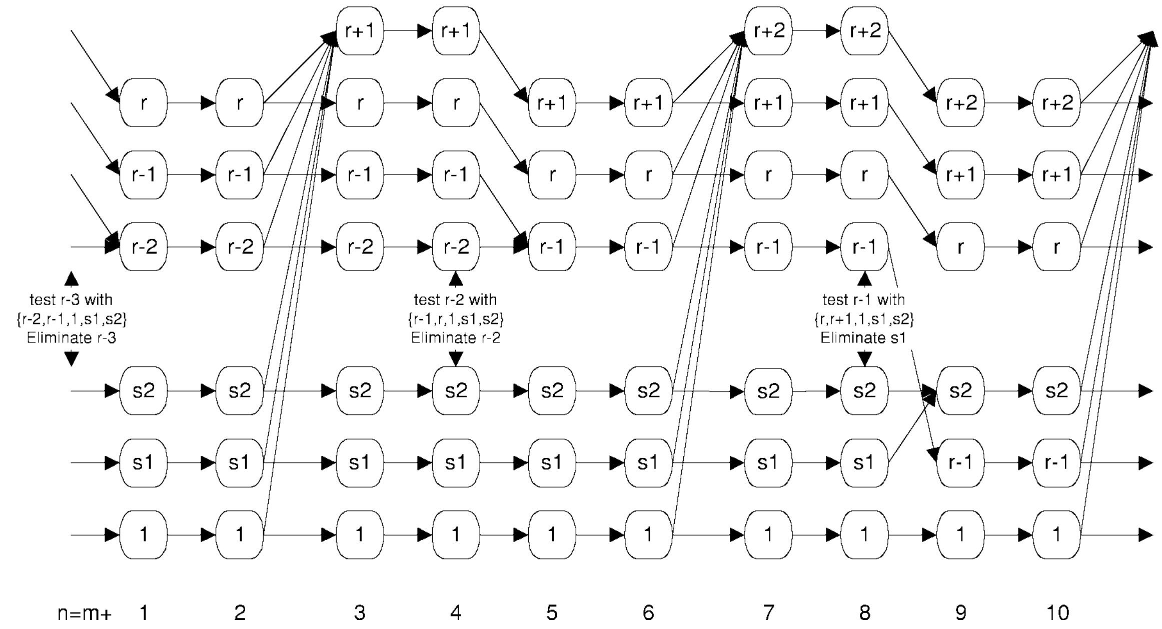

Fig. 3. Example of DW transition diagram in steady state for $k = 4$ and $S = 3$. The diagram starts right after time instant m at the first point of the $r + 1$ block. A single new state is created at each block midpoint, and a single low metric state is eliminated at each block partitioning point. Time instants are denoted below the diagram.

(The weight of a previously eliminated state j is zero, $G(j, x_1^{n-1}) = 0$). The transition weights depend on the relative block number from the last transition as in $\mathcal{L}_2$, and not on the absolute block number, as in $\mathcal{L}_1$. The proof of Theorem 2 will be based on this fact.

2) At the first point of a new block $n = mk + 1$; $m \geq 5$ fixed, at most a single state i is eliminated into another state j. The weight of i is added into the weight of j, which is the smallest state in the diagram that is still larger than i. Only the self-transitions are performed from all other states (see (51) at the bottom of this page).

3) At any other point n there are only self-transitions.

$$G(s_n, x_1^n) = Q(x_n \mid s_n) \cdot G(s_n = s_{n-1}, x_1^{n-1}). \quad (52)$$

The elimination retains a fixed number of states in the diagram. It also ensures that no more than one state of any three consecutive states remain in the diagram. This is done to avoid a situation where a single transition is represented by two or three states. If a transition occurs at the midpoint of block s, all three states $s - 1$, s, and $s + 1$ may have large metrics, but we only need to save one of them to represent the transition. We will therefore eliminate the states with the lower metric values among the three. The computation of the metric of state s requires delay of $1.5k$ time points, to obtain the empirical data of block $s + 1$. An additional delay of $2k$ points is required for computation of the metrics of $s + 1$ and $s + 2$ that are tested against s. Hence, every new state will exist at least $3.5k$ time points before it is tested for the first time. Due to the delay of $3.5k$ time points between creation and the first possible elimination of a state, the steady-state diagram contains $S + 3$ states at time points of first halves of partitioning blocks, and $S + 4$ states at time points of second halves.

The elimination procedure at $n = mk + 1$ takes three stages of testing the metric of the state s, created at $n - 3.5k$. If $s - 1$ or $s - 2$ exist in the diagram, state s is eliminated, since $M(s - 1) \geq M(s)$ or $M(s - 2) \geq M(s)$, respectively. Otherwise, state s is tested against states $s + 1$ and $s + 2$, and if $M(s + 1) > M(s)$ or $M(s + 2) > M(s)$, s is eliminated. If s passed both tests and there are less than S states, created before time $n - 3.5k$, no elimination is performed. If there are S such states, the state with the lowest metric among the existing states, created at time $n - 3.5k$ or earlier, is eliminated. A state is always eliminated by adding its weight into the weight of the closest newer state. This strategy minimizes the DR in case a true transition is eliminated by replacing it by the closest hypothesized transition point, still existing in the diagram.

The probability assigned to the subsequence x^n is obtained, as in the linear scheme, by the sum of the weights of all states that exist in the diagram at time instant n

$$Q_{\mathcal{D}}(x_1^n) = \sum_{s_n} G(s_n, x_1^n) \quad (53)$$

where the notation $\sum_{s_n}$ represents a sum over all states existing in the diagram at time instant n. In contrast to the linear scheme, this strategy does not satisfy the general mixture structure presented in (10), but can be easily shown to yield a valid probability function that satisfies both (6) and (7).

$$G(s_n, x_1^n) = Q(x_n \mid s_n) \cdot \begin{cases} 0, & s_n = i \\ G(i, x_1^{n-1}) + G(j, x_1^{n-1}), & s_n = j \\ G(s_n, x_1^{n-1}), & \text{otherwise.} \end{cases} \quad (51)$$

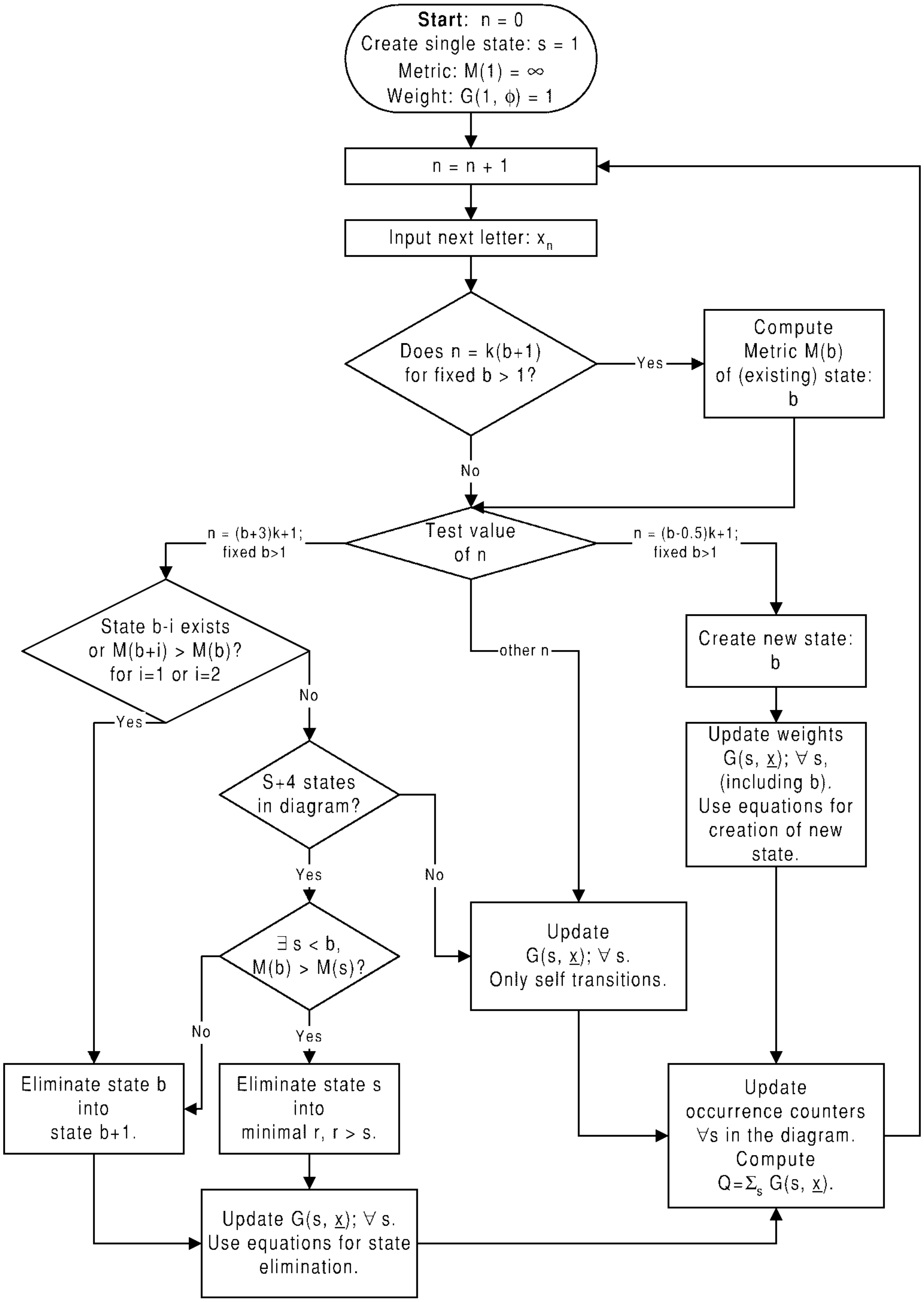
Start: n = 0
Create single state: s = 1
Metric: M(1) = ∞
Weight: G(1, ϕ) = 1
n = n + 1
Input next letter: x_n
Does n = k(b+1) for fixed b > 1?
Yes
Compute Metric M(b) of (existing) state: b
No
Test value of n
n = (b+3)k+1; fixed b>1
n = (b-0.5)k+1; fixed b>1
other n
State b-i exists or M(b+i) > M(b)? for i=1 or i=2
Yes
No
S+4 states in diagram?
Yes
No
∃ s < b, M(b) > M(s)?
No
Yes
Create new state: b
Update weights G(s, x); ∀ s, (including b). Use equations for creation of new state.
Update G(s, x); ∀ s. Only self transitions.
Eliminate state b into state b+1.
Eliminate state s into minimal r, r > s.
Update occurrence counters ∀s in the diagram. Compute Q=Σ_s G(s, x).
Update G(s, x); ∀ s. Use equations for state elimination.

Fig. 4. Flow diagram of the DW. Metrics of new states are computed at block endpoints. The diagram splits into three different cases for updating the state weights. At block midpoints, new states are created. At block partitioning points, states are eliminated using the elimination criteria. At any other point, no transitions occur between different states.

