

Lossless coding for distributed streaming sources

Stark C. Draper, Cheng Chang, and Anant Sahai

August 5, 2010

Abstract

Distributed source coding is traditionally viewed in a block coding context where all the source symbols are known in advance at the encoders. This paper argues that to develop a variable-length notion of distributed source coding, one should consider a streaming data model. In the streaming setting source symbol pairs are revealed to separate encoders in real time and need to be reconstructed at the decoder with some tolerable end-to-end delay. A causal sequential random binning encoder is introduced and paired with maximum likelihood and universal decoders. The latter uses a novel “weighted empirical suffix entropy” decoding rule. We derive a lower bounds on the error exponent with delay for each decoder. We show that, in fact, both decoders achieve the same positive error exponents for all rate pairs inside the Slepian-Wolf achievable rate region. The dominant error events in streaming are different from those in block-coding and result in different exponents. Because the sequential random binning scheme is also universal over delays, the resulting code eventually reconstructs every source symbol correctly with probability one.

1 Introduction

Traditionally, “lossless” coding is considered using two distinct paradigms: fixed-length block coding and variable-length coding. In both settings source symbols are known in advance at the encoder and must be mapped into strings of bits to be decoded by the receiver. Fixed-length block coding accepts a small probability of error and outputs a fixed-length bit string. On the other hand, in exchange for attaining a zero probability of error, variable-length approaches guarantee only an *expected* length of the outputted bit-string. In the point-to-point setting, both paradigms apply generically. In contrast, distributed source coding has traditionally been explored only within the block context. In their 1973 paper [16], Slepian and Wolf even ask:

“What is the theory of variable-length encodings for correlated sources?”

The objective of this paper is to pose one nontrivial answer to this question.

In the classical context of source realizations known entirely in advance, we argue that the answer to Slepian and Wolf’s question is simple: there is no nontrivial sense of variable-length

S. C. Draper is with the Department of Electrical and Computer Engineering, University of Wisconsin Madison, Madison, WI 53706 (E-mail: sdraper@ece.wisc.edu).

C. Chang was with the Department of Electrical Engineering and Computer Science, University of California Berkeley, Berkeley, CA 94720. He is now with D. E. Shaw, New York, NY (E-mail: cchang@eecs.berkeley.edu).

A. Sahai is with the Department of Electrical Engineering and Computer Science, University of California Berkeley, Berkeley, CA 94720 (E-mail: sahai@eecs.berkeley.edu).

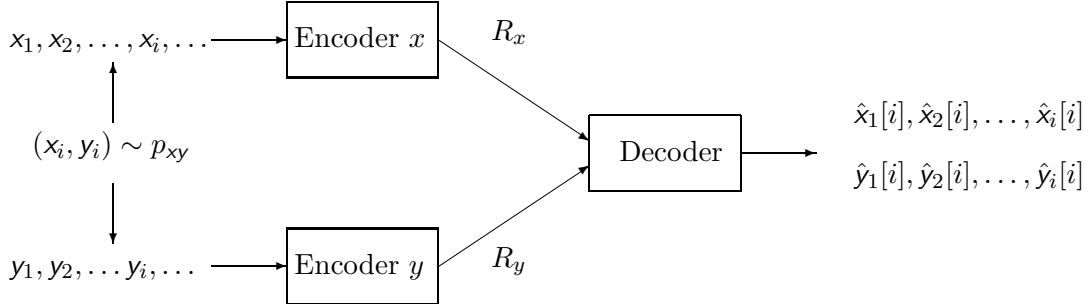


Figure 1: Streaming distributed source coding. At time i a source pair (x_i, y_i) is received at the encoder, R_x and R_y bits are sent by the respective encoders to the joint decoder, and an estimated of the j th pair $(\hat{x}_j[i], \hat{y}_j[i])$, for all $j \leq i$, is made. The delay on this estimate is $\Delta = i - j$. We bound the individual and joint error probabilities as a function of source statistics, R_x , R_y , and Δ .

encoding that applies generically while still being interesting at sum rates close to the joint source entropy rate. This is easiest to see by example. Suppose that the first encoder observes a random vector x^N , a sequence of N independent and identically distributed (i.i.d.) uniform binary random variables. Suppose that the second encoder observes y^N , which is related to x^N via a memoryless binary symmetric channel with crossover probability $p < 0.5$. The Slepian-Wolf sum-rate bound is $H(x, y) = 1 - p \log p - (1 - p) \log(1 - p) < 2 = H(x) + H(y)$. Since the individual encoders only see uniformly distributed binary sources, they cannot detect when the source pair is behaving *jointly* atypically, which is when a variable-rate code would map the source into an above-expected-length encoding. Thus, the individual encoders have no basis on which to adjust their encoding rates to combat jointly atypical behavior. Next observe that since all pairs of binary sequences are possible, and all individual sequences are equally likely, to achieve zero error each encoder must use distinct bit-strings to encode each source realization. Thus, since the expected length of each encoding can only depend on the uniform marginal distributions, the result must be that to attain zero error the expected length of each encoding must be at least N . Variable-length approaches therefore do not lead to zero-error Slepian-Wolf codes for generic sources at interesting rate-points (a rate pair $(R_x, R_y) = (\log |\mathcal{X}|, \log |\mathcal{Y}|)$ not being very interesting). One should note, however, that just as is the case for zero-error channel coding, when certain symbol pairs are known to have zero probability, there are special cases where zero-error Slepian-Wolf coding is possible [12].

Another distinct view of variable-length coding is as a tool that enables us to achieve meaningful compression when we lack a probabilistic characterization of the source. Variable-length flexibility can allow the rate to be adapted in an on-line manner to the unknown statistics. If a low-rate feedback link is available from the decoder to the two separate encoders, then this sense of variable-length Slepian-Wolf coding is possible (see, e.g., [15, 7, 8, 18]). In, e.g., [7] such fixed-to-variable length distributed source coding schemes are presented for lossless (and lossy) compression in which the stopping-time is chosen at the decoder and communicated back to the encoders over a low-rate feedback link. The goal of [7] is not achieving a zero probability of error — rather a very small probability of error is acceptable, in exchange for using a rate that is as small as possible.

1.1 Streaming distributed source coding and reconstruction delay

To answer the question posed by Slepian and Wolf we argue that we must consider a “streaming” version of Slepian-Wolf coding, our model of which is illustrated in Figure 1. The streaming setting models sources as being embedded in time, integrating the idea that all physically realizable encoders/decoders must obey some form of causality, and allows us to discern in a distributed setting a system-level analog to variable-length source coding. In particular, we eliminate the assumption that encoders have access to the entire source realization in advance and instead assume that source symbols continue to arrive at the encoder during the course of transmission. Within this model we want a probability of error that goes to zero for every source symbol, but at the cost of variable delay.

To see why the streaming setting is an appropriate model for investigation, and to justify the notion of variable delay in a streaming setting, first consider a simpler point-to-point streaming system – the system of Figure 1 with only a single source – when the communication rate exceeds the source entropy. The objective is a classic one – to convey a causally realized streaming source to the decoder over a reliable, but finite-rate communication link. First, consider using a sequence of fixed-rate block codes to compress successive sub-strings of the realized source sequence. After compression the resulting encodings are enqueued for transmission across the fixed-rate bit pipe. Since the source entropy rate is below the data-rate, errors will be rare, and the queue will be stable. Errors occur when sufficiently atypical source strings appear that cannot be compressed using the block code. With this implementation, the probability of error is fixed at design time and the end-to-end delay is constant.

In contrast, consider instead the use of variable-length source codes to compress the sub-strings. In this situation there are no decoding errors, but the use of variable-length codes results in a variable end-to-end delay. The more unlikely the realized source sequence, the longer the delay experienced. Thus, while *asymptotically* there are no errors when variable-length source codes are used (assuming an infinite buffer size), the delay till a given symbol can be recovered depends on the random source realizations around it. Because atypical source realizations are large-deviation events, the probability that some source symbol cannot be reconstructed Δ samples after it enters the encoder decays exponentially in Δ .

We now combine two of our observations. The first was that there is no nontrivial sense of variable-length Slepian-Wolf coding when fixed-length sources sequences are realized in advance. The second was that in streaming systems variable-length coding results in variable delay and there is an exponential decay in delay on the probability of not being able to recover correctly any particular symbol. These observations motivate us to ask whether a streaming data system that exhibits an exponential decay in delay on incorrect recovery can be realized in a generic distributed coding context.

The system we design in this paper has just such a characteristic. In particular, the probability of error on any specific symbol goes to zero in delay. The choice of acceptable delay is up to the designer. Essentially, every source symbol can eventually be recovered correctly (i.e., as Δ gets large) with probability one. There is an important distinction to be made between our results and those of the point-to-point setting with variable-length coding discussed above. While in that setting the exponential decay is on not being able to *recover* any specific symbol, in our setting it is on not being able to *recover correctly* any specific symbol. In the point-to-point setting the decoder can tell when each symbol is recovered (due, e.g., to the use of prefix codes), and up to that time the system knows it hasn’t yet recovered that symbol. On the other hand, in the system

we design the decoder will not know when the estimate for a particular symbol has converged to its final value. Rather, at any time the decoder can make a causal estimate of any specific symbol. The decoder can continue to refine these estimates, but the estimates can be wrong. The bound that decays exponentially in delay is on the probability that each such estimate is incorrect.

In this paper, we formally define a streaming Slepian-Wolf code, and develop coding strategies both for situations when source statistics are known and when they are not. The new tool we introduce is a sequential binning argument that parallels the tree-coding arguments used to study convolutional codes. We characterize the performance of the streaming schemes through an error-exponent analysis and demonstrate the same exponents can be achieved regardless of whether the system is informed of the source statistics (in which case we use maximum-likelihood decoding) or not (in which case we use universal decoding). The universal decoder we design for the streaming problem is somewhat different from those familiar from the block coding literature, as are the nature of the error exponents in both cases.

From an engineering perspective, four desirable qualities of our scheme would be: (i) low-rate transmission, (ii) small end-to-end latency, (iii) low probability of error, and (iv) low implementational complexity. As is often the case in information theoretic investigations, we will not consider implementation complexity. The theory we develop does tell us about the tradeoffs among the first three of these qualities. In Figure 2 we illustrate the tradeoff between rate, latency, and error probability that is revealed by our analysis for a bursty discrete memoryless source. For simplicity we plot results for a point-to-point streaming system which, as already mentioned, can be understood as the system illustrated in Figure 1 in the special case where y is a zero-entropy source. In the example, with probability 0.5 the realization of each i.i.d. source symbols x_i is 0 and with probability 0.5 the realization is uniformly distributed across $\{1, 2, 3, 4\}$. The entropy of this source is 2 bits. The surface plotted in the figure depicts the upper bound on achievable error probability derived in this paper as a function of rate and delay.

1.2 Outline

In Section 2 we review classic results on error exponents for fixed-block Slepian-Wolf source coding. In Section 3 we state the main result of the paper on error exponents for streaming Slepian-Wolf source coding and connect back to the form of the block coding exponents provided in Section 2. In Section 4 we present illustrative numerical results, including more detailed discussion of the example of Figure 2. The theorems of Section 3 are proved in Sections 5 and 6. Section 5 begins by deriving results for point-to-point streaming source coding. This is the simplest case and provides insights into the nature of sequential source coding problem and associated error events. We show that the streaming error exponent is the same as the random block source coding error exponent and in Section 5.5 consider point-to-point streaming source coding when side-information is available at the decoder. In Section 6 we present the proof of the main result of the paper on the error exponents of distributed streaming source coding for correlated sources. For all three scenarios, point-to-point source coding, decoding with side-information, and distributed source coding, both maximum likelihood (ML) and universal decoding rules are studied. In the appendix we show that the error exponents achieved by the ML and universal decoders are, in fact, the same.

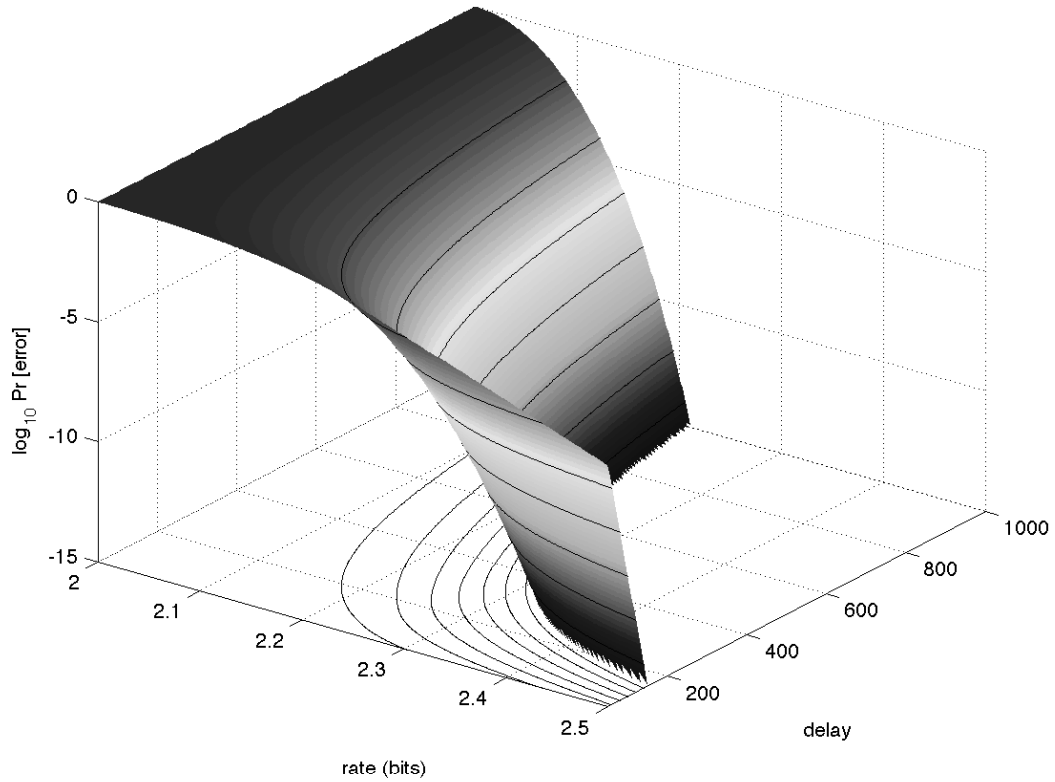


Figure 2: The achievable tradeoff between rate, delay, and probability of error for an i.i.d. source that has a 50% chance of emitting a 0 and a 12.5% chance of emitting a 1, 2, 3, or a 4. The entropy is 2 bits. The surface represents what the bounds in this paper achieve.