The update procedure of the transition diagram is fully described in a flowchart in Fig. 4. The per-letter computational complexity of this scheme is $O(S)$. Since we will assume a fixed S, $O(S) = O(1)$. Every state stores occurrence counts of $O(N)$, therefore, the storage complexity of the scheme is $O(\log N)$.

We conclude this section with two theorems that upper-bound the pointwise and average redundancies of the DW by expressions that vanish as k and N increase (as long as k is of smaller order). Both theorems show that the redundancy decays to zero, and specifically at the rate of the lower bound for large transitions. The next theorem upper-bounds the pointwise redundancy.

Theorem 2: The pointwise redundancy of the DW is bounded uniformly for all PSMS's by

$$R(x^N;\mathcal{D}) \le \frac{r-1}{6}\frac{\log k}{k} + O(\alpha(C,N,k)) \qquad (54)$$

for every x^N and any number of transitions C, where $\alpha(C, N, k)$ is defined by

$$\alpha(C, N, k) \triangleq \max\left\{\frac{1}{k}, \frac{Ck}{N}\right\}. \tag{55}$$

Theorem 2 makes no prior assumptions on the number of transitions C that does not have to be bounded. (Of course, if $C > o(N/k)$, then $\alpha(C, N, k)$ does not vanish, and the bound is no longer useful). The proof of Theorem 2 is presented in Appendix A. It is based on the choice of the transition weights in (50), that depend on the relative block number from the last transition and not on the absolute time or the absolute block number.

The DW scheme performs decisions. Obviously, there is a tradeoff consideration associated with the choice of the parameter k. A larger k provides a more reliable metric, leading to smaller probability of eliminating the best $\hat{\mathcal{T}}$. On the other hand, a larger k increases the DR caused by estimation of transitions at block midpoints. An upper bound on the average Nth-order redundancy as a function of k can be obtained based on the analysis in Appendix B. By differentiating this bound w.r.t. k, it can be shown that the optimal choice of k is of the form

$$k = A \log N + O(\log \log N) \tag{56}$$

where the parameter A depends on the parameters of the PSMS. Since we desire a universal scheme, we will define the block length as

$$k = A \log N \tag{57}$$

where the parameter A will be a design parameter of the DW scheme. By substituting the block length of (57) in Theorem 2, we conclude that the pointwise redundancy for this choice of k is upper-bounded by

$$R(x^N; \mathcal{D}) \leq \frac{r-1}{6A} \frac{\log \log N}{\log N} + O\left(\max\left\{\frac{1}{\log N}, \frac{C \log N}{N}\right\}\right). \tag{58}$$

In most practical applications, a typical choice of the design parameter k satisifies (57) w.r.t. the actual sequence length N, and determines the coefficient A. However, we want our scheme to be a strongly sequential one. Therefore, we cannot assume that N is known in advance, and use this knowledge to determine k as in (57). The solution in this case is to assume an initial horizon N_0, and use it to determine an initial block length k_0. If time unit $n = N_0$ is reached, the horizon can be updated to $N_1 = N_0^{1+\omega}$, $\omega > 0$, and the block length is updated accordingly by $k_1 = (1+\omega)k_0$. All surviving states can be reset, excluding the state created at the hypothesized horizon change point, and the algorithm starts over with the new parameter k_1. This process can go on at any time n, in which the most recently hypothesized horizon has been reached. It is possible to show that the upper bound of Theorem 2 remains of the same order even when this strongly sequential version of the algorithm is applied.

We now present the main theorem of the DW scheme. We begin with two definitions that characterize a PSMS. The *error exponent* of a PSMS, $E(\Theta)$, is defined as

$$E(\Theta) \triangleq \min_{0 \leq i \leq C-1} \left\{ -2 \log \frac{\sum_{u \in \Sigma} \sqrt{P_i(u) P_{i+1}(u)} + 1}{2} \right\}. \tag{59}$$

The error exponent expresses the "size" of the "minimal" transition between adjacent segments of a PSMS. It can easily be shown that $0 \leq E(\Theta) \leq 2$. The larger is $E(\Theta)$, the larger is the "minimal" transition of the PSMS.

The *divergence* (relative entropy) $D(P \,\|\, Q)$ between distributions P and Q is defined as

$$D(P \,\|\, Q) \triangleq \sum_{u \in \Sigma} P(u) \log \frac{P(u)}{Q(u)}. \tag{60}$$

We define the *mean PSMS DR divergence* as

$$D(\Theta) \triangleq \frac{1}{C} \sum_{i=1}^{C} \{D(P_{i-1} \,\|\, P_i) + D(P_i \,\|\, P_{i-1})\}. \tag{61}$$

Theorem 3: Let $k = A \log N$. Assume a PSMS $\{\Theta, \mathcal{T}\}$ with $C < S$ transitions, separated by segments all longer than $O(k)$. Then, the average Nth-order redundancy of the DW scheme is upper-bounded by (62) at the bottom of this page, where

$$\begin{aligned} K_1 &\triangleq \left[\frac{r-1}{2}(C+1)\right] + [(1+\varepsilon)C + \varepsilon] \\ &\quad + \{2.5A[\log e + D(\Theta)]C\}, \end{aligned} \tag{63}$$

$$K_2 \triangleq (A+1)^{4(r-1)} C \log(C+1), \tag{64}$$

$O(\beta)$ denotes the order of β, and $o(\beta)$ denotes order smaller than the order of β.

Theorem 3 demonstrates that we can achieve the order of the lower bound with a fixed complexity scheme if the transitions are large enough, while for smaller transitions we can still achieve decaying redundancy. It is also possible to show that the strongly sequential version of the DW scheme,

$$R_N(\mathcal{D}) \leq \begin{cases} K_1 \frac{\log N}{N} + o\left(\frac{C \log N}{N}\right), & \text{if } E(\Theta) > \frac{2}{A} \text{ and } D(\Theta) < \infty \\ 2.5AC \frac{\log^2 N}{N} + O\left(\frac{C \log N}{N}\right), & \text{if } E(\Theta) > \frac{2}{A} \text{ and } D(\Theta) = \infty \\ K_2 \frac{\log^{4(r-1)} N}{N^{AE(\Theta)-1}} + O\left(\frac{C \log N}{N}\right), & \text{if } \frac{1}{A} < E(\Theta) \leq \frac{2}{A} \\ \frac{r-1}{6A} \frac{\log \log N}{\log N} + O\left(\max\left\{\frac{1}{\log N}, \frac{C \log N}{N}\right\}\right), & \text{if } E(\Theta) \leq \frac{1}{A} \end{cases} \tag{62}$$

proposed in the discussion following (58), achieves the same asymptotical behavior (although the coefficients are larger). The proof of Theorem 3 is presented in Appendix B. The bounds obtained are not tight. Tighter bounds of the same orders may be obtained by a much more complicated analysis than the one presented in Appendix B.

We note the different behavior of the average redundancy for different transition "sizes." If $E(\Theta) > 1/A$, it is likely that there exists a surviving path $\hat{\mathcal{T}}$, such that (s.t.) $\hat{C} = C$, and all transitions are estimated near their true times. For large transitions, $E(\Theta) > 2/A$, the redundancy is mostly influenced by the block partitioning, i.e., by estimating transitions only at block midpoints. This factor increases if the PSMS contains transitions of infinite divergence, thus obtaining the second region of the bound. When the transitions are smaller, $1/A < E(\Theta) \leq 2/A$, the redundancy is determined by the probability that the smallest transition is not detected near its true time. If the PSMS contains very small transitions, $E(\Theta) \leq 1/A$, the scheme cannot ensure that a good estimate of $\mathcal{T}$ will be obtained, and therefore we can only achieve redundancy of higher order.

The number of segments must be bounded by S as a condition of Theorem 3 in order to ensure that if the last true transition has the smallest metric among all transitions, the scheme will still create a surviving state for it. This state represents the true transition path. In practice, however, only the state that represents the most recent true transition, and therefore the true transition path, needs to survive in the diagram. Furthermore, the weight $G(s_n, x_1^n)$ of this state is likely to be larger than the weights of all other states that represent past transitions in the true path, but need not survive in the diagram. Therefore, older surviving states with smaller weights $G(s_n, x_1^n)$ can be eliminated in spite of their large metrics, allowing reuse of states and hence better performance if more than $S-1$ transitions occur.

Theorem 3 requires all segments to be larger than $O(k)$ only for mathematical convenience purposes. This condition results in the very simple expressions for the error exponent and the PSMS divergence, presented above, but has no effect on the nature of the results. Therefore, similar asymptotic behavior, though with different coefficients, is achieved for PSMS's with shorter segments. The mathematical representations of the error exponent and the PSMS divergence, however, become much more complex, and they depend also on $\mathcal{T}$. For instance, in the first region of the upper bound, shorter segments will increase the low-order term to become of the same order as the dominant term. If segments shorter than $3k$ remain undetected (by not detecting at least one of the respective two transitions), they too will result in DR in the order of the bound.

VI. Block Codes

The DW scheme achieves low-order redundancy for some PSMS's, while for others it achieves redundancy that vanishes very slowly as $O(\log\log N/\log N)$. It can be shown that the LZ-78 algorithm has a pointwise upper bound with the same rate (see, e.g., [14]). We desire a scheme that attains better redundancy for the second group of sources, for which the DW scheme performs poorly and does not achieve better rate than LZ. In this section we present such a scheme, that can be combined with the DW scheme in order to achieve the better redundancy for any PSMS. The new scheme is referred to as the *block partitioning* (BP) scheme and will be denoted by $\mathcal{P}$.

Fig. 5. Description of block partitioning. The horizontal line represents the time axis and the vertical lines the partition points.

The BP scheme partitions the N-tuple into B blocks, and codes each block b, $1 \leq b \leq B$ of length m_b as if it were a stationary segment, using its KT estimate. The probability assigned to an N-tuple is defined as

$$Q_{\mathcal{P}}(x^N) \triangleq Q(x^N \mid \hat{\mathcal{T}}) \tag{65}$$

where $\hat{\mathcal{T}} \triangleq \{\hat{t}_1, \hat{t}_2, \cdots, \hat{t}_{B-1}\}$ is independent of x^N and is recursively defined as

$$\hat{t}_b \triangleq \hat{t}_{b-1} + m_b, \qquad 1 \leq b \leq B-1 \tag{66}$$

where $\hat{t}_0 \triangleq 1$. (Hence, by definition, $B \triangleq \hat{C} + 1$.) The idea is to choose the set $\{m_b\}$ that will give the fastest decay of the *pointwise* redundancy *uniformly* over all PSMS's. We will achieve decay rate slower than $O(\log N/N)$ but faster than $O(\log\log N/\log N)$.

From (16) we note that there is no TR, since there is a single transition path and no weighting. The redundancy is, therefore, obtained by trading off the PR, which decreases with the block length since $\hat{C}$ decreases (see (15)), and the DR which increases with the block length (see (17)). It can be shown that for a given N, the best tradeoff is achieved by selecting the block length as

$$m_{\text{opt}} = O(\sqrt{N \log N}), \qquad \forall b. \tag{67}$$

Since N is unknown in advance, we can define the block length to increase with n s.t. at time instant n the length of a block will be $O(\sqrt{n \log n})$. The BP block length, obtained by the following equation, satisfies this requirement:

$$m_b = \lfloor \alpha b \log b \rfloor, \qquad b > 1 \tag{68}$$

where $m_1 \triangleq 1$. The parameter α is a design parameter. This assignment ensures that the last blocks will be $O(\sqrt{N \log N})$ and larger than the preceding blocks, and therefore will dominate the redundancy. Fig. 5 demonstrates the partitioning of an N-tuple.

Theorem 4: The pointwise redundancy of the BP scheme is bounded uniformly for all PSMS's with C transitions by

$$R(x^N; \mathcal{P}) \leq \left(\frac{r-1}{2\sqrt{\alpha}} + C\sqrt{\alpha}\right)\sqrt{\frac{\log N}{N}} + o\left(C\sqrt{\frac{\log N}{N}}\right), \qquad \forall x^N. \tag{69}$$

The proof of Theorem 4 is presented in Appendix E. Again, we make no assumptions on the number of transitions C, although it is easy to see that the upper bound is no longer useful if

$C > o(\sqrt{(N/\log N)})$. It is easy to show that if C is known in advance, the choice of

$$\alpha = \frac{r-1}{2C} \tag{70}$$

will obtain the best upper bound

$$R(x^N; \mathcal{P}) \le \sqrt{2(r-1)C}\sqrt{\frac{\log N}{N}} + o\left(C\sqrt{\frac{\log N}{N}}\right), \qquad \forall x^N. \tag{71}$$

If C is unknown, we can choose $\alpha = (r-1)/2$ to obtain

$$R(x^N; \mathcal{P}) \le \frac{(C+1)\sqrt{r-1}}{\sqrt{2}}\sqrt{\frac{\log N}{N}} + o\left(C\sqrt{\frac{\log N}{N}}\right), \qquad \forall x^N. \tag{72}$$

The BP scheme is very simple to implement and requires a single state only. Its per-letter computational complexity is $O(1)$ and its total storage complexity is $O(\log N)$.

We can easily combine the DW and the BP schemes into a combined scheme, denoted by $\mathcal{C}$. Obviously, the probability assignment

$$Q_{\mathcal{C}}(x^N) \triangleq \frac{1}{2}Q_{\mathcal{D}}(x^N) + \frac{1}{2}Q_{\mathcal{P}}(x^N) \tag{73}$$

attains the minimal redundancy between the two schemes. The pointwise redundancy of this scheme is always upper-bounded by $O\left(C\sqrt{\log N/N}\right)$ as in Theorem 4. If a PSMS satisfies the conditions of Theorem 3, its average Nth-order redundancy is upper-bounded by (74) at the bottom of this page.

VII. Simulation Results

In this section we present numerical examples of the performance of the schemes presented in Sections IV–VI, and compare them to the performance of the schemes presented in [19]–[22]. We show that we achieve better performance with the new schemes, and that the true redundancies are much smaller than the upper bounds.

Fig. 6 compares pointwise redundancies of the linear schemes (Willems' scheme $\mathcal{W}$, and the two optimal schemes, $\mathcal{L}_1$ and $\mathcal{L}_2$) for a sequence of length 1000, drawn by a binary PSMS with $C = 3$ transitions. For both $\mathcal{L}_1$ and $\mathcal{L}_2$, we take $\varepsilon = 0.1$. The curves demonstrate that both $\mathcal{L}_1$ and $\mathcal{L}_2$ achieve better redundancies than $\mathcal{W}$. The redundancy of $\mathcal{L}_1$ is the better one between the two, before the last transition. However, $\mathcal{L}_2$ performs better after the last transition. This is because the third segment is relatively short, and justifies weights that are more "generous" to new transitions, as those used in $\mathcal{L}_2$, while the second segment is long enough to justify the less "generous" weights of $\mathcal{L}_1$. This result demonstrates the tradeoff in performance between the two schemes. If shorter segments are expected (i.e., more transitions in a given time interval), $\mathcal{L}_2$ achieves better results, while if longer segments are expected, it is better to use $\mathcal{L}_1$. The true performance of all schemes is better than the upper bounds of Theorem 1. This is because the upper bounds are pessimistic by not taking into account weights of adjacent transition paths, and by bounding all cases by the worst case, in which all segments are of the same length $N/(C+1)$.

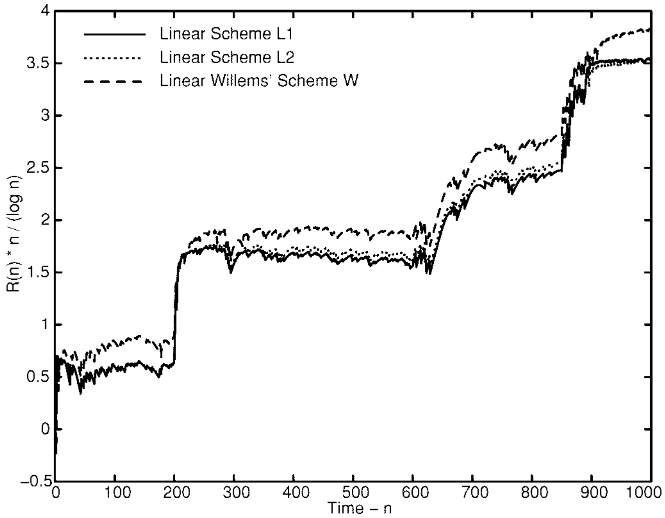

Fig. 6. Pointwise redundancies of the linear schemes for $N = 10^3$ bits drawn by abinary PSMS, $\Theta = \{0.8, 0.2, 0.1, 0.4\}$, $\mathcal{T} = \{201, 601, 851\}$. The redundancies are multiplied by $n/(\log n)$. For both $\mathcal{L}_1$ and $\mathcal{L}_2$, $\varepsilon = 0.1$.

Figs. 7 and 8 demonstrate redundancies obtained by Willems' logarithmic scheme [22], and by the DW scheme with different parameters for a binary PSMS with $C = 3$ large transitions. In Fig. 7, pointwise redundancies for a single N-tuple are presented, while Fig. 8 presents the mean of 50 trials. The DW scheme is shown in both graphs to perform better than the logarithmic scheme. Both graphs demonstrate that the DW scheme achieves redundancy of $O(\log N/N)$, even for transitions for which $E(\Theta)$ is much smaller than $2/A$. For the PSMS in the example, $E(\Theta) \approx 0.068$. Using block length $k = 200$, we have $2/A \approx 0.2 > E(\Theta)$, but the DW still achieves the order of the bound. However, for $k = 100$, the scheme may perform well for some N-tuples, as shown in Fig. 7, but the probability of not detecting the last transition is not negligible, as shown by the respective curve in Fig. 8. Using $k = 400$, we achieve larger redundancy than with the shorter blocks, because transitions can only be estimated every 400 time units. Note that the curve for $k = 300$ demonstrates similar performance to the performance demonstrated by the

$$R_N(\mathcal{C}) \le \begin{cases} K_1\frac{\log N}{N} + o\big(\frac{C\log N}{N}\big), & \text{if } E(\Theta) > \frac{2}{A} \text{ and } D(\Theta) < \infty \\ 2.5AC\frac{\log^2 N}{N} + O\big(\frac{C\log N}{N}\big), & \text{if } E(\Theta) > \frac{2}{A} \text{ and } D(\Theta) = \infty \\ \big(\frac{r-1}{2\sqrt{\alpha}} + C\sqrt{\alpha}\big)\sqrt{\frac{\log N}{N}} + o\big(C\sqrt{\frac{\log N}{N}}\big), & \text{otherwise.} \end{cases} \tag{74}$$

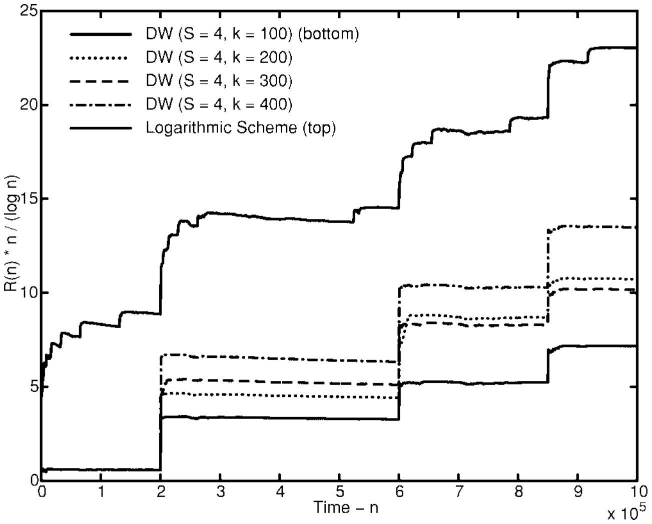

Fig. 7. Pointwise redundancies of the logarithmic scheme (top curve) and the DW scheme with different parameters (bottom curves) for $N = 10^6$ bits drawn by arbinary PSMS, $\Theta = \{0.8, 0.2, 0.7, 0.4\}$, $\mathcal{T} = \{2\cdot 10^5 + 1, 6\cdot 10^5 + 1, 8.5\cdot 10^5 + 1\}$. The redundancies are multiplied by $n/(\log n)$. For the DW scheme $\varepsilon = 0.1$.

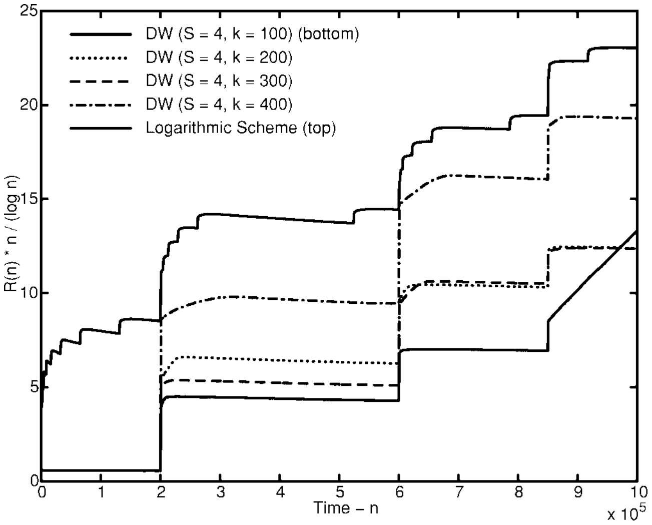

Fig. 8. Mean redundancies of 50 trials of the logarithmic scheme (top curve) and the DW scheme with different parameters (bottom curves) for $N = 10^6$ bits drawn by a binary PSMS, $\Theta = \{0.8, 0.2, 0.7, 0.4\}$, $\mathcal{T} = \{2\cdot 10^5 + 1, 6\cdot 10^5 + 1, 8.5\cdot 10^5 + 1\}$. The redundancies are multiplied by $n/(\log n)$. For the DW scheme $\varepsilon = 0.1$.

curve for $k = 200$ because of the specific source parameters, for which some of the transitions are estimated closer to their true times with $k = 300$ than with $k = 200$. The parameter S has no influence on the performance of the DW scheme as long as there are enough surviving states. Therefore, similar curves are obtained for the DW scheme with any $S > 4$. Furthermore, the combined DW-BP scheme $\mathcal{C}$ obtains the same curves as the DW scheme because the DW has better redundancy than the BP in this case. Since $D(\Theta) \approx 1.52$, it is apparent that the DW scheme attains much smaller redundancy than the upper bound of Theorem 3, because the bound is not tight.