1.3 Notation

We use serified-fonts, e.g., x to indicate sample values, and sans-serif, e.g., \mathbf{x} , to indicate random variables. Bolded fonts are reserved to indicate sample or random vectors, e.g., $\mathbf{x} = x^n$ and $\mathbf{x} = \mathbf{x}^n$, respectively, where the vector length (n here) is understood from the context. Subsequences, e.g., x_l, x_{l+1}, \dots, x_n are denoted as x_l^n where $x_l^j \triangleq \emptyset$ if $i > j$. Distributions are indicated with lower-case p , e.g., \mathbf{x} is distributed according to $p_{\mathbf{x}}(x)$. We use script font to denote sets, $\mathcal{X}, \mathcal{F}, \mathcal{W}$, etc., their cardinality by, e.g., $|\mathcal{X}|$, and reserve \mathcal{E} and \mathcal{D} to denote encoding and decoding functions, respectively. We use standard notation for types, see, e.g., [6]. Let $N(a; \mathbf{x})$ denote the number of symbols in the length- n vector \mathbf{x} that take on value a . Then, \mathbf{x} is of type P if $P(a) = N(a; \mathbf{x})/n$. The type-class, or set of length- n vectors of type P is denoted \mathcal{T}_P . A sequence \mathbf{y} has conditional type V given \mathbf{x} if $N(a, b; \mathbf{x}, \mathbf{y}) = N(a; \mathbf{x})V(b|a) = P(a)V(b|a)$ for every a, b . The set of sequences \mathbf{y} having conditional type V with respect to \mathbf{x} is called the V -shell of \mathbf{x} and is denoted by $\mathcal{T}_V(\mathbf{x})$. When considered together, the pair (\mathbf{x}, \mathbf{y}) is said to have joint type $V \times P$. We always use upper-case, e.g., P and V , to denote length- n types and conditional types. As we often discuss the types of subsequences we add a superscript notation to remind the reader of the length of the subsequence in question. If, for instance, the subsequence under consideration is x_l^n we write $x_l^n \in \mathcal{T}_{P^{n-l}}$. Similarly we use V^{n-l} for the conditional type of length- $(n-l+1)$, and $V^{n-l} \times P^{n-l}$ for the joint type. Given a joint type $V \times P$, entropies and conditional entropies are denoted as $H(P)$ and $H(V|P)$, respectively. Alternately, the empirical joint entropy of a pair of sequences (x^n, y^n) is denoted $H(x^n, y^n)$. The entropy of a Bernoulli- p distribution is denoted as $H_B(p)$. Generally we assume the natural-base for our logarithms, expressing entropies in nats. The one exception is in Section 4 where we use bits since one of our prominent examples is binary. The Kullback Leibler (KL) divergence between two distributions q and p is denoted by $D(q||p)$. Finally, $|\cdot|^+$ is used as shorthand to denote $\max(\cdot, 0)$.

2 Background results

In this section we review classical definitions and error exponent results for distributed block coding. In later sections we refer back to these results to contrast them with the results from the streaming framework.

In the classic block-coding Slepian-Wolf paradigm, length- N vectors \mathbf{x} and \mathbf{y} are observed by their respective encoders before communication commences. In this situation a rate- (R_x, R_y) length- N block source code consists of an encoder-decoder triplet $(\mathcal{E}_N^x, \mathcal{E}_N^y, \mathcal{D}_N)$:

Definition 1 *A randomized length- N rate- (R_x, R_y) block encoder-decoder triplet $(\mathcal{E}_N^x, \mathcal{E}_N^y, \mathcal{D}_N)$ is a set of maps*

$$\begin{aligned} \mathcal{E}_N^x &: \mathcal{X}^N \rightarrow \{0, 1\}^{NR_x}, & e.g., \quad \mathcal{E}_N^x(x^N) &= a^{NR_x} \\ \mathcal{E}_N^y &: \mathcal{Y}^N \rightarrow \{0, 1\}^{NR_y}, & e.g., \quad \mathcal{E}_N^y(y^N) &= b^{NR_y} \\ \mathcal{D}_N &: \{0, 1\}^{NR_x} \times \{0, 1\}^{NR_y} \rightarrow \mathcal{X}^n \times \mathcal{Y}^n, & e.g., \quad \mathcal{D}_N(a^{NR_x}, b^{NR_y}) &= (\hat{x}^N, \hat{y}^N) \end{aligned}$$

where common randomness, shared between the encoders and the decoder is assumed. This allows us to randomize the mappings independently of the source sequences.

While we state Definition 1 only for Slepian-Wolf coding, it immediately specializes to source coding with decoder side information (dropping the \mathcal{E}_N^y and revealing y^N to the decoder), and source coding without side information (dropping the \mathcal{E}_N^y).

The standard error probability considered in Slepian-Wolf coding is the joint error probability, $\Pr[(x^N, y^N) \neq (\hat{x}^N, \hat{y}^N)] = \Pr[(x^N, y^N) \neq \mathcal{D}_N(\mathcal{E}_N^x(x^N), \mathcal{E}_N^y(y^N))]$. In this paper we also consider the marginal error events $\Pr[x^N \neq \hat{x}^N]$ and $\Pr[y^N \neq \hat{y}^N]$. Distinguishing between these events is of interest in applications where \mathbf{x} and \mathbf{y} are decoded jointly, but used individually. All probabilities are taken over the random source vectors as well as the randomized mappings. A joint error exponent E is said to be achievable if there exists a family of rate- (R_x, R_y) encoders and decoders $\{(\mathcal{E}_N^x, \mathcal{E}_N^y, \mathcal{D}_N)\}$, indexed by N , such that

$$\lim_{N \rightarrow \infty} -\frac{1}{N} \log \Pr[(x^N, y^N) \neq (\hat{x}^N, \hat{y}^N)] \geq E. \quad (1)$$

Similarly, a marginal exponent E is achievable for source x^N if

$$\lim_{N \rightarrow \infty} -\frac{1}{N} \log \Pr[x^N \neq \hat{x}^N] \geq E. \quad (2)$$

In this paper, we study random source vectors (\mathbf{x}, \mathbf{y}) that are i.i.d. across time but may have dependencies at any given time:

$$p_{\mathbf{x}, \mathbf{y}}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^N p_{x, y}(x_i, y_i).$$

For such i.i.d. sources, upper and lower bounds on the achievable error exponents are derived in [10, 11, 6]. These results are summarized by the following theorems.

Theorem 1 *Given rate pair (R_x, R_y) , there exists a randomized encoder-decoder triplet (per Definition 1) that satisfy the following three decoding criteria:*

(i) *For all $E < E_{bl,x}(R_x, R_y)$, there is a constant $K > 0$ such that $\Pr[\hat{x}^N \neq x^N] \leq K \exp\{-NE\}$ where*

$$E_{bl,x}(R_x, R_y) = \min \left\{ \sup_{0 \leq \rho \leq 1} E_{x|y}(R_x, \rho), \sup_{0 \leq \rho \leq 1} E_{xy}(R_x, R_y, \rho) \right\}. \quad (3)$$

(ii) *For all $E < E_{bl,y}(R_x, R_y)$ there is a constant $K > 0$ such that $\Pr[\hat{y}^N \neq y^N] \leq K \exp\{-NE\}$ where*

$$E_{bl,y}(R_y) = \min \left\{ \sup_{0 \leq \rho \leq 1} E_{y|x}(R_y, \rho), \sup_{0 \leq \rho \leq 1} E_{xy}(R_x, R_y, \rho) \right\}. \quad (4)$$

(iii) *For all $E < E_{bl,xy}(R_x, R_y)$ there is a constant $K > 0$ such that $\Pr[(\hat{x}^N, \hat{y}^N) \neq (x^N, y^N)] \leq K \exp\{-NE\}$ where*

$$E_{bl,xy}(R_x, R_y) = \min \left\{ \sup_{0 \leq \rho \leq 1} E_{x|y}(R_x, \rho), \sup_{0 \leq \rho \leq 1} E_{y|x}(R_y, \rho), \sup_{0 \leq \rho \leq 1} E_{xy}(R_x, R_y, \rho) \right\}. \quad (5)$$

In the above,

$$E_{xy}(R_x, R_y, \rho) = \rho(R_x + R_y) - \log \left[\sum_{x,y} p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \quad (6)$$

$$E_{x|y}(R_x, \rho) = \rho R_x - \log \left[\sum_y \left[\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right] \quad (7)$$

$$E_{y|x}(R_y, \rho) = \rho R_y - \log \left[\sum_x \left[\sum_y p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right]. \quad (8)$$

As long as (R_x, R_y) is in the interior of the achievable Slepian-Wolf region, i.e., $R_x > H(x|y)$, $R_y > H(y|x)$ and $R_x + R_y > H(x, y)$, cf. [16, 5], all the above exponents are positive. Upper bounds on the error exponents are provided in [6], and match the lower bounds when the rate pair (R_x, R_y) is within, but close to the boundary of, the achievable region. This is analogous to the high-rate regime in channel coding where the random coding and sphere-packing bounds match.

Theorem 1 can be used to generate bounds on the exponent for source coding with decoder side information (i.e., \mathbf{y} observed at the decoder), and for source coding without side information (i.e., \mathbf{y} is a constant). These corollaries will serve as a basis for comparison as we build toward the complete solution for streaming Slepian-Wolf systems.

Corollary 1 *Consider a Slepian-Wolf problem where \mathbf{y} is known by the decoder. Given a rate R_x , then for all*

$$E < \sup_{0 \leq \rho \leq 1} \rho R_x - \log \left[\sum_y \left[\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right] \quad (9)$$

there exists a family of randomized encoder-decoder mappings as defined in Definition 1 such that (2) is satisfied.

The proof of Corollary 1 follows from Theorem 1 by letting R_y be arbitrarily large. Note that the exponent in (9) is identical to $E_{x|y}(R_x, \rho)$ in (6), which given an operational meaning to that exponent. That exponent bounds the event that \mathbf{x} is decoded incorrectly while \mathbf{y} is decoded correctly.

Next let \mathbf{y} be deterministic, e.g., $p_{x,y}(x, y) = p_{x|y}(x|y)1[y = a]$ for some $a \in \mathcal{Y}$ where $1[\cdot]$ is the indicator function. Then it follows that $H(x) = 0$, $H(x|y) = H(x)$ and, specializing the form of $E_{x|y}(R_x, \rho)$ to this distribution, we get the following random-coding bound for the point-to-point case of a single source \mathbf{x} .

Corollary 2 *Consider a Slepian-Wolf problem where \mathbf{y} is deterministic, i.e., $\mathbf{y} = \mathbf{y}$. Given a rate R_x , then for all*

$$E < \sup_{0 \leq \rho \leq 1} \rho R_x - \log \left[\sum_x p_x(x)^{\frac{1}{1+\rho}} \right]^{1+\rho} \quad (10)$$

there exists a family of randomized encoder-decoder triplet as defined in Definition 1 such that (2) is satisfied.

Gallager [10] and Koshelev [11] initiated the study of the error exponents of ML decoding for Slepian-Wolf systems, Gallager for source coding with decoder side information, and Koshelev for the two-encoder Slepian-Wolf problem. The joint decoding bound (5) is from [11] where (in the case of ML decoding considered therein) the constant $K = 1$. Koshelev did not consider the marginal exponents (3) and (4), but those can be extracted immediately from his derivation. As might be guessed from the discussion following Corollary 1, Koshelev partitions the joint error event (1) into three constituent events: (a) both \hat{x}^N and \hat{y}^N are erroneous, (b) only \hat{x}^N is erroneous, (c) only \hat{y}^N is erroneous. Respectively, the exponents bounding each of these events are given in (6)–(8). By ignoring either of the latter two events one get the marginal error bounds. For example, ignoring event (c) and accounting for events (a) and (b) leads to a bound on the event that only \hat{x}^N is erroneous, and to the error exponent of (3).

It is well known in the literature [6] that the results of Theorem 1 and Corollaries 1 and 2 can be achieved by universal decoders as well as ML decoders. Universal decoding results are often derived

using the methods of types, e.g., in [6]. The “Csiszar-style” exponents of [6] take a different form from the “Gallager-style” form of the exponents given in this section, due to the use of type-based arguments. The equivalence of the two forms of the exponents for these problems is a classic result. See, e.g., [6, pg. 44] exercise 13 and [6, pg. 192] exercise 23.

3 Main Results

In this section we present the main results of the paper. We define streaming source coding for both point-to-point and distributed systems. We present results for both maximum likelihood (ML) and universal decoding. The error exponents achieved are equal for both. We compare the forms of the streaming exponents with their block coding counterparts and in Section 4 illustrate the differences through numerical examples. Proofs of the results are provided in Sections 5 and 6, while we defer to the appendices proofs not needed to understand the fundamental differences between block and streaming coding.

3.1 Code definitions and error events for streaming systems

We start by defining a sequential random binning encoder/decoder for a streaming system. As for block coding this definition can immediately be specialized to source coding with decoder side information and point-to-point source coding without side information.

Definition 2 *A randomized sequential encoder-decoder triplet $(\{\mathcal{E}_j^x\}, \{\mathcal{E}_j^y\}, \{\mathcal{D}_j\})$ is a sequence of mappings, $\{\mathcal{E}_j^x\}, j = 1, 2, \dots$, $\{\mathcal{E}_j^y\}, j = 1, 2, \dots$ and $\{\mathcal{D}_j\}, j = 1, 2, \dots$ such that*

$$\begin{aligned} \mathcal{E}_j^x &: \mathcal{X}^j \longrightarrow \{0, 1\}^{R_x}, \quad \text{e.g., } \mathcal{E}_j^x(x^j) = a_{(j-1)R_x+1}^{jR_x}, \\ \mathcal{E}_j^y &: \mathcal{Y}^j \longrightarrow \{0, 1\}^{R_y}, \quad \text{e.g., } \mathcal{E}_j^y(y^j) = b_{(j-1)R_y+1}^{jR_y}. \end{aligned} \tag{11}$$

Common randomness, shared between encoders and decoder, is assumed. This allows us to randomize the mappings independently of the source sequence.

The decoder mapping

$$\begin{aligned} \mathcal{D}_j &: \{0, 1\}^{jR_x} \times \{0, 1\}^{jR_y} \longrightarrow \hat{\mathcal{X}}^j \times \hat{\mathcal{Y}}^j, \text{ e.g.,} \\ \mathcal{D}_j(a^{jR_x}, b^{jR_y}) &= (\hat{x}^j(j), \hat{y}^j(j)). \end{aligned}$$

At each time j the decoder \mathcal{D}_j outputs estimates of all the source symbols that have entered the encoder by time j .