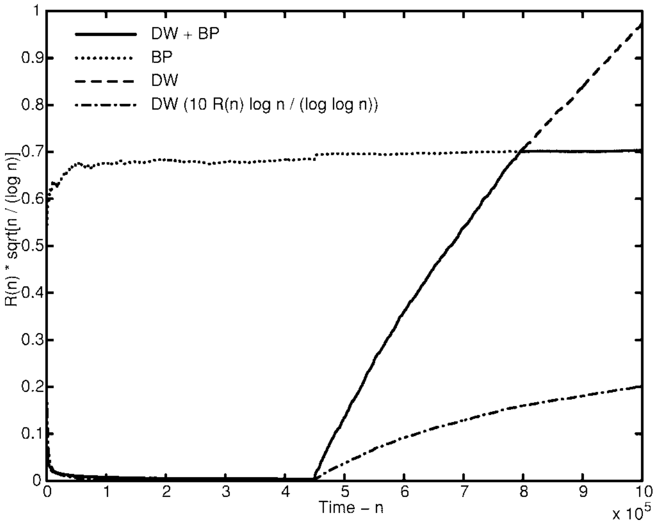

Fig. 9. Pointwise redundancies of the DW scheme ($S = 2, k = 50, \varepsilon = 0.1$), BP scheme ($\alpha = 0.5$), and the combined scheme with the same parameters for $N = 10^6$ bits drawn by a binary PSMS with a small transition $\Theta = \{0.2, 0.1\}$, $\mathcal{T} = \{4.5\cdot 10^5 + 1\}$. The redundancies in the upper three curves are multiplied by$\sqrt{n/(\log n)}$, and the redundancy in the bottom curve by $10 \log n/(\log\log n)$. The redundancy of the DW is shown twice with different scalings in a top curve and in the bottom curve.

Fig. 9 illustrates the performance of the coding schemes in the case of a single small transition, in which the BP performs better than the DW. The redundancy of the DW is shown twice, once normalized by the same factor as other redundancies, in order to compare performance to the other schemes, and a second time multiplied by the inverse of its expected order to demonstrate its order. The redundancy of the DW is shown to be of $O(\log\log N/\log N)$. Before the transition occurs, the DW attains better redundancy. Hence, the combined scheme takes this redundancy. However, after the transition occurs, the DW starts to perform poorly. At some point, where its redundancy becomes larger than that of the BP, the combined scheme attains the redundancy of the BP. The redundancy of the BP is shown to be of $O(\sqrt{\log N/N})$.

VIII. Summary and Conclusions

In this paper we investigated the problem of low-complexity universal coding of a PSMS. We showed that the entropy of the source can be asymptotically achieved with fixed complexity schemes, and that these schemes can attain redundancies that decay faster than those obtained by any known low-complexity scheme for coding PSMS's. Specifically, it was shown that the order of the lower bound on the decay rate of the redundancy can be achieved when the transitions in the statistics are large, and for smaller transitions the order of its square root is achieved. The lower bound itself was achieved by an optimal linear per-letter complexity scheme that was presented. Finally, all results were supported by simulations that showed that in practice all algorithms perform much better than the performance suggested by the analysis. All the schemes can be extended to more complex piecewise-stationary sources using context tree coding schemes.

Appendix A
Proof of Theorem 2

We begin the proof of Theorem 2 with a lemma.

Lemma A.1: Let $\hat{\mathcal{T}}$ be a transition path that is not eliminated from the transition diagram of the DW scheme, and let $\hat{C}$ be the number of transitions assumed by $\hat{\mathcal{T}}$. Then the pointwise TR is upper-bounded as in (16) by

$$\begin{aligned} R_t(x^N;\mathcal{D}) &\le -\frac{1}{N}\log W_{\mathcal{D}}(\hat{\mathcal{T}}) \\ &\le \frac{1}{N}\Bigg[\hat{C}\log\frac{N}{k\hat{C}} + (\hat{C}+1)\varepsilon\log\frac{N+0.5k}{k(\hat{C}+1)} \\ &\qquad + (\hat{C}+1)\log\frac{1+\varepsilon}{\varepsilon} + \log\varepsilon\Bigg] \qquad \text{(A.1)} \end{aligned}$$

where $W_{\mathcal{D}}(\hat{\mathcal{T}})$ is the cumulative weight assigned to the path $\hat{\mathcal{T}}$ by (50).

Proof: The DW scheme weighs all transition paths that survive in the diagram, with additional weights obtained from paths that lead to eliminated states. Hence, unlike equality (10)

$$Q_{\mathcal{D}}(x^N) \ge \sum_{\mathcal{T}'\in\mathcal{D}} W_{\mathcal{D}}(\mathcal{T}')Q(x^N \mid \mathcal{T}') \ge W_{\mathcal{D}}(\hat{\mathcal{T}})Q(x^N \mid \hat{\mathcal{T}}) \qquad \text{(A.2)}$$

where $\mathcal{T}'\in\mathcal{D}$ denotes the set of (surviving) transition paths that exist in the diagram at time N, and $\hat{\mathcal{T}}$ is one of these paths that is chosen as the estimate of the true transition path. Hence, by definition of the TR in (14), the first inequality of the lemma is proved, and we require an upper bound on $-\log W_{\mathcal{D}}(\hat{\mathcal{T}})$ to prove the second. □

The upper bound on $-\log W_{\mathcal{D}}(\hat{\mathcal{T}})$ can be attained from the bound on the TR of $\mathcal{L}_2$ in (34). Note that the only transitions that contribute nonzero elements to $-\log W_{\mathcal{D}}(\hat{\mathcal{T}})$ occur at block midpoints, where new states are created. The transition weights at these time points, which are defined in (50), are equivalent to the weights of $\mathcal{L}_2$, defined in (28), but with a shrinking transformation of the time axis. Since such transitions occur only every k time units at $\lfloor N/k+0.5\rfloor$ time points over the complete data sequence, the horizon N in the bound of (34) can be replaced by $N/k+0.5$. (Note that the horizon of the first dominant term is replaced by N/k, because it is derived from the first $\hat{C}$ segments, excluding the last hypothesized segment.) We conclude the proof of Lemma A.1, by replacing C in the bound of (34) by $\hat{C}$, since it can be applied to all surviving paths, and not only the true one.

Proof of Theorem 2: We now use the lemma to prove Theorem 2. The heart of the proof is the choice of $\hat{\mathcal{T}}$. Let

$$\hat{\mathcal{T}} \triangleq \{1.5k+1, 4.5k+1, 7.5k+1, \cdots\}.$$

The path $\hat{\mathcal{T}}$ is a partition of N into blocks of length $3k$. By definition of the DW scheme, a state is never eliminated before it is used for coding at least $3.5k$ data letters. Hence, the path $\hat{\mathcal{T}}$ always exists in the transition diagram and thus can be used to estimate $\mathcal{T}$. By definition of $\hat{\mathcal{T}}$

$$N/(3k) - 1 \le \hat{C} \le N/(3k) + 1.$$

Using (15), the PR can be upper-bounded by

$$R_p(x^N;\mathcal{D}) \le \frac{r-1}{6}\frac{\log k}{k} + O\left(\frac{1}{k}\right). \qquad \text{(A.3)}$$

Substituting $\hat{C}$ in (A.1), we bound the TR by

$$\begin{aligned} R_t(x^N;\mathcal{D}) &\le \frac{(1+\varepsilon)\log 3 + \log[(1+\varepsilon)/\varepsilon]}{3k} + O\left(\frac{1}{N}\right) \\ &= O\left(\frac{1}{k}\right). \qquad \text{(A.4)} \end{aligned}$$

To obtain the upper bound for the DR, we take the worst case in which each transition contributes the most. This case occurs when there is at most a single true transition in each hypothesized segment of $\hat{\mathcal{T}}$. Since the hypothesized segments are of length $3k$, the DR is bounded, using (17), by

$$R_d(x^N;\mathcal{D}) \le \frac{3Ck}{N} = O\left(\frac{Ck}{N}\right). \qquad \text{(A.5)}$$

The proof of Theorem 2 is concluded by realizing that R_p determines the dominant term of the redundancy.

Appendix B
Proof of Theorem 3

To prove Theorem 3, we address two different regions of the error exponent separately. For $E(\Theta) \le 1/A$, we simply use the upper bound of Theorem 2 that applies to the redundancy of any sequence, and thus can be applied to the average redundancy for PSMS's with small error exponents.

We now prove the upper bounds for the other three regions. To analyze the average redundancy we select a surviving path $\hat{\mathcal{T}}$ that is most likely to be a good estimate of the true path $\mathcal{T}$. This path is used to form a partition of all data sequences drawn by the PSMS into two disjoint sets. The first is the set of N-tuples for which $\hat{\mathcal{T}}$ is a good estimate, i.e., all transitions are estimated near their true time unit. The second set contains all the other N-tuples. We then upper-bound the probability of each set and the average redundancy of all N-tuples in the set. Summing up both terms we obtain an upper bound for the average redundancy.

Let $\hat{\mathcal{T}} = \hat{\mathcal{T}}(x^N)$ be an estimate of the true path $\mathcal{T}$, s.t. $\hat{\mathcal{T}}$ connects the $C+1$ states with largest metrics (including the first state $s = 1$), and $\hat{C} = C$. It is assumed as a condition of the theorem that $C+1 \le S$. Hence, we will always have $C+1$ states with largest metrics, and, therefore, this path will exist. Each transition $t_i \in \mathcal{T}$ is estimated by a transition $\hat{t}_i \in \hat{\mathcal{T}}$. Let $\mathcal{S} \triangleq \phi$ be a second estimate of the true transition path that assumes no transitions following $t_0 = 1$, i.e., $C(\mathcal{S}) = 0$. This path always exists in the diagram, and is represented by the state $s_N = 1$, which must survive due to its infinite metric.

Let the set $\bar{F}$ be the set of all N-tuples, for which $|\hat{t}_i - t_i| \le 2.5k$; $\forall i,\ 1 \le i \le C$, i.e., all transitions are detected by $\hat{\mathcal{T}}$ near their true time points. Let F be the complimentary set of N-tuples for which there exists a transition that is detected farther than $2.5k$ time points away from its true time. As shown in Fig. 10, if a transition t_i is detected at $\hat{t}_i$ more than $2.5k$ time points away from its true time, nonoverlapping blocks

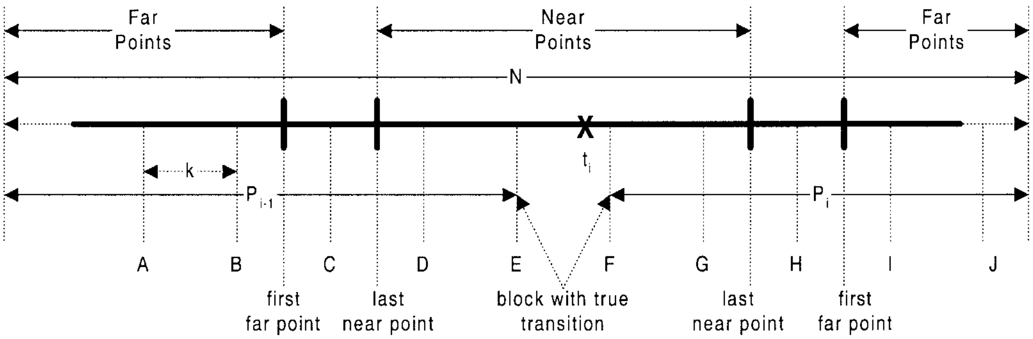

Fig. 10. Far- and near-point definitions. The solid line represents the time axis. A true transition occurs within the central block EF and is noted by an "X." Points referred to as near points are at most $2.5k$ time units away from the transition. The first far point is one block away from the last near point, and obtains its metric from empirical data of blocks that do not overlap the blocks that are used to obtain the metric of the true transition.

are used to obtain $M(t_i)$ and $M(\hat{t}_i)$, while if $|\hat{t}_i - t_i| \le 2.5k$, overlapping blocks may have been used for both metrics. For example: If a true transition occurs in block EF, then block FG is used for its metric. Block GH, that is the first nonoverlapping block, is used to detect transition in block HI. If transition is detected in HI, it is estimated at its midpoint, which is $2.5k+1$ to $3.5k$ time points away from any point in EF, where the true transition occurs.

The average Nth-order redundancy can be expressed as

$$R_N(\mathcal{D}) = \Pr(F)\cdot R_N(\mathcal{D}\mid F) + (1-\Pr(F))\cdot R_N(\mathcal{D}\mid \bar{F}) \tag{B.1}$$

where $R_N(\mathcal{D}\mid \mathcal{E})$ denotes the average Nth-order redundancy given event $\mathcal{E}$ occurs. We can now treat each term separately to prove the theorem. We next present three propositions that upper-bound different terms of (B.1), all assuming the conditions of the theorem. The proofs will be presented in Appendix C and Appendix D. The mean DR of $\mathcal{D}$ given event $\mathcal{E}$ is denoted by $R_d^N(\mathcal{D}\mid\mathcal{E})$.

Proposition B.1: The probability of F is upper-bounded by

$$\Pr(F) \le C\cdot N\cdot 2^{-k[E(\Theta)-\frac{4(r-1)}{k}\log(k+1)]}. \tag{B.2}$$

Proposition B.2:

$$R_d(x^N) \le 2.5C\frac{k\log N}{N} + O\left(\frac{Ck}{N}\right), \qquad \forall x^N \in \bar{F} \tag{B.3}$$

$$R(x^N) \le 2.5AC\frac{\log^2 N}{N} + O\left(\frac{C\log N}{N}\right), \qquad \forall x^N \in \bar{F}. \tag{B.4}$$

Proposition B.3:

$$\Pr(\bar{F})\cdot R_d^N(\mathcal{D}\mid\bar{F}) \le 2.5[\log e + D(\Theta)]C\frac{k}{N} + o\left(\frac{Ck}{N}\right) \tag{B.5}$$

$$\Pr(\bar{F})\cdot R_N(\mathcal{D}\mid\bar{F}) \le K_1\frac{\log N}{N} + o\left(\frac{C\log N}{N}\right). \tag{B.6}$$

We will use the path $\mathcal{S}$ to estimate the transition paths of all N-tuples in F, for which $\hat{\mathcal{T}}$ is not a good estimate, and $\hat{\mathcal{T}}$ to estimate the paths of N-tuples in $\bar{F}$. Since the path $\mathcal{S}$ assumes the whole N-tuple is coded as a single hypothesized stationary segment, we can use (17) with $m = N$ and $s = C+1$ to upper-bound $R_d(x^N;\mathcal{D})$

$$R_d(x^N;\mathcal{D}) \le \frac{m\log s}{N} = \log(C+1), \qquad \forall x^N \in F. \tag{B.7}$$

The path $\mathcal{S}$ assumes no transitions. Therefore, using Lemma A.1 we obtain TR of $O(\log N/N)$ and using (15) we obtain PR of the same order. Thus

$$R(x^N;\mathcal{D}) \le \log(C+1) + O\left(\frac{\log N}{N}\right), \qquad \forall x^N \in F. \tag{B.8}$$

Using this bound and Proposition B.1 and taking $k = A\log N$ and $(A\log N + 1) \le (A+1)\log N$, we can upper-bound the first term of (B.1) by

$$\Pr(F)\cdot R_N(\mathcal{D}\mid F) \le (A+1)^{4(r-1)}C\frac{\log^{4(r-1)}N}{N^{AE(\Theta)-1}} \cdot \left[\log(C+1) + O\left(\frac{\log N}{N}\right)\right]. \tag{B.9}$$

Summing up (B.9) and (B.6) of Proposition B.3, we obtain an upper bound on the average redundancy for $D(\Theta) < \infty$. If $E(\Theta) > 2/A$, the term of (B.6) is dominant, thus obtaining the first region of the upper bound. If $1/A < E(\Theta) \le 2/A$, the term of (B.9) is dominant, resulting in the third region of the upper bound. If $E(\Theta) \le 1/A$, the upper bound of (B.9) is no longer useful. If $D(\Theta) = \infty$, we upper-bound the probability of $\bar{F}$ by 1, and (B.4) of Proposition B.2 results in the dominant term of the redundancy if $E(\Theta) > 2/A$, obtaining the second region of the bound. This concludes the proof of Theorem 3.

Appendix C
Proof of Proposition B.1

We begin the proof of Proposition B.1 with a few definitions. We then present and prove a lemma, upon which we base the proof of the proposition. Let $\hat{\mathcal{T}}$ be defined as in Appendix B, and let t_i, $0 < i \le C$ be the ith true transition and s_i the block that contains t_i. The distribution before t_i is P_{i-1} and at t_i it becomes P_i. Now, let r be any block, s.t. there exists j, $0 < j \le C$ for which $s_j + 1 < r < s_{j+1} - 1$. For generalization we define $s_0 \triangleq 0$ and $s_{\lfloor N/k\rfloor} \triangleq \lfloor N/k\rfloor + 1$. Block r is defined s.t. the three blocks $r-1$, r, and $r+1$ are entirely within the single true stationary segment j with distribution P_j.

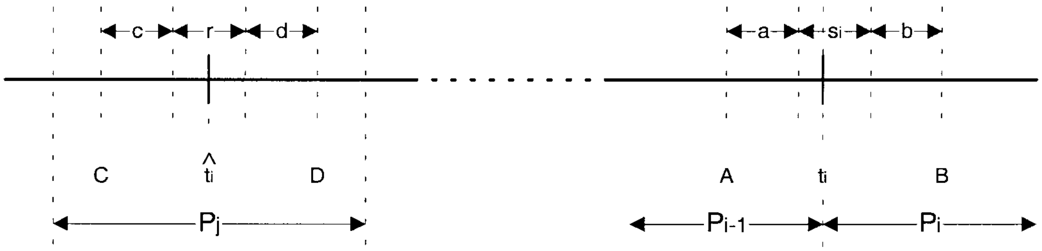

Fig. 11 Typical occurrence of event F. The likelihood $M(r)$, obtained from blocks c and d both drawn by distribution P_j, is larger than $M(s_i)$ that represents the true transition t_i and is obtained from blocks a and b.

Lemma C.1: If $x^N \in F$ there exist s_i, $1 \le i \le C$, and r as defined above s.t. $M(s_i) \le M(r)$, where both metrics are obtained from nonoverlapping blocks.

Proof: For a transition in a block s_l, there are at most three states $s_l - 1$, s_l, and $s_l + 1$ that may have metrics obtained by data of two different stationary segments. The DW scheme allows at most one of these three consecutive states to survive in the diagram and eliminates the other two. By definition of F, there exists $\hat{t}_i$, s.t. $|\hat{t}_i - t_i| > 2.5k$. Thus there is a transition t_i, for which all these three states were eliminated, but since $\hat{C} = C$ this transition has an estimate. This estimate must be at some block r, s.t. r and its neighboring blocks are entirely within a stationary segment, because all surviving states with metrics obtained from two different segments are associated with other true transitions. Since s_i is eliminated, and it is assumed that the distance between transitions is larger than $O(k)$, it must be that s_i was eliminated because $M(s_i) \le M(r)$. One of the blocks used to obtain $M(r)$ may still overlap a block used for a metric of some s_j, but $s_j \ne s_i$ since $|\hat{t}_i - t_i| > 2.5k$. □

As a result of the lemma, we can bound $\Pr(F)$ by

$$\Pr(F) \le \Pr\{\exists s_i, r : M(s_i) \le M(r)\} \triangleq \Pr(A). \qquad \text{(C.1)}$$

We base the proof of the proposition on this result. We select some fixed s_i and r, and upper-bound the probability of event $A_{ir} \triangleq \{M(s_i) \le M(r)\}$. Then we use the union bound twice, once over all possible values of r and then on the values of i, to upper-bound the probability of the right-hand side (r.h.s.) of inequality (C.1), which we denote as the probability of event A.

Observe some fixed s_i and r as defined above. For convenience, let us define $a \triangleq s_i - 1$, $b \triangleq s_i + 1$, $c \triangleq r - 1$, and $d \triangleq r + 1$. Fig. 11 illustrates this block partitioning. We define the empirical PMF P_α of block α as

$$P_\alpha(u) \triangleq \frac{n_\alpha(u)}{k}, \qquad \forall u \in \Sigma \qquad \text{(C.2)}$$

where $n_\alpha(u)$ is the number of occurrences of letter u in block α. Similarly, we define the empirical PMF of the concatenation of blocks α and β as

$$P_{\alpha\beta}(u) \triangleq \frac{n_\alpha(u) + n_\beta(u)}{2k}, \qquad \forall u \in \Sigma. \qquad \text{(C.3)}$$

The empirical per-letter entropies of a block and of a concatenation of two blocks are obtained by (43) and (44), respectively.