Note that sometimes we will allow an extra “failure” symbol “?” so that $\hat{\mathcal{X}} = \mathcal{X} \cup \{?\}$.

In this paper, the sequential encoding maps will always work by assigning random “parity bits” in a causal manner to the observed source sequence. That is, the R_x (or R_y) bits generated at each time in (11), are i.i.d. Bernoulli-(0.5).¹ Since parity bits are assigned causally, if two source sequences share the same length- l prefix, then their first lR_x parity bits must match. Subsequent

¹We assume that R_x and R_y are integer. To justify this assumption note that we can always group sets of α successive symbols into super-symbols. These larger symbols can be encoded at an average rate αR_x . Generally, if we group α symbols together, and transmit β bits per super-symbol, we can realize an average rate α/β , i.e., a rational rate. If desired, non-integer average rates are easily implemented by a mildly time-varying transmission rate.

parities are drawn independently. Such a sequential coding strategy is the source-coding parallel to tree and convolutional codes used for channel coding [9]. In fact, we call these “parity bits” as they can be generated using an infinite constraint-length time-varying random convolutional code.

We will often restrict our attention to the set of source sequences that are compatible with the received parities up to time n . Given that $x^n = x^n$ this set is denoted as

$$\mathcal{B}_x(x^n) = \{\tilde{x}^n \in \mathcal{X}^n : \mathcal{E}_j^x(\tilde{x}^j) = \mathcal{E}_j^x(x^j), j = 1, 2, \dots, n\}. \quad (12)$$

An analogous definition holds for $\mathcal{B}_y(y^n)$.

We define the pair of source estimates at time n as $(\hat{x}^n, \hat{y}^n) = \mathcal{D}_n(\prod_{j=1}^n \mathcal{E}_j^x, \prod_{j=1}^n \mathcal{E}_j^y)$, where $\prod_{j=1}^n \mathcal{E}_j^x$ indicates the full nR_x bit stream from encoder x up to time n . We use $(\hat{x}^{n-\Delta}, \hat{y}^{n-\Delta})$ to indicate the first $n - \Delta$ symbols of each estimate, where for conciseness of notation both the estimate time, n , and the decoding delay, Δ , are indicated in the superscript. With these definitions the two marginal error probabilities are

$$\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] \quad \text{and} \quad \Pr[\hat{y}^{n-\Delta} \neq y^{n-\Delta}].$$

A pair of exponents $E_x > 0$ and $E_y > 0$ is said to be achievable if there exists a family of rate- (R_x, R_y) encoders and decoders $\{(\mathcal{E}_j^x, \mathcal{E}_j^y, \mathcal{D}_j)\}$ such that

$$\lim_{\Delta \rightarrow \infty} \inf_{n > \Delta} -\frac{1}{\Delta} \log \Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] \geq E_x \quad (13)$$

$$\lim_{\Delta \rightarrow \infty} \inf_{n > \Delta} -\frac{1}{\Delta} \log \Pr[\hat{y}^{n-\Delta} \neq y^{n-\Delta}] \geq E_y \quad (14)$$

In contrast to the block-coding error event of (1) this error exponent is in delay, Δ , rather than total observation time, n . While the definitions of the exponents (13)–(14) and of (1) are asymptotic in nature, the error bounds stated in the theorems hold for finite n and Δ . Finally, we note that, as in the block coding case, the error exponent of the joint error event can be found by taking the minimum of the individual exponents, i.e.,

$$\lim_{\Delta \rightarrow \infty} \inf_{n > \Delta} -\frac{1}{\Delta} \log \Pr[(\hat{x}^{n-\Delta}, \hat{y}^{n-\Delta}) \neq (x^{n-\Delta}, y^{n-\Delta})] \geq \min\{E_x, E_y\}.$$

3.2 Point-to-point streaming

Our first results concern streaming coding in the point-to-point setting. The first theorem provides achievable bounds on the random coding error exponents both for ML and universal decoding.

Theorem 2 *Given any rate R , there exist both ML and universal randomized sequential encoder-decoder pairs (per Definition 2) such that for all $E < E_{pt,x}(R)$ there is a constant $K > 0$ such that $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] \leq K \exp\{-\Delta E\}$ for all $n, \Delta \geq 0$ where*

$$E_{pt,x}(R) = \sup_{0 \leq \rho \leq 1} \rho R_x - \log \left[\sum_x p_x(x)^{\frac{1}{1+\rho}} \right]^{1+\rho} \quad (15)$$

$$= \inf_{\bar{x}} D(\bar{x} \| p_x) + |R - H(\bar{x})|^+. \quad (16)$$

For example, say we want to implement an average encoding rate of 5/4 bits per source symbol. Say we generate one new parity bit per symbol for each symbol observed except for the fourth symbol, eighth symbol, etc, when we generate two. The average encoding rate is 5/4. As long as the decoding delay Δ we target is long enough so that the decoder received an “average” number, ΔR_x , of encoded bits before we make an estimate (e.g., if $\Delta \gg 1/R_x$), these small-scale issues even out. In particular, they do not effect the exponents.

where in (16) \bar{x} is a random variable on \mathcal{X} with distribution $p_{\bar{x}}$ and entropy $H(\bar{x})$.

Proof: In Section 5.3 a Gallager-style analysis of ML decoding yields the form of the exponent specified in (15). This analysis is the source-coding parallel to the traditional one for convolutional channel codes. In Section 5.4 a types-based analysis of a novel universal decoder yields the form of the exponent specified in (16). Here, the crucial issue that must be side stepped is the non-additivity of empirical entropy. The equality of the two forms of the exponent in (15) and (16) is a classic result. For example, see [6, pg. 44] exercise 13. ■

The error exponent of Theorem 2 equals the random source coding exponent for block-coding (10). The main difference in the formulation is that the error probability in a streaming system decays with delay Δ rather than block length N . For any fixed source symbol with time index j , as time progresses ($n \rightarrow \infty$) the delay $\Delta = n - j$ also increases without bound. Thus all symbols are eventually recovered with probability one.

3.3 Streaming with decoder side information

Our result for distributed streaming source coding when the side information is observed at the decoder, but not the encoder, is encapsulated in the following theorem:

Theorem 3 *Given any rate R , there exist both ML and universal randomized sequential encoder-decoder pairs (per Definition 2) such that for all $E < E_{si}(R)$ there is a constant $K > 0$ such that $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] \leq K \exp\{-\Delta E\}$ for all $n, \Delta \geq 0$ where*

$$E_{si}(R) = \sup_{0 \leq \rho \leq 1} \rho R_x - \log \left[\sum_y \left[\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right] \quad (17)$$

$$= \inf_{\bar{x}, \bar{y}} D(p_{\bar{x}, \bar{y}} \| p_{x, y}) + |R - H(\bar{x}|\bar{y})|^+, \quad (18)$$

and (\bar{x}, \bar{y}) are random variables with joint distribution $p_{\bar{x}, \bar{y}}$ and $H(\bar{x}|\bar{y})$ is their conditional entropy.

Similar to the point-to-point case in Theorem 2, the error exponent of Theorem 3 equals its random block-coding counterpart (9). Similarly, (17) and (18) can be shown to be equal. We do not prove this equivalence herein but, as a first step, the interested reader could consider [6, pg. 192] exercise 23. We sketch the proof of this theorem in Section 5.5, which requires only small modifications of the techniques used to prove Theorem 2.

3.4 Distributed coding of streaming sources

In contrast to streaming point-to-point coding and streaming source coding with decoder side information, the general case of streaming Slepian-Wolf coding with two separate encoders results in error exponents that differ from their block coding counterparts.

Theorem 4 *Given any rate pair (R_x, R_y) , there exist both ML and universal randomized encoder-decoder triplets (per Definition 2) that satisfy the following three criteria:*

(i) *For all $E < E_{st,x}(R_x, R_y)$, there is a constant $K > 0$ such that $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] \leq K \exp\{-\Delta E\}$ for all $n, \Delta \geq 0$ where*

$$E_{st,x}(R_x, R_y) = \min \left\{ \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma), \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma) \right\}. \quad (19)$$

(ii) For all $E < E_{st,y}(R_x, R_y)$ there is a constant $K > 0$ such that $\Pr[\hat{y}^{n-\Delta} \neq y^{n-\Delta}] \leq K \exp\{-\Delta E\}$ for all $n, \Delta \geq 0$ where

$$E_{st,y}(R_x, R_y) = \min \left\{ \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_x(R_x, R_y, \gamma), \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) \right\}. \quad (20)$$

(iii) For all $E < E_{st,xy}(R_x, R_y)$ there is a constant $K > 0$ such that $\Pr[(\hat{x}^{n-\Delta}, \hat{y}^{n-\Delta}) \neq (x^{n-\Delta}, y^{n-\Delta})] \leq K \exp\{-\Delta E\}$ for all $n, \Delta \geq 0$ where

$$E_{st,xy}(R_x, R_y) = \min \left\{ \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma), \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) \right\}. \quad (21)$$

There are two alternate, but equivalent, ways to specify the above error exponents. The first is the ‘‘Gallager-style’’

$$\begin{aligned} E_x(R_x, R_y, \gamma) &= \sup_{\rho \in [0,1]} [\gamma E_{x|y}(R_x, \rho) + (1-\gamma) E_{xy}(R_x, R_y, \rho)] \\ E_y(R_x, R_y, \gamma) &= \sup_{\rho \in [0,1]} [\gamma E_{y|x}(R_y, \rho) + (1-\gamma) E_{xy}(R_x, R_y, \rho)], \end{aligned} \quad (22)$$

where $E_{xy}(\cdot, \cdot, \cdot)$, $E_{x|y}(\cdot, \cdot)$, and $E_{y|x}(\cdot, \cdot)$ are defined as in (6)–(8), repeated here for convenience:

$$\begin{aligned} E_{xy}(R_x, R_y, \rho) &= \rho(R_x + R_y) - \log \left[\sum_{x,y} p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \\ E_{x|y}(R_x, \rho) &= \rho R_x - \log \left[\sum_y \left[\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right] \\ E_{y|x}(R_y, \rho) &= \rho R_y - \log \left[\sum_x \left[\sum_y p_{xy}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right]. \end{aligned}$$

Alternately, the second ‘‘Csiszar-style’’ form of the exponents is

$$\begin{aligned} E_x(R_x, R_y, \gamma) &= \inf_{\tilde{x}, \tilde{y}, \bar{x}, \bar{y}} \gamma D(p_{\tilde{x}, \tilde{y}} \| p_{x, y}) + (1-\gamma) D(p_{\bar{x}, \bar{y}} \| p_{x, y}) \\ &\quad + \left| \gamma [R_x - H(\tilde{x} | \tilde{y})] + (1-\gamma) [R_x + R_y - H(\bar{x}, \bar{y})] \right|^+ \\ E_y(R_x, R_y, \gamma) &= \inf_{\tilde{x}, \tilde{y}, \bar{x}, \bar{y}} \gamma D(p_{\tilde{x}, \tilde{y}} \| p_{x, y}) + (1-\gamma) D(p_{\bar{x}, \bar{y}} \| p_{x, y}) \\ &\quad + \left| \gamma [R_y - H(\tilde{y} | \tilde{x})] + (1-\gamma) [R_x + R_y - H(\bar{x}, \bar{y})] \right|^+, \end{aligned} \quad (23)$$

where the random variables (\tilde{x}, \tilde{y}) and (\bar{x}, \bar{y}) have joint distributions $p_{\tilde{x}, \tilde{y}}$ and $p_{\bar{x}, \bar{y}}$, respectively.

Proof: In Section 6.3 a Gallager-style analysis of ML decoding yields the form of the exponents specified in (22). In Section 6.4 a types-based analysis of a novel universal decoder yields the form of the exponent specified in (23). The equality of the two forms of the exponent is considered in Lemma 5, stated and proved in Appendix C. ■

It is revealing to compare the form of the exponents of block coding (3)–(5) with those of streaming (19)–(21). The streaming exponent contains an extra degree of freedom in the parameter γ . If γ were restricted to be either zero or one, then the block and streaming exponents would be the same. The minimization over γ where $0 \leq \gamma \leq 1$ results from a fundamental difference in the types of events that can cause errors in streaming as opposed to the block setting. In block coding

there are only 3 error events (error in x^n , error in y^n , and errors in both), regardless of block length. In contrast, there are n^2 mutually exclusive error events when decoding x^n and y^n . These arise from the n^2 pairs of time indexes at which the error patterns can commence, i.e., $l, k \in \{1, 2, \dots, n\}$ such that $\hat{x}^{l-1} = x^{l-1}$ and $\hat{y}^{k-1} = y^{k-1}$, but $\hat{x}_l \neq x_l$ and $\hat{y}_k \neq y_k$. We examine these error events in Section 6.

While the error exponents of block coding are always at least as large as the streaming exponents due to the lack of the γ parameter, direct comparison of the two is not really appropriate for two reasons. The first is that buffering delay is not accounted for in block coding. Streaming data must first be packetized into “chunks” of data of the appropriate length to which the block-encoding can be applied. Such packetization delay is not accounted for in the block coding exponents and, at worst, would double the delay on a particular symbols (those at the beginning of each block). The second reason is that in block coding the block length is fixed and therefore so is the resulting error probability. In the streaming context the error probability on the estimate of any fixed source symbol continues to decrease as time increments and the decoding delay Δ (for that particular symbol) increases.

Finally, as in the point-to-point setting, the two forms of the exponents in (22) and (23) are equal. But, due to new classes of error events possible in streaming, this equivalence now requires proof. This proof is provided in Lemma 5.

4 Numerical Results

In this section we detail two examples. The first example is presented in part in Fig. 2 in the Introduction and helps us understand how source “burstiness” relates to the achievable error exponent in delay. For simplicity we present these results for lossless point-to-point streaming, i.e., Theorem 2. The second example illustrates the difference between the block-coding and streaming exponents for a simple distributed asymmetric binary source. In this section we express entropy in bits.

The source considered in the first example is a discrete memoryless source with alphabet $\{0, 1, \dots, L\}$ where $p_x(0) = (1 - \beta)$ and $p_x(x) = \beta/L$ for all $x \neq 0$. The β parameter specifies the “burstiness” of the source and the entropy of this source is $H(X) = H_B(\beta) + \beta \log L$. In Figure 2 we consider $L = 4$ and $\beta = 0.5$, hence $H(X) = 2$ bits. Figure 2 plots the trade-off between rate, delay, and probability of error. As would be expected, the probability of decoding error drops both as a function of communication rate and delay.