By definition of A_{ir} and $M(\cdot)$, we have

$$\begin{aligned} A_{ir} = \{x^N : &H(c,d) - 0.5H(c) - 0.5H(d) \\ &\ge H(a,b) - 0.5H(a) - 0.5H(b)\}. \end{aligned} \qquad \text{(C.4)}$$

Rearranging terms, we can express A_{ir} by means of divergence

$$\begin{aligned} A_{ir} \triangleq \{x_1^N : &D(P_c \,\|\, P_{cd}) + D(P_d \,\|\, P_{cd}) \\ &\ge D(P_a \,\|\, P_{ab}) + D(P_b \,\|\, P_{ab})\}. \end{aligned} \qquad \text{(C.5)}$$

By typical sets analysis (see [2] and [3]), and since blocks a, b, c, and d are independent of each other, we can bound the probability of A_{ir} by

$$\Pr(A_{ir}) \le 2^{-k[E(P_{i-1}, P_i, P_j) - \frac{4(r-1)}{k} \log(k+1)]} \qquad \text{(C.6)}$$

where $E(P_{i-1}, P_i, P_j)$ is defined as

$$\begin{aligned} E(P_{i-1}, P_i, P_j) = \min_{A_{ir}} \{&D(P_a \,\|\, P_{i-1}) + D(P_b \,\|\, P_i) \\ &+ D(P_c \,\|\, P_j) + D(P_d \,\|\, P_j)\}. \end{aligned} \qquad \text{(C.7)}$$

(The second-order term of (C.6) results from the bound on the number of types in a block of length k, which is $(k+1)^{(r-1)}$. This bound is used to bound the number of types in each of the four independent blocks, and is therefore raised to the power of 4.) We will now define an event B_{ir}, for which the minimum of (C.7) is easier to compute, s.t. $A_{ir} \subseteq B_{ir}$. Since A_{ir} is a subset of B_{ir}, the minimization over B_{ir} will lower-bound the exponent defined by (C.7). It is easy to show that

$$\begin{aligned} &D(P_c \,\|\, P_{cd}) + D(P_d \,\|\, P_{cd}) \\ &\quad = D(P_c \,\|\, P_j) + D(P_d \,\|\, P_j) - 2D(P_{cd} \,\|\, P_j) \\ &\quad \le D(P_c \,\|\, P_j) + D(P_d \,\|\, P_j). \end{aligned} \qquad \text{(C.8)}$$

We define B_{ir} as

$$\begin{aligned} B_{ir} \triangleq \{x_1^N : &D(P_c \,\|\, P_j) + D(P_d \,\|\, P_j) \ge D(P_a \,\|\, P_{ab}) \\ &+ D(P_b \,\|\, P_{ab})\}. \end{aligned} \qquad \text{(C.9)}$$

Since $A_{ir} \subseteq B_{ir}$

$$\begin{aligned} &E(P_{i-1}, P_i, P_j) \\ &\quad \ge \min_{B_{ir}, C} \{D(P_a \,\|\, P_{i-1}) + D(P_b \,\|\, P_i) \\ &\qquad + D(P_c \,\|\, P_j) + D(P_d \,\|\, P_j)\} \\ &\quad \ge \min_{C} \{D(P_a \,\|\, P_{i-1}) + D(P_b \,\|\, P_i) \\ &\qquad + D(P_a \,\|\, P_{ab}) + D(P_b \,\|\, P_{ab})\} \\ &\quad = \min_{C} \left\{ \begin{array}{l} \sum_{x \in \Sigma} P_a(x) \log \frac{2P_a^2(x)}{P_{i-1}(x)[P_a(x) + P_b(x)]} + \\ \sum_{x \in \Sigma} P_b(x) \log \frac{2P_b^2(x)}{P_i(x)[P_a(x) + P_b(x)]} \end{array} \right\} \end{aligned} \qquad \text{(C.10)}$$

where the constraint $\mathcal{C}$ is defined by

$$\sum_{x\in\Sigma} P_a(x) = \sum_{x\in\Sigma} P_b(x) = 1. \tag{C.11}$$

The second inequality is obtained by applying the constraint of B_{ir}, while the last equality is obtained by definition of divergence and by expressing $P_{ab}(x)$ as $0.5(P_a(x)+P_b(x))$. The constrained minimization is performed using Lagrange multipliers. We define the functional $J(P_a, P_b)$ as

$$\begin{aligned} J(P_a,P_b) \triangleq & \sum_{x\in\Sigma} P_a(x) \log \frac{2P_a^2(x)}{P_{i-1}(x)[P_a(x)+P_b(x)]} \\ & + \sum_{x\in\Sigma} P_b(x) \log \frac{2P_b^2(x)}{P_i(x)[P_a(x)+P_b(x)]} \\ & + \lambda_a \left(\sum_{x\in\Sigma} P_a(x) - 1 \right) + \lambda_b \left(\sum_{x\in\Sigma} P_b(x) - 1 \right). \end{aligned} \tag{C.12}$$

It is straightforward to show that the Hessian metrix of $J(P_a, P_b)$ w.r.t. $P_a(x)$ and $P_b(x)$ for $x \in \Sigma$ is positive-definite. Hence the functional is convex and obtains the minimum where the first derivatives are zeros. By differentiation we obtain

$$E(P_{i-1}, P_i, P_j) \geq -2 \log \frac{\sum_{x\in\Sigma} \sqrt{P_{i-1}(x)P_i(x)} + 1}{2}. \tag{C.13}$$

Substituting the error exponent by its minimal value over all transitions, we can upper-bound the probability of A_{ir} by

$$\Pr(A_{ir}) \leq 2^{-k[E(\Theta) - \frac{4(r-1)}{k}\log(k+1)]}. \tag{C.14}$$

Using the union bound and accounting for all $\lfloor N/k{-}3C \rfloor < N$ values of r first and then for all C values of i, we obtain the upper bound on $\Pr(A)$. By using (C.1) we apply this bound to $\Pr(F)$ and conclude the proof of Proposition B.1.

Appendix D
Proofs of Propositions B.2 and B.3

In this appendix we present the proofs of Propositions B.2 and B.3, that summarize the contribution of event $\bar{F}$ to the average redundancy. The difficulty in proving Proposition B.3 lies in the fact that the time instants of the estimates $\hat{t}_i$ vary for different N-tuples $x^N \in \bar{F}$, thus we cannot assume anything about the distances $|\hat{t}_i - t_i|$ except that they are bounded by $2.5k$. We begin with analyzing the DR of an N-tuple $x^N \in \bar{F}$ by breaking it into $C{+}1$ terms, each representing the contribution of a single segment. We then break each such term into two different terms, which are analyzed separately. For each term we obtain a pointwise upper bound and an average one. Finally, we reconstruct the total DR by adding all the separate terms in a pointwise manner to prove Proposition B.2 and in the average to prove Proposition B.3. The second parts of both propositions are obtained by adding the PR and the TR to the DR. Throughout this section, we use the definition of $\hat{\mathcal{T}}$ presented in Appendix B.

We begin with some definitions. For $0 \leq i \leq C$, we define the following block lengths:

$$\begin{aligned} a_i &\triangleq \max(t_i - \hat{t}_i, 0) \\ b_i &\triangleq \max(\hat{t}_i - t_i, 0) \\ c_i &\triangleq \min(t_{i+1}, \hat{t}_{i+1}) - \max(t_i, \hat{t}_i). \end{aligned} \tag{D.1}$$

We note that $a_0 = b_0 = 0$ and that for any i, either a_i or b_i must be zero. For generalization purposes we define $a_{C+1} = b_{C+1} \triangleq 0$. We next define the vectors associated with each length.

$$\begin{aligned} x_a^i &\triangleq (x_{\hat{t}_i}, x_{\hat{t}_i+1}, \cdots, x_{t_i-1}) \\ x_b^i &\triangleq (x_{t_i}, x_{t_i+1}, \cdots, x_{\hat{t}_i-1}) \\ x_c^i &\triangleq (x_{t_i+b_i}, x_{t_i+b_i+1}, \cdots, x_{t_{i+1}-a_{i+1}-1}). \end{aligned} \tag{D.2}$$

If the last index of a vector is smaller than the first, the vector will be the empty vector ϕ by definition. Therefore, for every i either x_a^i or x_b^i must be the empty vector. The probability of the empty vector with any distribution will be defined as 1. The nonempty vectors obtained from the $C+1$ sets of the above three vectors are a complete parsing of the N-tuple into disjoint strings. Hence, we can express the DR as the sum of the contributions of all these vectors, where the contribution of ϕ is zero. Finally, the empirical distribution of vector $x_\alpha^i \neq \phi$ is defined as

$$P_\alpha^i(u) \triangleq \frac{n_\alpha^i(u)}{\alpha_i}, \qquad \forall u \in \Sigma \tag{D.3}$$

where $n_\alpha^i(u)$ is the number of occurrences of u in x_α^i. Fig. 12 demonstrates the definitions presented above. Each sting x_α^i is noted by its length α_i.

We can now use the above definitions to define the distributions Q_i that the path $\hat{\mathcal{T}}$ assigns to each of its hypothesized segments. For convenience, we define

$$A_i \triangleq a_i + c_i + b_{i+1} \tag{D.4}$$

and then

$$Q_i(u) \triangleq \frac{1}{A_i}[a_i P_{i-1}(u) + c_i P_i(u) + b_{i+1} P_{i+1}(u)], \qquad 0 \leq i \leq C, \quad \forall u \in \Sigma \tag{D.5}$$

where P_i is the true distribution of segment i. The distributions P_{-1} and P_{C+1} need not be defined, since they will always be multiplied by zero. Fig. 12 illustrates the true and the hypothesized distributions around transition t_i for both cases $\hat{t}_i > t_i$ and $\hat{t}_i < t_i$. In both diagrams we assume $a_{i-1} > 0$ and $b_{i+1} > 0$.

We can now represent the DR of $x^N \in \bar{F}$ as the sum of $C{+}1$ terms (one for each segment), each consists of two terms, the first is the contribution of vectors a, b and the second of

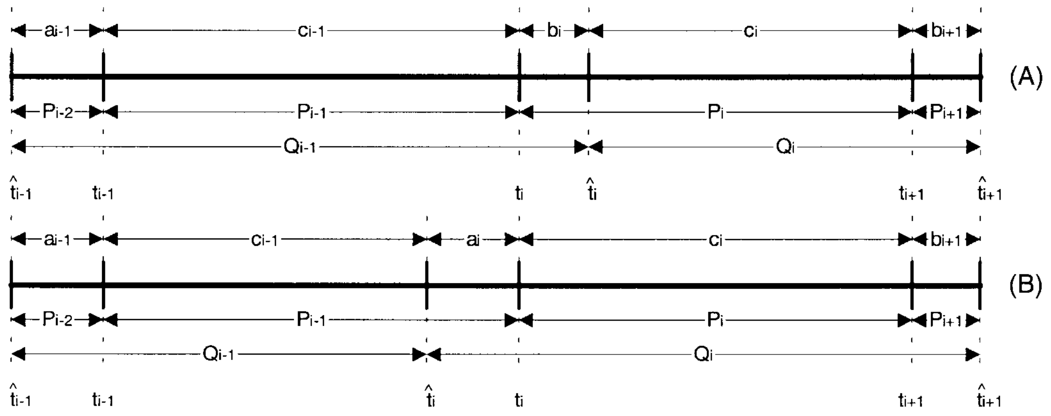

Fig. 12. Typical occurrence of event $\bar{F}$ and its effect on the estimated distributions. The dark line represents the time axis, that is partitioned into strings, whose lengths are noted above the dark line.The true distributions of the segments are noted by P and the distributions assigned to the hypothesized segments by the choice of $\hat{\mathcal{T}}$ are noted by Q. In diagram (A) $\hat{t}_i > t_i$ and in diagram (B) $\hat{t}_i < t_i$. In both diagrams $\hat{t}_{i-1} < t_{i-1}$ and $\hat{t}_{i+1} > t_{i+1}$.

vector c.

$$\begin{aligned} R_d(x^N;\mathcal{D}) &\triangleq \frac{1}{N}\log\frac{P(x^N \mid \Theta,\mathcal{T})}{Q(x^N \mid \hat{\Theta},\hat{\mathcal{T}})} \\ &= \frac{1}{N}\sum_{i=0}^{C}\log\frac{P_{i-1}(x_a^i)P_i(x_b^i)P_i(x_c^i)}{Q_i(x_a^i)Q_{i-1}(x_b^i)Q_i(x_c^i)} \\ &= \frac{1}{N}\sum_{i=0}^{C}\log\frac{P_{i-1}(x_a^i)P_i(x_b^i)}{Q_i(x_a^i)Q_{i-1}(x_b^i)} \\ &\quad + \frac{1}{N}\sum_{i=0}^{C}\log\frac{P_i(x_c^i)}{Q_i(x_c^i)} \\ &\triangleq \sum_{i=0}^{C}R_i + \sum_{i=0}^{C}r_i = \sum_{i=1}^{C}R_i + \sum_{i=0}^{C}r_i. \end{aligned} \tag{D.6}$$

The equality in the second line of (D.6) is obtained by definition of strings x_a^i, x_b^i, and x_c^i in (D.2). The string x_a^i is drawn by P_{i-1} but is assumed to be drawn by Q_i. The other two strings x_b^i and x_c^i are drawn by P_i, but are assumed to be drawn by Q_{i-1} and Q_i, respectively. The last equality is obtained by noticing that $x_a^0 = x_b^0 = \phi$. We summarize the pointwise and average bounds of r_i and R_i in the following lemma.

Lemma D.1:

$$r_i \le \frac{a_i + b_{i+1}}{N}\log e, \qquad 0 \le i \le C \tag{D.7}$$

$$R_i \le \frac{(a_i + b_i)\log N}{N}, \qquad 1 \le i \le C \tag{D.8}$$

$$\begin{aligned} \Pr(\bar{F})E[R_i \mid \bar{F}] \le \frac{2.5k}{N}[D(P_{i-1}\,\|\,P_i) + D(P_i\,\|\,P_{i-1})] \\ + o\left(\frac{k}{N}\right), \qquad 1 \le i \le C. \end{aligned} \tag{D.9}$$

The proof of the lemma is presented at the end of this section. Using (D.7) and noting that $a_0 = b_{C+1} = 0$, we obtain

$$\sum_{i=0}^{C} r_i \le \sum_{i=1}^{C}\frac{a_i + b_i}{N}\log e \le 2.5(\log e)C\frac{k}{N} \tag{D.10}$$

where the last inequality is obtained by the fact that for each i either a_i or b_i must be zero and by the definition of $\bar{F}$ that ensures that either must be bounded by $2.5k$. Similarly, we can show that

$$\sum_{i=1}^{C} R_i \le 2.5C\frac{k\log N}{N}. \tag{D.11}$$

Adding both bounds of the last two equations, we conclude the proof of the first equation of Proposition B.2. The proof of Proposition B.2 is concluded by simply adding the bounds for the PR and TR in (15) and (A.1), respectively, with $\hat{C} = C$, and taking $k = A\log N$, noticing the DR is the dominant term. Proposition B.3 is proved similarly to Proposition B.2 with one difference. Instead of taking the bound on R_i of (D.8), we take the upper bound of (D.9) for the average R_i over all transitions to obtain $D(\Theta)$. To conclude this section, we present the proof of Lemma D.1.

Proof of Lemma D.1: Equations (D.7) and (D.8) are proved by straightforward manipulations. Before proving (D.9) we present and prove another lemma. We can upper-bound r_i in the following manner:

$$\begin{aligned} r_i &\triangleq \frac{1}{N}\log\frac{P_i(x_c^i)}{Q_i(x_c^i)} && \text{(D.12)} \\ &= \frac{c_i}{N}\sum_{x\in\Sigma}P_c^i(x)\log\frac{P_i(x)}{Q_i(x)} && \text{(D.13)} \\ &\le \frac{c_i}{N}\sum_{x\in\Sigma}P_c^i(x)\log\frac{P_i(x)}{\frac{c_i}{A_i}P_i(x)} && \text{(D.14)} \\ &= \frac{c_i}{N}\log\frac{A_i}{c_i} = \frac{c_i}{N}\log\left(1 + \frac{a_i + b_{i+1}}{c_i}\right) && \text{(D.15)} \\ &\le \frac{a_i + b_{i+1}}{N}\log e. && \text{(D.16)} \end{aligned}$$

Equation (D.13) is obtained by representing the probability of vector x_c^i as the sum of the probabilities of its components. Using the definition of Q_i in (D.5) we obtain inequality (D.14). Then we use the fact that for all $x > 0$, $\log(1 + x) < x\log e$ to obtain inequality (D.16). This concludes the proof of (D.7).

We perform similar analysis to show that

$$\begin{aligned} R_i &\le \frac{a_i}{N}\log\frac{A_i}{a_i}+\frac{b_i}{N}\log\frac{A_i}{b_i} \\ &\le \frac{a_i+b_i}{N}\log N. \end{aligned} \tag{D.17}$$

The second inequality is obtained by taking $A_i \to N$ and discarding the denominators of the terms inside the logarithms, thus concluding the proof of (D.8).

Lemma D.2: Under the conditions of Theorem 3

$$\begin{aligned} \frac{1}{N}\log\frac{P_{i-1}(x_a^i)}{Q_i(x_a^i)} &\le \frac{1}{N}\log\frac{P_{i-1}(x_a^i)}{P_i(x_a^i)}+o\left(\frac{k}{N}\right) \\ \frac{1}{N}\log\frac{P_i(x_b^i)}{Q_{i-1}(x_b^i)} &\le \frac{1}{N}\log\frac{P_i(x_b^i)}{P_{i-1}(x_b^i)}+o\left(\frac{k}{N}\right). \end{aligned} \tag{D.18}$$

Proof of Lemma D.2: We prove the first inequality, but the same analysis can be performed to prove the second. We first upper-bound the expression $\log(P_{i-1}(x)/Q_i(x))$ for $x \in \Sigma$ where $P_{i+1}(x) > 0$.

$$\begin{aligned} &\log\frac{P_{i-1}(x)}{Q_i(x)} \\ &= \log\frac{P_{i-1}(x)}{\frac{1}{A_i}[a_iP_{i-1}(x)+c_iP_i(x)+b_{i+1}P_{i+1}(x)]} \\ &\le \frac{c_i}{A_i}\log\frac{P_{i-1}(x)}{P_i(x)}+\frac{b_{i+1}}{A_i}\log\frac{P_{i-1}(x)}{P_{i+1}(x)} \\ &\le \log\frac{P_{i-1}(x)}{P_i(x)}+o(1). \end{aligned} \tag{D.19}$$

The first inequality is obtained by Jensen's inequality, and the second by the assumption that transitions are more than $O(k)$ time points apart, thus $O(A_i) > O(b_{i+1})$. We now show that we can obtain the same bound when $P_{i+1}(x) = 0$. We define $B_i \triangleq a_i + c_i$.

$$\begin{aligned} \log\frac{P_{i-1}(x)}{Q_i(x)} &= \log\frac{\frac{A_i}{B_i}P_{i-1}(x)}{\frac{a_i}{B_i}P_{i-1}(x)+\frac{c_i}{B_i}P_i(x)} \\ &\le \log\frac{A_i}{B_i}+\frac{c_i}{B_i}\log\frac{P_{i-1}(x)}{P_i(x)} \\ &\le \log\frac{P_{i-1}(x)}{P_i(x)}+o(1). \end{aligned} \tag{D.20}$$

The first inequality is obtained by Jensen's inequality, and the second by the fact that $O(B_i) > O(b_{i+1})$, since transitions are more than $O(k)$ time points apart. We obtain the inequality by the well-known fact that if $x \to 0$, $\log(1+x) = O(x)$. Summing up for all terms of x_a^i we obtain that

$$\begin{aligned} \frac{1}{N}\log\frac{P_{i-1}(x_a^i)}{Q_i(x_a^i)} &= \frac{a_i}{N}\sum_{x\in\Sigma}P_a^i(x)\log\frac{P_{i-1}(x)}{Q_i(x)} \\ &\le \frac{a_i}{N}\sum_{x\in\Sigma}P_a^i(x)\left[\log\frac{P_{i-1}(x)}{P_i(x)}+o(1)\right] \\ &= \frac{1}{N}\log\frac{P_{i-1}(x_a^i)}{P_i(x_a^i)}+o\left(\frac{k}{N}\right) \end{aligned} \tag{D.21}$$

which concludes the proof of Lemma D.2.