The source of Figure 2 is only mildly bursty, half the time it emits a 0 and half the time some other letter. In Figure 3 we plot the error exponent of the same family of sources for a range of burst probabilities β and alphabet sizes $L + 1$ where we hold the entropy constant at $H(X) = 2$ bits. As the source becomes more bursty (smaller β) we increase the alphabet size to maintain the equality $H(X) = 2 = H_B(\beta) + \beta \log L$. The figure shows that that the more bursty the source (smaller β and large L) the smaller the error exponent for any given rate.

Our second example illustrates a distributed source coding situation where the streaming and block coding error exponents differ. The reason for the difference is the new type of error event (reflected in the minimization over γ in Theorem 4) that can dominant in the distributed streaming setting. However, it turns out that when the distributed source has uniform symmetric marginals there is no gap between the streaming and block coding error exponents. Thus, we consider the following asymmetric example (asymmetric marginals and asymmetric channel relating x to y). The

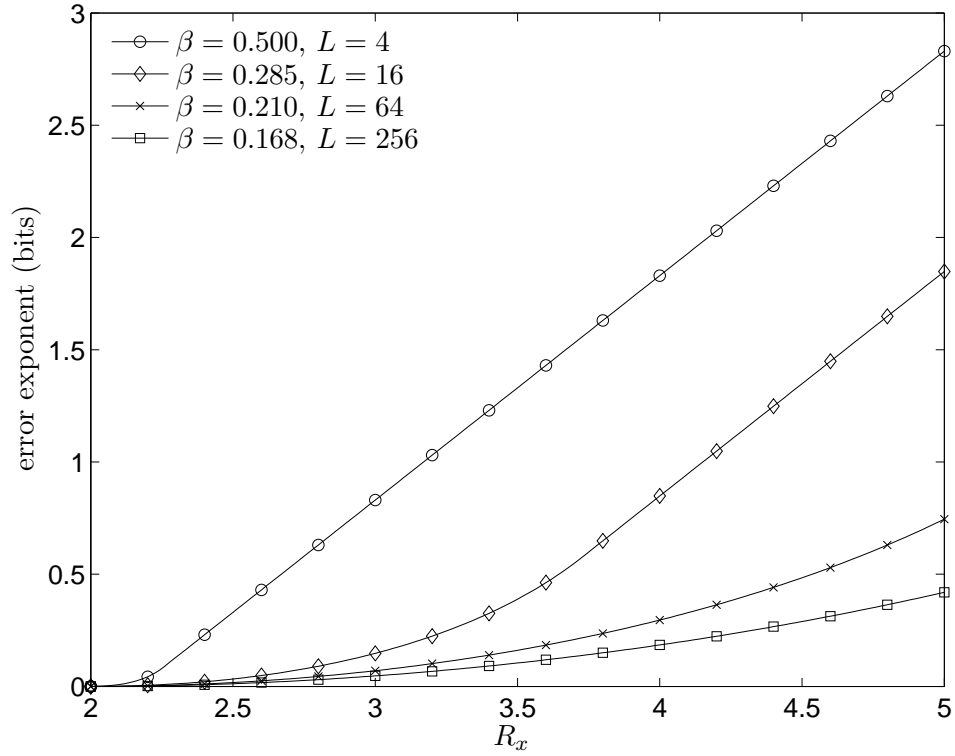


Figure 3: The effect of burstiness on the error exponent as a function of the excess rate beyond the entropy. The source is 0 with probability $1 - \beta$ and, with probability β , the source is uniformly distributed on $\{1, 2, \dots, L\}$. We scale L with the burst probability β to hold the source entropy constant at 2 bits. A lower burst probability β means more variability in the instantaneous rate, the effect of which is a lowered exponent.

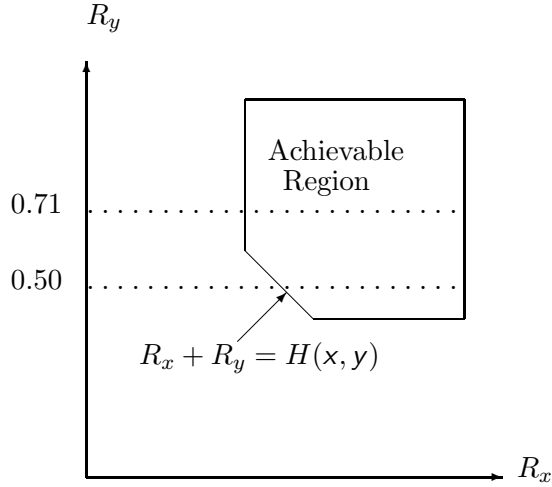


Figure 4: Rate region for the asymmetric example source.

pair of sources x_i and y_i are binary i.i.d. sources where $p_{x,y}(0,0) = 0.1$, $p_{x,y}(0,1) = p_{x,y}(1,0) = 0.05$ and $p_{x,y}(1,1) = 0.8$. For this source $H(x) = H(y) = 0.61$ bits, $H(x|y) = H(y|x) = 0.42$ bits and $H(x,y) = 1.02$ bits. The Slepian-Wolf achievable rate region is shown in Fig. 4. We consider various error exponents for this source as a function of R_x where we keep R_y fixed. We consider both a low R_y -rate situation where $R_y = 0.5$ bits and a high R_y -rate situation where $R_y = 0.71$ bits.

Figure 5 plots the streaming exponent $E_{st,x}(R_x, R_y)$ for source x from Theorem 4, the block coding exponent $E_{bl,x}(R_x, R_y)$ from Theorem 1, and the point-to-point exponent $E_{pt,x}(R_x)$ from Theorem 2. All are plotted as a function of R_x , and the first two for both $R_y = 0.5$ and $R_y = 0.71$. A note on the plots: since $E_{st,x}(R_x, R_y) = E_{bl,x}(R_x, R_y)$ for many choices of R_x and R_y , we choose to plot the block coding exponents with solid or dashed lines and the streaming exponents with circles or diamonds. Both are, of course, continuous functions of R_x . Our choice of plotting the streaming exponents at a discrete set of points was made purely to aid in making visual comparison between the exponents.

There are a few observations to make about Figure 5. Perhaps the most significant is that, in order to recover the x -source with the greatest likelihood, it can be better *not* to use joint decoding if R_y is too low. For example, when $R_y = 0.5$ and $R_x > 0.75$ bits, $E_{pt,x}(R_x)$ is greater than either $E_{st,x}(R_x, 0.5)$ or $E_{bl,x}(R_x, 0.5)$. This occurs because joint decoding errors are more likely due to atypical behavior of source y . Thus, it can be better to ignore the y -source and decode the x -source individually. As R_y is increased, e.g., to $R_y = 0.71$ bits, the information about the y -source is more reliable and the joint decoding exponents dominate that of point-to-point source coding without side information. The next observation to make is that the difference between the block and streaming error exponents is small and often zero. In Figure 5 the difference between the two is only clearly apparent at rates close to $R_x = 0$ for the $R_y = 0.5$ case. To see more clearly where the streaming and block coding exponents differ, in Figure 6 we plot the ratio $E_{bl,x}(R_x, R_y)/E_{st,x}(R_x, R_y)$. In this figure we see that for both $R_y = 0.71$ and $R_y = 0.5$ there are ranges of R_x in which the exponents

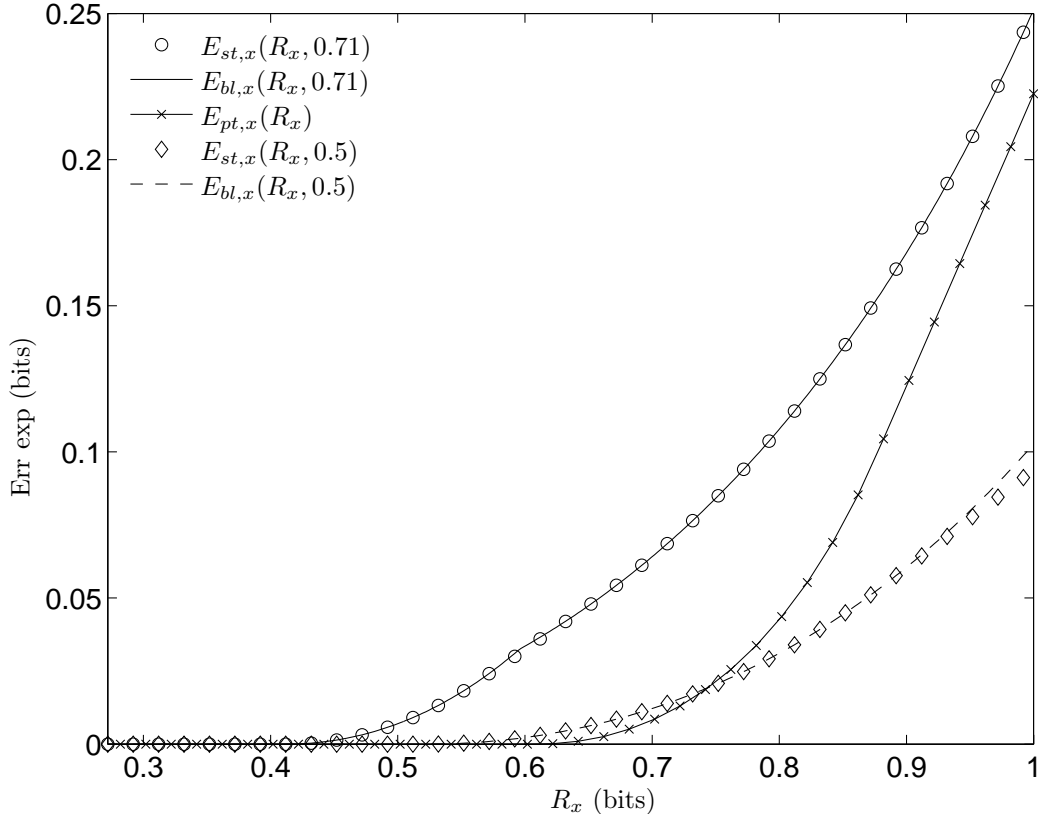


Figure 5: Error exponents for x-source: streaming $E_{st,x}(R_x, R_y)$, block-coding $E_{bl,x}(R_x, R_y)$, and point-to-point source coding $E_{pt,x}(R_x)$ at two rates: $R_y = 0.71$ and $R_y = 0.5$ bits per sample.

differ. Interestingly the ranges do not overlap.

Figures 7 and 8 plot the corresponding story for $E_{st,y}(R_x, R_y)$, $E_{bl,y}(R_x, R_y)$, and $E_{pt,y}(R_y)$ for $R_y = 0.5$ and $R_y = 0.71$. Note that $E_{pt,y}(0.5) = 0$ since $R_y = 0.5 < H(y) = 0.61$ bits and $E_{pt,y}(0.71)$ is constant at about 0.01. This means that joint decoding is required to get any positive exponent on the y -source for $R_y = 0.5$ bits joint decoding is required. On the other hand, at $R_y = 0.71$ bits the joint decoding exponent is better than individual encoding only at sufficiently high R_x rates. Next note that when R_x is sufficiently high, the the error exponent for y saturates to what it would be if source x were known perfectly to the decoder. Recalling the discussion in Sections 2 and 3, this is the contribution of the $E_{y|x}(R_y, \rho)$ term to the exponents in (4) and (20). Finally, note that the difference between the block and streaming exponents is most apparent for $R_y = 0.71$ bits right around $R_x = 0.4$ bits and around $R_x = 0.75$ bits. As before, to better visualize the difference between the two exponents, in Figure 8 we plot the ratio $E_{bl,y}(R_x, R_y)/E_{st,y}(R_x, R_y)$. In this plot we note two features that didn't appear in Figure 6. The first is that there are two disjoint ranges of R_x in which $E_{bl,y}(R_x, 0.71) > E_{st,y}(R_x, 0.71)$. The second is that a situation is illustrated where the ratio between the two exponents can be unbounded. Namely, in the range $0.3 < R_x < 0.45$ the ratio is unbounded as $E_{bl,y}(R_x, 0.71) > 0$ while $E_{st,y}(R_x, 0.71) = 0$.

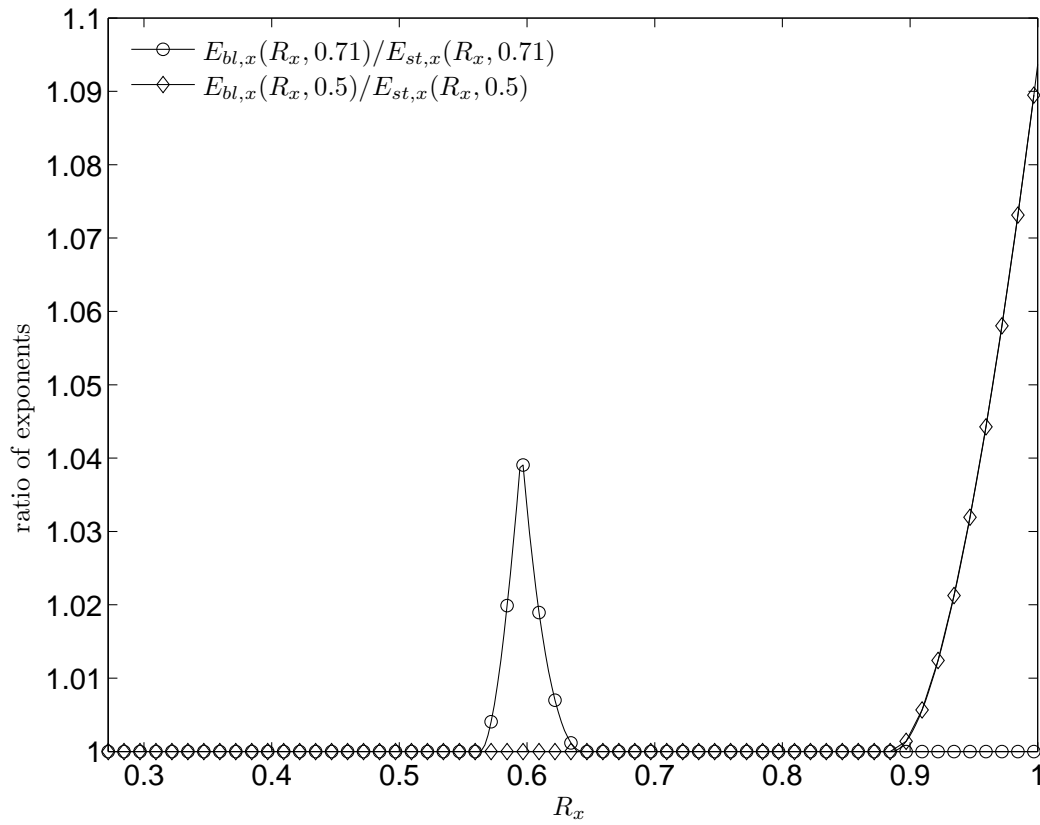


Figure 6: Ratio of block-coding to streaming exponents for source-x. The block coding exponent is always the larger of the two.