To conclude the proof of Lemma D.1, we first extend the definition of a_i and b_i s.t.

$$a_i = b_i = 0, \qquad 1 \le i \le C, \quad \forall x^N \in F. \tag{D.22}$$

The respective strings x_a^i and x_b^i are defined as the null strings. We define $\alpha \triangleq 2.5k$, and for all i, s.t. $1 \le i \le C$, we make the following definitions:

$$\begin{aligned} u_i &\triangleq \alpha - a_i \\ v_i &\triangleq \alpha - b_i \\ x_u^i &\triangleq \left(x_{t_i-\alpha}, x_{t_i-\alpha+1}, \cdots, x_{t_i-a_i-1}\right) \\ x_v^i &\triangleq \left(x_{t_i+b_i}, x_{t_i+b_i+1}, \cdots, x_{t_i+\alpha-1}\right). \end{aligned} \tag{D.23}$$

The idea is to create the concatenated strings x_{ua}^i and x_{bv}^i, s.t. the first will contain all the α letters right before the ith true transition and the second the α letters right after the transition, for all $x^N \in \Sigma^N$. By using this notation, we can generalize the analysis for R_i and average over x^N. We conclude by showing the proof of (D.9). For convenience, we omit the superscript i from all vectors.

$$\Pr(\bar{F})E[R_i \mid \bar{F}]$$

$$= \frac{1}{N}\sum_{x^N\in\bar{F}}\Pr(x^N)\log\frac{P_{i-1}(x_a)P_i(x_b)}{Q_i(x_a)Q_{i-1}(x_b)} \tag{D.24}$$

$$\le \frac{1}{N}\sum_{x^N\in\bar{F}}\Pr(x^N)\log\frac{P_{i-1}(x_a)P_i(x_b)}{P_i(x_a)P_{i-1}(x_b)}+o\left(\frac{k}{N}\right) \tag{D.25}$$

$$\le \frac{1}{N}\sum_{x^N\in\bar{F}}\Pr(x^N)\left[\left|\log\frac{P_{i-1}(x_a)}{P_i(x_a)}\right| + \left|\log\frac{P_i(x_b)}{P_{i-1}(x_b)}\right|\right]+o\left(\frac{k}{N}\right) \tag{D.26}$$

$$\begin{aligned} \le \frac{1}{N}\sum_{x^N\in\Sigma^N}\Pr(x^N)\Bigg[&\left|\log\frac{P_{i-1}(x_a)}{P_i(x_a)}\right|+\left|\log\frac{P_i(x_b)}{P_{i-1}(x_b)}\right| \\ &+\left|\log\frac{P_{i-1}(x_u)}{P_i(x_u)}\right| \\ &+\left|\log\frac{P_i(x_v)}{P_{i-1}(x_v)}\right|\Bigg]+o\left(\frac{k}{N}\right) \end{aligned} \tag{D.27}$$

$$\begin{aligned} \le \frac{1}{N}\sum_{x^N\in\Sigma^N}\Pr(x^N)\Bigg[&\log\frac{P_{i-1}(x_a)}{P_i(x_a)}+\frac{2\log e}{e}\frac{P_i(x_a)}{P_{i-1}(x_a)} \\ &+\log\frac{P_i(x_b)}{P_{i-1}(x_b)}+\frac{2\log e}{e}\frac{P_{i-1}(x_b)}{P_i(x_b)} \\ &+\log\frac{P_{i-1}(x_u)}{P_i(x_u)}+\frac{2\log e}{e}\frac{P_i(x_u)}{P_{i-1}(x_u)} \\ &+\log\frac{P_i(x_v)}{P_{i-1}(x_v)}+\frac{2\log e}{e}\frac{P_{i-1}(x_v)}{P_i(x_v)}\Bigg]+o\left(\frac{k}{N}\right) \end{aligned} \tag{D.28}$$

$$\begin{aligned} = \frac{1}{N}&\sum_{x_{ua}\in\Sigma^\alpha}P_{i-1}(x_{ua})\log\frac{P_{i-1}(x_{ua})}{P_i(x_{ua})} \\ &+\frac{1}{N}\sum_{x_{bv}\in\Sigma^\alpha}P_i(x_{bv})\log\frac{P_i(x_{bv})}{P_{i-1}(x_{bv})}+o\left(\frac{k}{N}\right) \end{aligned} \tag{D.29}$$

$$= \frac{2.5k}{N}[D(P_{i-1}\,\|\,P_i)+D(P_i\,\|\,P_{i-1})]+o\left(\frac{k}{N}\right). \tag{D.30}$$

Inequality (D.25) is obtained by Lemma D.2. We then upper-bound the two logarithmic terms by their absolute values to obtain (D.26), and add nonnegative terms for (D.27). The bound $|\log x| \leq 2(\log e/e)x^{-1} + \log x$ (see [5]) is then used to obtain (D.28). Finally, we sum over all data letters independent of x_{ua} and x_{bv}, and realize that the distributions of x_{ua} and x_{bv} are P_{i-1} and P_i, respectively (in (D.29)). This reduces the nonlogarithmic terms of (D.28), which sum up to a term of $O(1/N)$, and results in the sum of divergences in (D.30), multiplied by the length of the strings $2.5k$, therefore concluding the proof of (D.9).

Appendix E
Proof of Theorem 4

To prove Theorem 4, we first need to upper-bound the number of blocks B. We then use this bound to upper-bound both the PR and the DR. By definition of the scheme

$$\begin{aligned} N &\geq 1 + \sum_{b=2}^{B-1} \lfloor \alpha b \log b \rfloor \geq \sum_{b=1}^{B-1} [\alpha b \log b - 1] \\ &\geq \alpha \int_0^{B-1} x \log x \, dx - B \\ &= \frac{\alpha \log e}{2} [x^2(\ln x - 0.5)]_0^{B-1} - B \\ &= \frac{\alpha}{2} \left[(B-1)^2 \log \frac{B-1}{\sqrt{e}} \right] - B \\ &= \frac{\alpha}{2} B^2 \log B - O(B^2). \end{aligned} \tag{E.1}$$

Therefore,

$$N + O(B^2) \geq \frac{\alpha}{2} B^2 \log B. \tag{E.2}$$

To satisfy the last equation, we must have

$$B \leq \sqrt{\frac{4}{\alpha}} \sqrt{\frac{N}{\log N}} + o\left(\sqrt{\frac{N}{\log N}} \right). \tag{E.3}$$

As a result of the discussion before (15), each block of length m_b, coded by the KT estimate, produces

$$[(r-1)/2] \log m_b + O(1)$$

extra PR bits. Hence, we can upper-bound the PR by

$$\begin{aligned} R_p(x^N; \mathcal{P}) &\leq \frac{r-1}{2N} \sum_{b=1}^{B} [\log \lfloor \alpha b \log b \rfloor + O(1)] && \text{(E.4)} \\ &\leq \frac{r-1}{2N} \left[\sum_{b=1}^{B} \log(\alpha b \log b) + O(B) \right] && \text{(E.5)} \\ &= \frac{r-1}{2N} \Bigg[B \log \alpha + \log(B!) \\ &\qquad + \sum_{b=1}^{B} \log \log b + O(B) \Bigg] && \text{(E.6)} \\ &\leq \frac{r-1}{2N} [B \log B + O(B \log \log B)] \\ &= \frac{r-1}{2\sqrt{\alpha}} \sqrt{\frac{\log N}{N}} + o\left(\sqrt{\frac{\log N}{N}} \right). && \text{(E.7)} \end{aligned}$$

Equation (E.6) is obtained by opening the expression inside the logarithm, and inequality (E.7) by the fact that $B! \leq B^B$.

It is obvious from (17) that the largest DR is obtained for PSMS's with true transitions at the midpoints of the last C blocks. For these PSMS's we can upper-bound the DR by

$$\begin{aligned} R_d(x^N; \mathcal{P}) &\leq C \frac{m_B}{N} \leq C\alpha \frac{B \log B}{N} \\ &= C\sqrt{\alpha} \sqrt{\frac{\log N}{N}} + o\left(C \sqrt{\frac{\log N}{N}} \right). \end{aligned} \tag{E.8}$$

We conclude the proof of Theorem 4 by summing up the last two upper bounds.

Acknowledgment

The authors gratefully acknowledge Prof. J. Ziv, who served as the first author's M.Sc. co-adviser, for his helpful suggestions. The authors also wish to thank a reviewer for his valuable comments.

References

[1] R. E. Blahut, *Principles and Practice of Information Theory*. Reading, MA: Addison-Wesley, 1991.

[2] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[3] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. New York: Academic, 1981.

[4] R. G. Gallager, "Variations on a theme by Huffman," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 668–674, Nov. 1978.

[5] R. M. Gray, *Entropy and Information Theory*. New York: Springer-Verlag, 1990.

[6] M. Gutman, "Asymptotically optimal classification for multiple tests with empirically observed statistics," *IEEE Trans. Inform. Theory*, vol. 35, pp. 401–408, Mar. 1989.

[7] P. G. Howard and J. S. Vitter, "Arithmetic coding for data compression," *Proc. IEEE*, vol. 82, pp. 857–865, June 1994.

[8] F. Jelinek, *Probabilistic Information Theory*. New York: McGraw-Hill, 1968, pp. 476–489.

[9] D. E. Knuth, "Dynamic Huffman coding," *J. Algorithms*, vol. 6, pp. 163–180, June 1985.

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

[11] G. Louchard and W. Szpankowski, "On the average redundancy rate of the Lempel–Ziv code," *IEEE Trans. Inform. Theory*, vol. 43, pp. 2–8, Jan. 1997.

[12] N. Merhav, "On the minimum description length principle for sources with piecewise constant parameters," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1962–1967, Nov. 1993.

[13] N. Merhav and M. Feder, "A strong version of the redundancy-capacity theorem of universal coding," *IEEE Trans. Inform. Theory*, vol. 41, pp. 714–722, May 1995.

[14] E. Plotnik, M. J. Weinberger, and J. Ziv, "Upper bounds on the probability of sequences emitted by finite-state sources and on the redundancy of the Lempel-Ziv algorithm," *IEEE Trans. Inform. Theory*, vol. 38, pp. 66–72, Jan. 1992.

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

[16] S. A. Savari, "Redundancy of the Lempel-Ziv incremental parsing rule," *IEEE Trans. Inform. Theory*, vol. 43, pp. 9–21, Jan. 1997.

[17] Y. M. Shtarkov, T. J. Tjalkens, and F. M. J. Willems, "Multi-alphabet universal coding of memoryless sources," *Probl. Inform. Transm.*, vol. 31, No. 2, pp. 20–35, Apr.–June 1995.

[18] J. S. Vitter, "Design and analysis of dynamic Huffman codes," *J. Assoc. Comput. Mach.*, vol. 34, pp. 825–845, Oct. 1987.

[19] F. M. J. Willems, "Coding for binary piecewise memoryless sources," in *Proc. Japan–Benelux Workshop*, 1994.

[20] F. M. J. Willems, "Coding for a binary independent piecewise-identically-distributed source," *IEEE Trans. Inform. Theory*, vol. 42, pp. 2210–2217, Nov. 1996.

[21] F. M. J. Willems and F. Casadei, “Weighted coding methods for binary piecewise memoryless sources,” in *Proc. 1995 IEEE Int. Symp. Information Theory* (Whistler, BC, Canada, Sept. 17–22, 1995).

[22] F. M. J. Willems and M. Krom, “Live-and-die coding for binary piecewise i.i.d. sources,” in *Proc. 1997 IEEE Int. Symp. Information Theory* (Ulm, Germany, June 29–July 4, 1997), p. 68.

[23] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, “The context-tree weighting method: Basic properties,” *IEEE Trans. Inform. Theory*, vol. 41, pp. 653–664, May 1995.

[24] A. D. Wyner and J. Ziv, “The sliding-window Lempel-Ziv algorithm is asymptotically optimal,” *Proc. IEEE*, vol. 82, pp. 872–877, June 1994.

[25] H. Yokoo, “An improvement of dynamic Huffman coding with a simple repetition finder,” *IEEE Trans. Commun.*, vol. 39, pp. 8–10, Jan. 1991.

[26] J. Ziv, “On classification with empirically observed statistics and universal data compression,” *IEEE Trans. Inform. Theory*, vol. 34, pp. 278–286, Mar. 1988.

[27] J. Ziv and A. Lempel, “A universal algorithm for sequential data compression,” *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 337–343, May 1977.

[28] ——, “Compression of individual sequences via variable-rate coding,” *IEEE Trans. Inform. Theory*, vol IT-24, pp. 530–536, Sept. 1978.