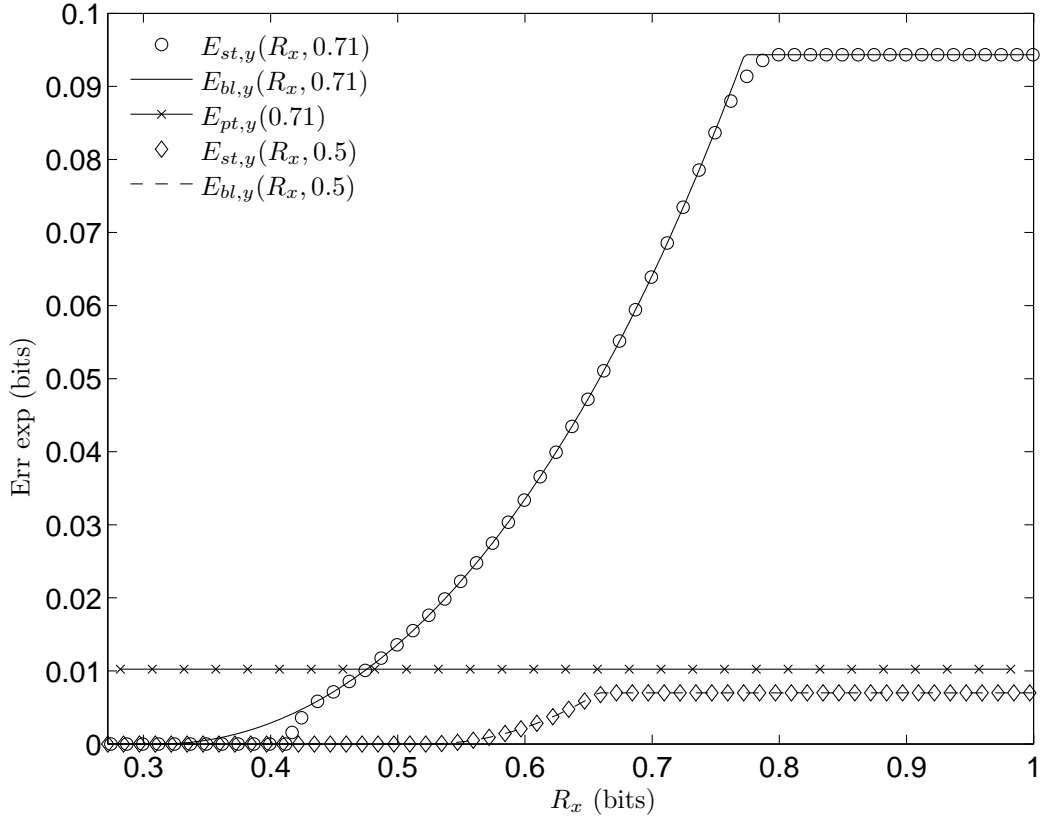


Figure 7: Error exponents for y -source: streaming $E_{st,y}(R_x, R_y)$, block-coding $E_{bl,y}(R_x, R_y)$, and point-to-point source coding $E_{pt,y}(R_y)$ at two rates: $R_y = 0.71$ and $R_y = 0.5$ bits per sample. The point-to-point exponent $E_{pt,y}(R_y)$ is non-zero only for $R_y = 0.71$ since $R_y = 0.5 < H(y) = 0.61$.

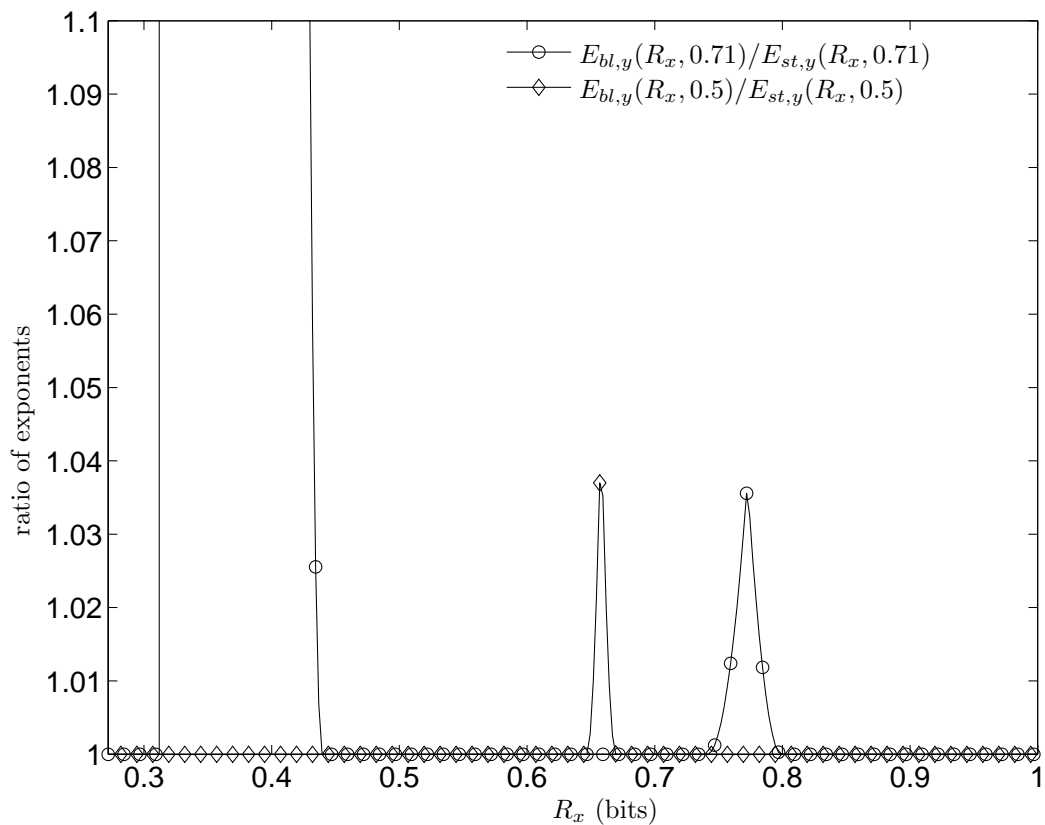


Figure 8: Ratio of block-coding to streaming exponents for source- y . Note that for the high-rate case ($R_y = 0.71$) and in the rough range $0.3 < R_x < 0.45$, the ratio of the exponents is unbounded as the block-coding exponent $E_{bl,y}(R_x, 0.71) > 0$ while the streaming exponent $E_{st,x}(R_x, 0.71) = 0$.

5 Streaming point-to-point coding via sequential random binning

In this section we prove Theorems 2 and 3. While the emphasis of the paper is on distributed source coding, the strategy of causal random binning, the appropriate ML and universal decoders, and the associated analysis techniques, are most easily developed in the point-to-point context.

5.1 Sequential scoring decoders

In this section we introduce the class of decoders used for streaming source coding and streaming source coding with decoder side information. Both ML and universal decoders can be cast as a type of decision-directed sequential scoring decoder, where different scoring functions are used in each case.

Definition 3 *A sequential scoring decoder constructs its estimate in a sequential manner starting from $l = 1$ where*

$$\hat{x}_l = \begin{cases} \bar{x}_l & \text{if } \begin{array}{l} \text{for some } \bar{\mathbf{x}}^n \in \mathcal{B}_x(\mathbf{x}) \text{ s.t. } \bar{x}^{l-1} = \hat{x}^{l-1} \\ S_l(\bar{\mathbf{x}}) \geq S_l(\tilde{\mathbf{x}}) \text{ for all } \tilde{\mathbf{x}} \in \mathcal{B}_x(\mathbf{x}) \text{ s.t. } \tilde{x}^{l-1} = \hat{x}^{l-1}, \tilde{x}_l \neq \bar{x}_l \end{array} \\ ? & \text{otherwise} \end{cases} \quad (24)$$

where $S_l(\cdot)$ is a (possibly time-dependent) scoring function and the (failure) symbol “?” is included in case such a $\bar{\mathbf{x}}$ does not exist for some l . Randomly resolve any ties that occur.

Since the sequential scoring decoder is a decision-directed decoder, it considers as candidates only those sequences whose parities match the received bit stream up to time n , i.e., if the length- n source sequence is $\mathbf{x} = \mathbf{x}$ then the set of such candidates is $\{\bar{\mathbf{x}} \text{ s.t. } \bar{\mathbf{x}} \in \mathcal{B}_x(\mathbf{x})\}$. The l th symbol of the estimate, \hat{x}_l , is made with the estimates of the first $l - 1$ symbols already fixed. One should note that as soon as the next set of parities arrive at the receiver, all symbols are estimated anew since n is now replaced by $n + 1$ and $\mathcal{B}_x(\mathbf{x}^{n+1})$ will be different from $\mathcal{B}_x(\mathbf{x}^n)$.

For ML decoding case we use the scoring function

$$S_l(\bar{x}) = p_{x_l^n}(\bar{x}_l^n). \quad (25)$$

Note that this scoring function simply leads to the ML estimate $\hat{\mathbf{x}}_{ML} = \arg \max_{\bar{\mathbf{x}} \in \mathcal{B}_x(\mathbf{x})} p_{\mathbf{x}}(\bar{\mathbf{x}})$ being constructed in a sequential manner. This is the case since the decision regarding which of a pair of sequences is more likely depends only on which sequence has the more likely suffix. Another way of saying this is that, if we were to consider the log-probability score $S_l = \log p_{x_l^n}(\bar{x}_l^n)$, then the score would be additive for i.i.d. sequences. Thus, we could equally have chosen $S_l(\bar{\mathbf{x}}) = p_{\mathbf{x}}(\bar{\mathbf{x}})$. On the other hand, since empirical entropy is not additive (think of a sequence of all 0s followed by a sequence of all 1s) the use of sequential scoring decoders will be more crucial in universal decoding.

For universal decoding we use the reciprocal of the empirical suffix-entropy as the score

$$S_l(\bar{x}) = 1/H(\bar{x}_l^n). \quad (26)$$

and term the resulting decoding the “minimum empirical suffix entropy decoder”. The reason for using this decoder instead of the standard minimum empirical block-entropy decoder is because (due to the summing over type classes) the probability of error bound for the block-entropy decoder has a pre-multiplier term that grows polynomially in n . Since our bound on error probability will decay exponential in Δ , for n large, the polynomial can dominate. This would prevent us from deriving a bound on the probability of error that depends only upon the decoding delay Δ . Using the minimum empirical suffix-entropy decoder results in a term that grows polynomially only in Δ .

5.2 Error analysis of sequential scoring decoders

To show Theorem 2 we first develop the common core of the proof that applies to both ML and universal decoding. The proof strategy is as follows. A decoding error can only occur if there is some spurious source sequence \tilde{x}^n that satisfies three conditions: (i) $\tilde{x}^n \in \mathcal{B}_x(x^n)$, i.e., it must be in the same bin (share the same parities) as x^n , (ii) $\tilde{x}_l \neq x_l$ for some $l \leq n - \Delta$, and (iii) for the time index l of event (ii) it must have a score at least as large as the correct sequence, i.e., $S_l(\tilde{x}^n) \geq S_l(x^n)$.

To help track condition (ii) and to keep notation compact we introduce a partition of all length- n source sequences $\tilde{x}^n \in \mathcal{X}^n$ into non-overlapping sets $\mathcal{F}_n(l, x^n)$ defined by the time index l of the first sample in which each sequence differs from the realized sequence x^n . Formally,

$$\mathcal{F}_n(l, x^n) = \{\tilde{x}^n \in \mathcal{X}^n | \tilde{x}^{l-1} = x^{l-1}, \tilde{x}_l \neq x_l\}, \quad (27)$$

where we define $\mathcal{F}_n(n+1, x^n) = \{x^n\}$, thus $\cup_{l=1}^{n+1} \mathcal{F}_n(l, x^n) = \mathcal{X}^n$.

With these definitions we rewrite the error probability as

$$\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] = \sum_{x^n} \Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta} | x^n = x^n] p_{\mathbf{x}}(x^n) \quad (28)$$

$$= \sum_{x^n} \sum_{l=1}^{n-\Delta} \Pr[\exists \tilde{x}^n \in \mathcal{B}_x(x^n) \cap \mathcal{F}_n(l, x^n) \text{ s.t. } S_l(\tilde{x}^n) \geq S_l(x^n)] p_{\mathbf{x}}(x^n) \quad (29)$$

$$= \sum_{l=1}^{n-\Delta} \left\{ \sum_{x^n} \Pr[\exists \tilde{x}^n \in \mathcal{B}_x(x^n) \cap \mathcal{F}_n(l, x^n) \text{ s.t. } S_l(\tilde{x}^n) \geq S_l(x^n)] p_{\mathbf{x}}(x^n) \right\} \quad (30)$$

After conditioning on the realized source sequence in (28), the remaining randomness is only in the binning. In (29) the error event is decomposed into mutually exclusive events based on the discussion of conditions (i)-(iii) above, and the partitioning of all length- n sources sequences into the sets $\mathcal{F}_n(l, x^n)$. Finally, defining

$$p_n(l) = \sum_{x^n} \Pr[\exists \tilde{x}^n \in \mathcal{B}_x(x^n) \cap \mathcal{F}_n(l, x^n) \text{ s.t. } S_l(\tilde{x}^n) \geq S_l(x^n)] p_{\mathbf{x}}(x^n). \quad (31)$$

and substituting the results into (30) yields the relation

$$\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] = \sum_{l=1}^{n-\Delta} p_n(l). \quad (32)$$

5.3 Maximum-likelihood decoding

the following lemma provides an upper bound on $p_n(l)$ for ML decoding with the score function $S_l = p_{x_l^n}(\tilde{x}_l^n)$ specified in (25). The proof is given in Appendix A and uses a Chernoff bounding argument similar to [10].

Lemma 1

$$p_n(l) \leq \exp\{-(n-l+1)E_{pt,x}(R)\},$$

where the form of $E_{pt,x}(R)$ is given in (15).

Using Lemma 1 in (32) gives

$$\begin{aligned} \Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] &\leq \sum_{l=1}^{n-\Delta} \exp\{-(n-l+1)E_{pt,x}(R)\} & (33) \\ &= \sum_{l=1}^{n-\Delta} \exp\{-(n-l+1-\Delta)E_{pt,x}(R)\} \exp\{-\Delta E_{pt,x}(R)\} \\ &\leq K_0 \exp\{-\Delta E_{pt,x}(R)\} & (34) \end{aligned}$$

In (34) we pull out the exponent in Δ . The remaining summation is a geometric sum over decaying exponentials and can thus be bounded by some constant K_0 . This proves Theorem 2 for ML decoding.

The derivation illustrates the insight that sequential decision made for each symbol is analogous to a classic block-coding problem. This is because we only need to decide between sequences that start to differ in the symbol we are trying to estimate — previous symbols have been fixed, and subsequent symbols are not yet in question. Thus, all sequences that could lead to different estimates of symbol l are binned independently for the remainder of the block. This is why the error exponent we derive equals Gallager’s block coding exponent [10]. Since the error exponent for each block-decoding problem is the same, the dominant error event is the hard-decision with the shortest block-length. This corresponds to the last symbol we need to estimate and its block-length equals the estimation delay Δ .

5.4 Universal decoding

The following lemma provides an upper bound on $p_n(l)$ for universal decoding with the score function $S_l(\bar{x}) = 1/H(\bar{x}_l^n)$ defined in (26).

$$p_n(l) = \sum_{x^n} \Pr [\exists \tilde{x}^n \in \mathcal{B}_x(x^n) \cap \mathcal{F}_n(l, x^n) \text{ s.t. } H(\tilde{x}_l^n) \leq H(x_l^n)] p_{\mathbf{x}}(x^n), \quad (35)$$

and the following lemma bounds $p_n(l)$.

Lemma 2 *For minimum empirical suffix-entropy decoding, $p_n(l) \leq (n-l+2)^{2|\mathcal{X}|} \exp\{-(n-l+1)E_{pt,x}(R)\}$.*

Proof: We define P^{n-l} to be the type of length- $(n-l+1)$ sequence x_l^n , and $\mathcal{T}_{P^{n-l}}$ to be the corresponding type class so that $x_l^n \in \mathcal{T}_{P^{n-l}}$. Analogous definitions hold for \tilde{P}^{n-l} and \tilde{x}_l^n . We

rewrite the constraint $H(\tilde{x}_l^n) < H(\tilde{x}_l^n)$ as $H(\tilde{P}^{n-l}) < H(P^{n-l})$. Thus,

$$\begin{aligned}
p_n(l) &= \sum_{x^n} \Pr [\exists \tilde{x}^n \in \mathcal{B}_x(x^n) \cap \mathcal{F}_n(l, x^n) \text{ s.t. } H(\tilde{x}_l^n) \leq H(x_l^n)] p_{\mathbf{x}}(x^n) \\
&\leq \sum_{x_1^n} \min \left[1, \sum_{\substack{\tilde{x}_1^n \in \mathcal{F}_n(l, x_1^n) \\ H(\tilde{x}_1^n) \leq H(x_1^n)}} \Pr[\tilde{x}_1^n \in \mathcal{B}_x(x_1^n)] \right] p_{\mathbf{x}}(x^n) \\
&= \sum_{x_1^{l-1}, x_l^n} \min \left[1, \sum_{\substack{\tilde{x}_l^n \text{ s.t.} \\ H(\tilde{x}_l^n) \leq H(x_l^n)}} \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x^{l-1}) p_{\mathbf{x}}(x_l^n) \\
&= \sum_{x_l^n} \min \left[1, \sum_{\substack{\tilde{x}_l^n \text{ s.t.} \\ H(\tilde{x}_l^n) \leq H(x_l^n)}} \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x_l^n) \tag{36}
\end{aligned}$$

$$= \sum_{P^{n-l}} \sum_{x_l^n \in \mathcal{T}_{P^{n-l}}} \min \left[1, \sum_{\substack{\tilde{P}^{n-l} \text{ s.t.} \\ H(\tilde{P}^{n-l}) \leq H(P^{n-l})}} \sum_{\tilde{x}_l^n \in \mathcal{T}_{\tilde{P}^{n-l}}} \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x_l^n) \tag{37}$$

$$\leq \sum_{P^{n-l}} \sum_{x_{l+1}^n \in \mathcal{T}_{P^{n-l}}} \min \left[1, (n-l+2)^{|\mathcal{X}|} \exp\{-(n-l)[R - H(P^{n-l})]\} \right] p_{\mathbf{x}}(x_l^n) \tag{38}$$

$$\begin{aligned}
&\leq (n-l+2)^{|\mathcal{X}|} \sum_{P^{n-l}} \sum_{x_l^n \in \mathcal{T}_{P^{n-l}}} \exp\{-(n-l+1)[|R - H(P^{n-l})|^+]\} \\
&\quad \exp\{-(n-l+1)[D(P^{n-l} \| p_{\mathbf{x}}) + H(P^{n-l})]\} \tag{39}
\end{aligned}$$

$$\leq (n-l+2)^{|\mathcal{X}|} \sum_{P^{n-l}} \exp\{-(n-l+1) \inf_q [D(q \| p_{\mathbf{x}}) + |R - H(q)|^+]\} \tag{40}$$

$$\leq (n-l+2)^{2|\mathcal{X}|} \exp\{-(n-l+1)E_{pt,x}(R)\} \tag{41}$$

that the argument of the inner-most summation (over \tilde{x}_l^n) does not depend on \mathbf{x} . We then use the following relations: (i) $\sum_{\tilde{x}_l^n \in \mathcal{T}_{\tilde{P}^{n-l}}} = |\mathcal{T}_{\tilde{P}^{n-l}}| \leq \exp\{(n-l+1)H(\tilde{P}^{n-l})\}$, which is a standard bound on the size of the type class, (ii) $H(\tilde{P}^{n-l}) \leq H(P^{n-l})$ by the minimum-suffix-entropy decoding rule, and (iii) the polynomial bound on the number of types, $|\{\tilde{P}^{n-l}\}| \leq (n-l+2)^{|\mathcal{X}|}$. In (39) we recall the function definition $|\cdot|^+ \triangleq \max\{0, \cdot\}$. We pull the polynomial term out of the minimization and use $p_{\mathbf{x}}(x_l^n) = \exp\{-(n-l+1)[D(P^{n-l} \| p_{\mathbf{x}}) + H(P^{n-l})]\}$ for all $p_{\mathbf{x}}(x_l^n) \in \mathcal{T}_{P^{n-l}}$. It is also in (39) that we see why we use a minimum empirical suffix-entropy decoding rule instead of a minimum empirical block-entropy decoding rule. If we had not marginalized out over x^{l-1} in (36) then we would have a polynomial term out front in terms of n rather than $n-l$, which for large n could dominate the exponential decay in $n-l$. As the expression in (40) no longer depends on x_l^n , we simplify by using $|\mathcal{T}_{P^{n-l}}| \leq \exp\{(n-l+1)H(P^{n-l})\}$. In (41) we use the form of $E_{pt,x}(\cdot)$ specified in (16) together with the polynomial bound on the number of types. ■

Starting from (32) together the definition of $p_n(l)$ for minimum-suffix decoding from (35) and Lemma 2 provides a bound on the probability of error for universal decoding. Using the definition

of $p_n(l)$ in (35) we have

$$\begin{aligned} \Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] &\leq \sum_{l=1}^{n-\Delta} p_n(l) \\ &\leq \sum_{l=1}^{n-\Delta} (n-l+2)^{2|\mathcal{X}|} \exp\{-(n-l+1)E_{pt,x}(R)\} \\ &\leq \sum_{l=1}^{n-\Delta} K_1 \exp\{-(n-l+1)[E_{pt,x}(R) - \eta]\} \end{aligned} \quad (42)$$

$$\leq K_2 \exp\{-\Delta[E_{pt,x}(R) - \eta]\} \quad (43)$$

In (42) we incorporate the polynomial into the exponent, resulting in the constants K_1 and η . Namely, for all $a > 0$, $b > 0$, there exists a C such that $z^a \leq C \exp\{b(z-1)\}$ for all $z \geq 1$. We then make explicit the delay-dependent term. Pulling out the exponent in Δ , the remaining summation is a sum over decaying exponentials, and can be bounded by a constant. Together with K_1 , this gives the constant K_2 in (43). This proves that universal coding achieves the exponent specified in Theorem 2. Note that the η in (43) does not enter the optimization because $\eta > 0$ can be picked equal to any arbitrarily small constant. The choice of η only effects the constant K in the theorem.

5.5 Comment on streaming source coding with side information at the decoder

If a random sequence y^n , related to the source x^n through a discrete memoryless channel, is observed at the decoder, then this side information can be used to reduce the rate of the source code. In the model we study $p_{\mathbf{x},\mathbf{y}}(x^n, y^n) = \prod_{i=1}^n p_{x,y}(x_i, y_i) = \prod_{i=1}^n p_{x|y}(x_i|y_i)p_y(y_i)$. The source x^n is observed at the encoder, and the decoder, which observes y^n and a bit stream from the encoder, wants to estimate each source symbol x_i with a probability of error that decreases exponentially in the decoding delay Δ .

The earlier analysis of this section applies to this problem with a few very minor modifications. For ML decoding, we need to pick the sequence with the maximum conditional probability given y^n . The error exponent can be derived using a similar Chernoff bounding argument as in Section 5. For universal decoding, the only change is that we now use a minimum suffix conditional-entropy decoder that compares sequence pairs (\bar{x}^n, y^n) and (\tilde{x}^n, y^n) . In terms of the analysis, one change enters in (28) where we must also sum over the possible side information sequences. And in (37) the entropy condition in the summation over $\tilde{\mathbf{x}}$ changes to $H(\tilde{x}_{l+1}^n|y_{l+1}^n) < H(x_{l+1}^n|y_{l+1}^n)$ (or the equivalent type notation). Since y^n is observed at the decoder, there is no ambiguity in the side information. Therefore, this condition is equivalent to $H(\tilde{x}_{l+1}^n, y_{l+1}^n) < H(x_{l+1}^n, y_{l+1}^n)$.

We do not include the full derivation as no new ideas are required.

6 Streaming Slepian-Wolf source coding

In this section we prove ML decoding yields the form of the error exponent specified in (22) and that universal decoding yields the form of the exponent specified in (23). The equivalence of the two forms is deferred to Appendix C. As with the proof of Theorem 2 we first develop the common core of the proof that pertains to both ML and universal decoding. The development for more than

$l > 2$ sources would essentially be the same, just with more notation and additional minimization parameters $\gamma_1, \gamma_2, \dots, \gamma_{l-1}$.

6.1 Sequential joint scoring decoders

We now introduce the class of decoders needed for joint decoding. As in Section 5 both ML and universal decoders can be cast as a type of decision-directed sequential scoring decoder, where different scoring functions are used in each case. The definition is a bit more involved for joint decoders as all possible pairings between $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ sequences, respectively in $\mathcal{B}_x(\mathbf{x})$ and $\mathcal{B}_y(\mathbf{y})$, must be considered.

Definition 4 *A sequential joint scoring decoder constructs its estimate in a sequential manner starting from $l = 1$ where*

$$\hat{x}_l = \begin{cases} \bar{x}_l & \text{if } \begin{array}{l} \text{for some } \bar{\mathbf{x}} \in \mathcal{B}_x(\mathbf{x}) \text{ s.t. } \bar{x}^{l-1} = \hat{x}^{l-1} \\ \text{there exists a } \bar{\mathbf{y}} \in \mathcal{B}_y(\mathbf{y}) \text{ s.t.} \\ \text{for all } (\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathcal{B}_x(\mathbf{x}) \times \mathcal{B}_y(\mathbf{y}) \text{ where } \tilde{x}^{l-1} = \hat{x}^{l-1}, \tilde{x}_l \neq \bar{x}_l \\ S_{l,k}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \geq S_{l,k}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \text{ for the } k \in \{1, 2, \dots, n\} \text{ s.t. } \tilde{y}^{k-1} = \bar{y}^{k-1}, \tilde{y}_k \neq \bar{y}_k \end{array} \\ ? & \text{otherwise} \end{cases} \quad (44)$$

where $S_{l,k}(\cdot, \cdot)$ is a (possibly time-dependent) joint scoring function and the (failure) symbol “?” is included in case such an $\bar{\mathbf{x}}$ does not exist for some l . Ties are resolved randomly.

Just as with the (non-joint) sequential scoring decoders of Definition 3 the estimate of \mathbf{x} is built up sequentially, but now all possible pairings with sequences in $\mathcal{B}_y(\mathbf{y})$ are considered. The “error” symbol “?” is allowed in case there is no $\bar{\mathbf{x}} \in \mathcal{B}_x(\mathbf{x})$ that satisfies the definition. In such an event all subsequent symbol estimates, i.e., \hat{x}_{l+1}^n are also equal to “?”, at least until the next parity symbols become available to the decoder.

In the case of known statistics, we use the scoring function

$$S_{l,k}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = p_{\mathbf{x}, \mathbf{y}}(\bar{\mathbf{x}}_{\min\{l,k\}}^n, \bar{\mathbf{y}}_{\min\{l,k\}}^n). \quad (45)$$

where, since we only consider i.i.d. sources, the dimension of the subscripts in $p_{\mathbf{x}, \mathbf{y}}(\cdot, \cdot)$ can be inferred from the arguments. Just as was the case for (25), this scoring function leads to the ML estimate being constructed in a sequential manner.

In the case of unknown statistics, we use the scoring function

$$S_{l,k}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = 1/H_S(l, k, \bar{\mathbf{x}}, \bar{\mathbf{y}}) \quad (46)$$

where $H_S(\cdot, \cdot, \cdot, \cdot)$ is the “weighted empirical suffix-entropy” function, defined as

$$H_S(l, k, \bar{\mathbf{x}}, \bar{\mathbf{y}}) = \begin{cases} H(\bar{x}_l^n, \bar{y}_l^n) & \text{if } l = k \\ \frac{k-l}{n+1-l} H(\bar{x}_l^{k-1} | \bar{y}_l^{k-1}) + \frac{n+1-k}{n+1-l} H(\bar{x}_k^n, \bar{y}_k^n) & \text{if } l < k \\ \frac{l-k}{n+1-k} H(\bar{y}_k^{l-1} | \bar{x}_k^{l-1}) + \frac{n+1-l}{n+1-k} H(\bar{x}_l^n, \bar{y}_l^n) & \text{if } l > k. \end{cases} \quad (47)$$

Due to the fact that $H_S(l, k, \bar{\mathbf{x}}, \bar{\mathbf{y}})$ weights the empirical suffix entropies differently, based upon the values of l and k , we term the resulting decoder the “minimum weighted empirical suffix entropy” decoder.

Note that the form of the two scoring functions (45) and (47) is more similar than may initially appear. For instance compare the ML scores of two pairs $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ where $\bar{x}^{l-1} = \tilde{x}^{l-1}$ and $\bar{y}^{k-1} = \tilde{y}^{k-1}$, but $\bar{x}_l \neq \tilde{x}_l$ and $\bar{y}_k \neq \tilde{y}_k$, and $l < k$. Then the question of whether $S_{l,k}(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is larger than $S_{l,k}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ is the same as asking whether $-\log p_{\mathbf{x}|\mathbf{y}}(\bar{x}_l^{k-1}|b^{k-l}) - \log p_{\mathbf{x},\mathbf{y}}(\bar{x}_k^n, \bar{y}_k^n)$ is smaller than $-\log p_{\mathbf{x}|\mathbf{y}}(\tilde{x}_l^{k-1}|b^{k-l}) - \log p_{\mathbf{x},\mathbf{y}}(\tilde{x}_k^n, \tilde{y}_k^n)$ where $b^{k-1} = \bar{y}_l^{k-1} = \tilde{y}_l^{k-1}$. The analog to the weightings of (47) comes from the dimensions of the various subsequences.

An error can only occur if there is some erroneous source pair $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathcal{B}_x(\mathbf{x}) \times \mathcal{B}_y(\mathbf{y})$ such that $S_{l,k}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \leq S_{l,k}(\mathbf{x}, \mathbf{y})$ for some $l \leq n - \Delta$. Otherwise, the realized source \mathbf{x} will match \hat{x}^n at least through the $n - \Delta$ th symbol. For both our choices of score functions we show in the following sections that the probability of such an event decays exponentially in Δ .

6.2 Error analysis of sequential joint scoring decoders

We follow the same approach to prove Theorem 4 that we used for lossless streaming source coding in Section 5. We first develop the common core of the proof for sequential joint scoring decoders in general. We then specialize the scoring function to the ML and universal scoring functions, (45) and (47), respectively.

In Theorem 4 three error events are considered: (a) $\Pr[x^{n-\Delta} \neq \hat{x}^{n-\Delta}]$, (b) $\Pr[y^{n-\Delta} \neq \hat{y}^{n-\Delta}]$, and (c) $\Pr[(x^{n-\Delta}, y^{n-\Delta}) \neq (\hat{x}^{n-\Delta}, \hat{y}^{n-\Delta})]$. We develop the error exponent for event (a). The exponent of event (b) follows from a similar derivation, and that of event (c) from an application of the union bound resulting in an exponent that is the minimum of the exponents of events (a) and (b).

For there to be a decoding error there must be some spurious source pair $(\tilde{x}^n, \tilde{y}^n)$ that satisfies three conditions: (i) $\tilde{x}^n \in \mathcal{B}_x(x^n)$ and $\tilde{y}^n \in \mathcal{B}_y(y^n)$, (ii) $\tilde{x}_l \neq x_l$ for some $l \leq n - \Delta$ while $\bar{x}^{l-1} = x^{l-1}$ and (iii) for the time index l of event (ii) and for the $k \in \{1, \dots, n\}$ such that $\tilde{y}^{k-1} = y^{k-1}$ but $\tilde{y}_k \neq y_k$, the spurious source pair has a higher score than the true pair, i.e., $S_{l,k}(\tilde{x}^n, \tilde{y}^n) > S_{l,k}(x^n, y^n)$.

As in (27) we again introduce a partition of source sequences to track condition (ii). This time we partition all source pairs $(\tilde{x}^n, \tilde{y}^n) \in \{\mathcal{X}^n, \mathcal{Y}^n\}$ into sets $\mathcal{F}_n(l, k, x^n, y^n)$ defined by the times l and k at which \tilde{x}^n and \tilde{y}^n respectively diverge from the realized source sequences. Formally,

$$\mathcal{F}_n(l, k, x^n, y^n) = \{(\bar{x}^n, \bar{y}^n) \in \mathcal{X}^n \times \mathcal{Y}^n \text{ s.t. } \bar{x}^{l-1} = x^{l-1}, \bar{x}_l \neq x_l, \bar{y}^{k-1} = y^{k-1}, \bar{y}_k \neq y_k\}, \quad (48)$$

and $\mathcal{F}_n(n+1, n+1, x^n, y^n) = \{(x^n, y^n)\}$ so $\cup_{l=1}^{n+1} \cup_{k=1}^{n+1} \mathcal{F}_n(l, k, x^n, y^n) = \mathcal{X}^n \times \mathcal{Y}^n$. In contrast to streaming point-to-point or side-information coding (cf. (48) with (27)), the partition is now doubly-indexed. To find the dominant error event, we will need to search over both indices. This search is the reason why the streaming exponents differ from the block coding exponents, manifesting itself in the γ parameter of Theorem 4.

We now bound the marginal error probability $\Pr[x^{n-\Delta} \neq \hat{x}^{n-\Delta}]$.

$$\begin{aligned} \Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] &= \sum_{x^n, y^n} \Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta} | x^n = x^n, y^n = y^n] p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \\ &= \sum_{x^n, y^n} p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \left\{ \sum_{l=1}^{n-\Delta} \sum_{k=1}^{n+1} \right. \\ &\quad \left. \Pr \left[\exists (\tilde{x}^n, \tilde{y}^n) \in \mathcal{B}_x(x^n) \times \mathcal{B}_y(y^n) \cap \mathcal{F}_n(l, k, x^n, y^n) \text{ s.t. } S_{l,k}(\tilde{x}^n, \tilde{y}^n) \geq S_{l,k}(x^n, y^n) \right] \right\} \quad (49) \end{aligned}$$

$$\begin{aligned} &= \sum_{l=1}^{n-\Delta} \sum_{k=1}^{n+1} \left\{ \sum_{x^n, y^n} p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \right. \\ &\quad \left. \Pr \left[\exists (\tilde{x}^n, \tilde{y}^n) \in \mathcal{B}_x(x^n) \times \mathcal{B}_y(y^n) \cap \mathcal{F}_n(l, k, x^n, y^n) \text{ s.t. } S_{l,k}(\tilde{x}^n, \tilde{y}^n) \geq S_{l,k}(x^n, y^n) \right] \right\} \quad (50) \end{aligned}$$

where in (49) we decompose the error event according to conditions (i)–(iii) discussed above, and the equality results from the fact that $\mathcal{F}_n(l, k, x^n, y^n) \cap \mathcal{F}_n(l', k', x^n, y^n) = \{\}$, the null set, for $(l, k) \neq (l', k')$. Defining $p_n(l, k)$ as

$$p_n(l, k) = \sum_{x^n, y^n} p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \Pr \left[\exists (\tilde{x}^n, \tilde{y}^n) \in \mathcal{B}_x(x^n) \times \mathcal{B}_y(y^n) \cap \mathcal{F}_n(l, k, x^n, y^n) \text{ s.t. } S_{l,k}(\tilde{x}^n, \tilde{y}^n) \geq S_{l,k}(x^n, y^n) \right]. \quad (51)$$

and substituting the definition into (50) we get

$$\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] = \sum_{l=1}^{n-\Delta} \sum_{k=1}^{n+1} p_n(l, k). \quad (52)$$

6.3 Maximum likelihood decoding

To develop our results for ML decoding we use the joint score function of (45) in (51). With this choice the following lemma, proved in Appendix B, provides an upper bound on $p_n(l, k)$.

Lemma 3

$$\begin{aligned} p_n(l, k) &\leq \exp \left\{ -(n-l+1) E_x(R_x, R_y, \frac{k-l}{n-l+1}) \right\} \quad \text{if } l \leq k, \\ p_n(l, k) &\leq \exp \left\{ -(n-k+1) E_y(R_x, R_y, \frac{l-k}{n-k+1}) \right\} \quad \text{if } l \geq k, \end{aligned} \quad (53)$$

where $E_x(R_x, R_y, \gamma)$ and $E_y(R_x, R_y, \gamma)$ are defined in (22).

Notice that $l, k \leq n$ and that for $l \leq k$ the fraction $\frac{k-l}{n-l+1} \in [0, 1]$ serves as γ in the error exponent $E_x(R_x, R_y, \gamma)$. An analogous discussion holds for $l \geq k$.

We use Lemma 3 together with (52) to bound $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}]$ for two distinct cases. The first, simpler case, is when $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) > \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)$. To bound $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}]$ in this case, we split the sum over the $p_n(l, k)$ into two terms, as is visualized in Figure 9. There are $(n+1) \times (n-\Delta)$ such events to account for. In Figure 9 these are inside the box. The probability of the event within each oval are summed together to give an upper bound on $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}]$. We add extra probabilities outside of the box but within the ovals to make the summation symmetric thus simpler. Those extra error events do not impact the error exponent because this case assumes

that $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \rho, \gamma) \geq \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \rho, \gamma)$. The possible dominant error events are highlighted in Figure 9 . Thus,

$$\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] = \sum_{l=1}^{n-\Delta} \sum_{k=l}^{n+1} p_n(l, k) + \sum_{k=1}^{n-\Delta} \sum_{l=k}^{n+1} p_n(l, k) \quad (54)$$

$$\leq \sum_{l=1}^{n-\Delta} \sum_{k=l}^{n+1} \exp\{-(n-l+1) \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)\} + \sum_{k=1}^{n-\Delta} \sum_{l=k}^{n+1} \exp\{-(n-k+1) \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)\} \quad (55)$$

$$\begin{aligned} &= \sum_{l=1}^{n-\Delta} \left[(n-l+2) \exp\{-(n-l+1) \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)\} \right. \\ &\quad \left. + \sum_{k=1}^{n-\Delta} \left[(n-k+2) \exp\{-(n-k+1) \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)\} \right] \right] \\ &\leq 2 \sum_{l=1}^{n-\Delta} \left[(n-l+2) \exp\{-(n-l+1) \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)\} \right] \end{aligned} \quad (56)$$

$$\leq \sum_{l=1}^{n-\Delta} C_1 \exp\{-(n-l+2) [\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma) - \alpha]\} \quad (57)$$

$$\leq C_2 \exp\{-\Delta [\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma) - \alpha]\} \quad (58)$$

Equation (54) follows directly from (52), in the first term $l \leq k$, in the second term $l \geq k$. In (55), we use Lemma 3. In (56) we use the assumption that $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) > \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)$. In (57) the $\alpha > 0$ results from incorporating the polynomial into the first exponent, and can be chosen as small as desired. Combining terms and summing out the decaying exponential yield the bound (58).

The second, more involved case, is when $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \rho, \gamma) < \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \rho, \gamma)$. To bound $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}]$, we could use the same bounding technique used in the first case. This gives the error exponent $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)$ which is generally smaller than what we can get by dividing the error events in a new grouping shown in Figure 10. In this situation we split (52) into three terms, as visualized in Fig 10. Just as in the first case shown in Fig 9, there are $(n+1) \times (n-\Delta)$ such events to account for (those inside the box). The error events are partitioned into 3 regions. Region 2 and 3 are separated by $k^*(l)$ using a dotted line. In region 3, we add extra probabilities outside of the box but within the ovals to make the summation simpler. Those extra error events do not affect the error exponent as shown in the proof. The possible dominant error events are highlighted in Fig 10. Thus,

$$\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}] \leq \sum_{l=1}^{n-\Delta} \sum_{k=l}^{n+1} p_n(l, k) + \sum_{l=1}^{n-\Delta} \sum_{k=k^*(l)}^{l-1} p_n(l, k) + \sum_{l=1}^{n-\Delta} \sum_{k=1}^{k^*(l)-1} p_n(l, k) \quad (59)$$

Where $\sum_{k=1}^0 p_k = 0$. The lower boundary of Region 2 is $k^*(l) \geq 1$ as a function of n and l :

$$k^*(l) = \max \left\{ 1, n+1 - \left\lceil \frac{\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)}{\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)} \right\rceil (n+1-l) \right\} = \max \{1, n+1 - G(n+1-l)\} \quad (60)$$

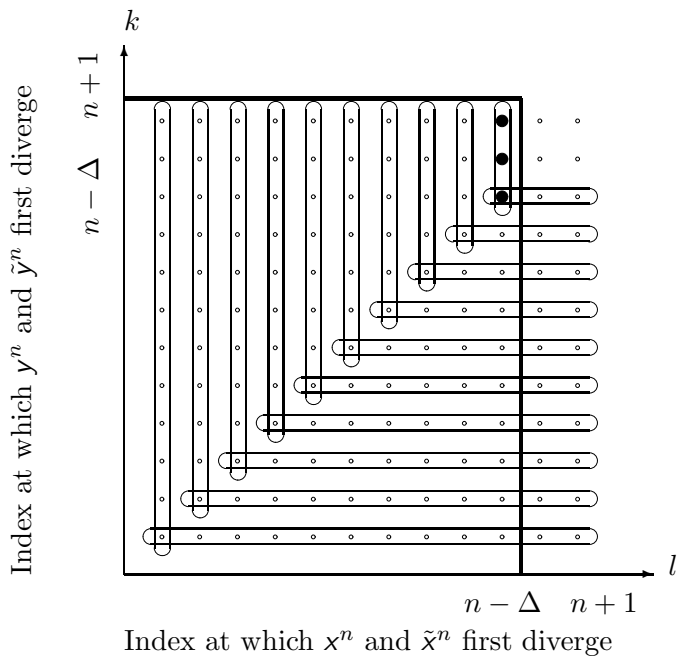


Figure 9: Two dimensional plot of the error probabilities $p_n(l, k)$, corresponding to error events (l, k) , contributing to $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}]$ in the situation where $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \rho, \gamma) \geq \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \rho, \gamma)$.

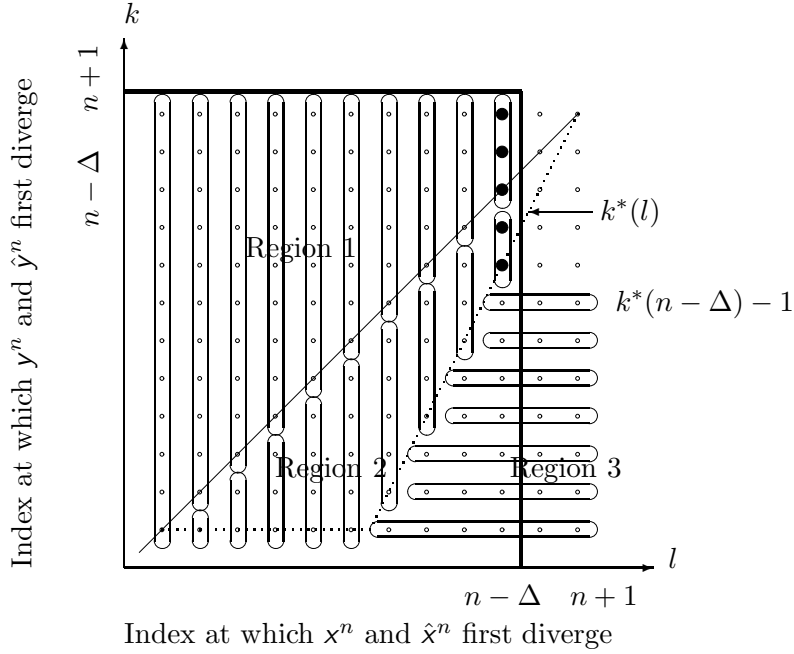


Figure 10: Two dimensional plot of the error probabilities $p_n(l, k)$, corresponding to error events (l, k) , contributing to $\Pr[\hat{x}^{n-\Delta} \neq x^{n-\Delta}]$ in the situation where $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) < \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)$.

where we use G to denote the ceiling of the ratio of exponents. Note that when $\inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) > \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)$ then $G = 1$ and region two of Figure 10 disappears. In other words, the middle term of (59) equals zero. This was the first case considered. We now consider the cases when $G \geq 2$ (because of the ceiling function G is a positive integer).

The first term of (59), i.e., region one in Figure 10 where $l \leq k$, is bounded in the same way that the first term of (54) is, giving

$$\sum_{l=1}^{n-\Delta} \sum_{k=l}^{n+1} p_n(l, k) \leq C_2 \exp\{-\Delta [\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma) - \alpha]\}. \quad (61)$$

In Figure 10, region two is upper bounded by the 45-degree line, and lower bounded by $k^*(l)$. The second term of (59), corresponding to this region where $l \geq k$,

$$\begin{aligned} \sum_{l=1}^{n-\Delta} \sum_{k=k^*(l)}^{l-1} p_n(l, k) &\leq \sum_{l=1}^{n-\Delta} \sum_{k=k^*(l)}^{l-1} \exp\left\{-(n-k+1)E_y\left(R_x, R_y, \frac{l-k}{n-k+1}\right)\right\} \\ &= \sum_{l=1}^{n-\Delta} \sum_{k=k^*(l)}^{l-1} \exp\left\{-(n-k+1)\frac{n-l+1}{n-l+1}E_y\left(R_x, R_y, \frac{l-k}{n-k+1}\right)\right\} \end{aligned} \quad (62)$$

$$\leq \sum_{l=1}^{n-\Delta} \sum_{k=k^*(l)}^{l-1} \exp\left\{-(n-l+1) \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma)\right\} \quad (63)$$

$$= \sum_{l=1}^{n-\Delta} (l-k^*(l)) \exp\left\{-(n-l+1) \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma)\right\} \quad (64)$$

In (62) we note that $l \geq k$, so define $\frac{l-k}{n-k+1} = \gamma$ as in (63). Then $\frac{n-k+1}{n-l+1} = \frac{1}{1-\gamma}$.

The third term of (59), i.e., the intersection of region three and the “box” in Figure 10 where $l \geq k$, can be bounded as,

$$\sum_{l=1}^{n-\Delta} \sum_{k=1}^{k^*(l)-1} p_n(l, k) \leq \sum_{l=1}^{n+1} \sum_{k=1}^{\min\{l, k^*(n-\Delta)-1\}} p_n(l, k) \quad (65)$$

$$= \sum_{k=1}^{k^*(n-\Delta)-1} \sum_{l=k}^{n+1} p_n(l, k) \quad (66)$$

$$\leq \sum_{k=1}^{k^*(n-\Delta)-1} \sum_{l=k}^{n+1} \exp\left\{-(n-k+1)E_y\left(R_x, R_y, \frac{l-k}{n-k+1}\right)\right\}$$

$$\leq \sum_{k=1}^{k^*(n-\Delta)-1} \sum_{l=k}^{n+1} \exp\left\{-(n-k+1) \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)\right\}$$

$$\leq \sum_{k=1}^{k^*(n-\Delta)-1} (n-k+2) \exp\left\{-(n-k+1) \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)\right\} \quad (67)$$

In (65) we note that $l \leq n - \Delta$ thus $k^*(n - \Delta) - 1 \geq k^*(l) - 1$, also $l \geq 1$, so $l \geq k^*(l) - 1$. This can be visualized in Fig 10 as we extend the summation from the intersection of the “box” and region 3 to the whole region under the diagonal line and the horizontal line $k = k^*(n - \Delta) - 1$. In (66) we simply switch the order of the summation.

Finally when $G \geq 2$, we substitute (61), (64), and (67) into (59) to give

$$\begin{aligned}
\Pr[\tilde{x}^{n-\Delta} \neq x^{n-\Delta}] &\leq C_2 \exp\{-\Delta[\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma) - \alpha]\} \\
&+ \sum_{l=1}^{n-\Delta} (l - k^*(l)) \exp\{-(n-l+1) \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma)\} \\
&+ \sum_{k=1}^{k^*(n-\Delta)-1} (n-k+2) \exp\{-(n-k+1) \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)\} \\
&\leq C_2 \exp\{-\Delta[\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma) - \alpha]\} \\
&+ \sum_{l=1}^{n-\Delta} (l - n - 1 + G(n+1-l)) \exp\{-(n-l+1) \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma)\} \\
&+ \sum_{k=1}^{n+1-G(\Delta+1)} (n-k+2) \exp\{-(n-k+1) \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma)\} \\
&\leq C_2 \exp\{-\Delta[\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma) - \alpha]\} \\
&+ (G-1)C_3 \exp\{-\Delta[\inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma) - \alpha]\} \\
&+ C_4 \exp\{-[\Delta G \inf_{\gamma \in [0,1]} E_y(R_x, R_y, \gamma) - \alpha]\} \\
&\leq C_5 \exp\left\{-\Delta \left[\min \left\{ \inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma), \inf_{\gamma \in [0,1]} \frac{1}{1-\gamma} E_y(R_x, R_y, \gamma) \right\} - \alpha \right]\right\}.
\end{aligned} \tag{68}$$

$$\tag{69}$$

$$\tag{70}$$

To get (69), we use the fact that $k^*(l) \geq n+1-G(n+1-l)$ from the definition of $k^*(l)$ in (60) to upper bound the second term. We exploit the definition of G to convert the exponent in the third term to $\inf_{\gamma \in [0,1]} E_x(R_x, R_y, \gamma)$. Finally, to get (70) we gather the constants together, sum out over the decaying exponentials, and are limited by the smaller of the two exponents.

One might note that in this proof we regularly double count the error events and add some extra small probabilities to simplify sums. The error exponent is not modified by these manipulations.

6.4 Universal decoding

To develop our universal results we use the joint universal scoring function $S_{l,k}(\tilde{x}^n, \tilde{y}^n) = 1/H_S(l, k, \tilde{x}^n, \tilde{y}^n)$ from (47) in (51).

$$\begin{aligned}
p_n(l, k) &= \sum_{x^n} \sum_{y^n} p_{\mathbf{xy}}(x^n, y^n) \Pr \left[\exists (\tilde{x}_1^n, \tilde{y}_1^n) \in \mathcal{B}_x(x^n) \times \mathcal{B}_y(y^n) \cap \mathcal{F}_n(l, k, x^n, y^n) \right. \\
&\quad \left. \text{s.t. } H_S(l, k, \tilde{x}^n, \tilde{y}^n) \leq H_S(l, k, x^n, y^n) \right]
\end{aligned} \tag{71}$$

The following lemma bound the contributions of each $p_n(l, k)$ to the overall error probability.

Lemma 4 *Upper bound on $p_n(l, k)$ for $l \leq k$. For all $\eta > 0$, there exists a constant $K_1 < \infty$, s.t.*

$$p_n(l, k) \leq K_1 \exp\{-(n-l+1)[E_x(R_x, R_y, \lambda) - \eta]\}$$

where $\lambda = (k - l)/(n - l + 1) \in [0, 1]$.

Proof: Starting from (71) we have

$$\begin{aligned}
p_n(l, k) &\leq \sum_{x^n, y^n} \min \left[1, \sum_{\substack{(\tilde{x}^n, \tilde{y}^n) \in \mathcal{F}_n(l, k, x^n, y^n) \\ H_S(l, k, \tilde{x}^n, \tilde{y}^n) \leq H_S(l, k, x^n, y^n)}} \Pr[\tilde{x}^n \in \mathcal{B}_x(x^n), \tilde{y}^n \in \mathcal{B}_y(y^n)] \right] p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \\
&\leq \sum_{x_l^n, y_l^n} \min \left[1, \sum_{\substack{(\tilde{x}_l^n, \tilde{y}_l^n) \text{ s.t. } \tilde{y}_l^{k-1} = y_l^{k-1} \\ H_S(\tilde{x}_l^n, \tilde{y}_l^n) \leq H_S(x_l^n, y_l^n)}} \exp\{-(n-l+1)R_x - (n-k+1)R_y\} \right] p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n) \\
&= \sum_{P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l}} \sum_{\substack{y_l^{k-1} \in \mathcal{T}_{P^{k-l}}, \\ y_k^n \in \mathcal{T}_{P^{n-k}}} } \sum_{\substack{x_l^{k-1} \in \mathcal{T}_{V^{k-l}}(y_l^{k-1}), \\ x_k^n \in \mathcal{T}_{V^{n-k}}(y_k^n)}} \min \left[1, \sum_{\substack{\tilde{V}^{n-k}, \tilde{V}^{k-l}, \tilde{P}^{n-k} \text{ s.t.} \\ S(\tilde{P}^{n-k}, P^{k-l}, \tilde{V}^{n-k}, \tilde{V}^{k-l}) < \\ S(P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l})}} \right] p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \\
&\quad \sum_{\tilde{y}_k^n \in \mathcal{T}_{\tilde{P}^{n-k}}} \sum_{\tilde{x}_l^{k-1} \in \mathcal{T}_{\tilde{V}^{k-l}}(y_l^{k-1})} \sum_{\tilde{x}_k^n \in \mathcal{T}_{\tilde{V}^{n-k}}(\tilde{y}_k^n)} \exp\{-(n-l+1)R_x - (n-k+1)R_y\}
\end{aligned} \tag{72}$$

In (72) we enumerate all the source sequences in a way that allows us to focus on the types of the important subsequences. We enumerate the possibly misleading candidate sequences in terms of their suffixes types. We restrict the sum to those pairs $(\tilde{x}^n, \tilde{y}^n)$ that could lead to mistaken decoding, defining the compact notation $S(P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l}) \triangleq (k-l)H(V^{k-l}|P^{k-l}) + (n-k+1)H(P^{n-k} \times V^{n-k})$, which is the weighted empirical suffix entropy condition rewritten in terms of types.

Note that the summations within the minimization in (72) do not depend on the arguments within these sums. Thus, we can bound this sum separately to get a bound on the number of possibly misleading source pairs $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$.

$$\begin{aligned}
&\sum_{\substack{\tilde{V}^{n-k}, \tilde{V}^{k-l}, \tilde{P}^{n-k} \text{ s.t.} \\ S(\tilde{P}^{n-k}, P^{k-l}, \tilde{V}^{n-k}, \tilde{V}^{k-l}) < \\ S(P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l})}} \sum_{\tilde{y}_k^n \in \mathcal{T}_{\tilde{P}^{n-k}}} \sum_{\tilde{x}_l^{k-1} \in \mathcal{T}_{\tilde{V}^{k-l}}(y_l^{k-1})} \sum_{\tilde{x}_k^n \in \mathcal{T}_{\tilde{V}^{n-k}}(\tilde{y}_k^n)} \\
&\leq \sum_{\substack{\tilde{V}^{n-k}, \tilde{V}^{k-l}, \tilde{P}^{n-k} \text{ s.t.} \\ S(\tilde{P}^{n-k}, P^{k-l}, \tilde{V}^{n-k}, \tilde{V}^{k-l}) < \\ S(P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l})}} \sum_{\tilde{y}_k^n \in \mathcal{T}_{\tilde{P}^{n-k}}} |\mathcal{T}_{\tilde{V}^{k-l}}(y_l^{k-1})| |\mathcal{T}_{\tilde{V}^{n-k}}(\tilde{y}_k^n)|
\end{aligned} \tag{73}$$

$$\leq \sum_{\substack{\tilde{V}^{n-k}, \tilde{V}^{k-l}, \tilde{P}^{n-k} \text{ s.t.} \\ S(\tilde{P}^{n-k}, P^{k-l}, \tilde{V}^{n-k}, \tilde{V}^{k-l}) < \\ S(P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l})}} |\mathcal{T}_{\tilde{P}^{n-k}}| \exp\{(k-l)H(\tilde{V}^{k-l}|P^{k-l})\} \exp\{(n-k+1)H(\tilde{V}^{n-k}|\tilde{P}^{n-k})\} \tag{74}$$

$$\leq \sum_{\substack{\tilde{V}^{n-k}, \tilde{V}^{k-l}, \tilde{P}^{n-k} \text{ s.t.} \\ S(\tilde{P}^{n-k}, P^{k-l}, \tilde{V}^{n-k}, \tilde{V}^{k-l}) < \\ S(P^{n-k}, P^{k-l}, V^{n-k}, V^{k-l})}} \exp\{(k-l)H(\tilde{V}^{k-l}|P^{k-l}) + (n-k+1)H(\tilde{P}^{n-k} \times \tilde{V}^{n-k})\} \tag{75}$$

$$\leq \sum_{\tilde{V}^{n-k}, \tilde{V}^{k-l}, \tilde{P}^{n-k}} \exp\{(k-l)H(V^{k-l}|P^{k-l}) + (n-k+1)H(P^{n-k} \times V^{n-k})\} \tag{76}$$

$$\leq (n-l+2)^{2|\mathcal{X}||\mathcal{Y}|} \exp\{(k-l)H(V^{k-l}|P^{k-l}) + (n-k+1)H(P^{n-k} \times V^{n-k})\} \tag{77}$$

In (73) we sum over all $\tilde{x}_l^{k-1} \in \mathcal{T}_{\tilde{V}^{k-l}}(y_l^{k-1})$. In (74) we use standard bounds, e.g., $|\mathcal{T}_{\tilde{V}^{k-l}}(y_l^{k-1})| \leq \exp\{(k-l)H(\tilde{V}^{k-l}|P^{k-l})\}$ since $y_l^{k-1} \in \mathcal{T}_{P^{k-l}}$. We also sum over all $\tilde{x}_k^n \in \mathcal{T}_{\tilde{V}^{n-k}}(\tilde{y}_k^n)$ and over all $\tilde{y}_k^n \in \mathcal{T}_{\tilde{P}^{n-k}}$ in (74). By definition of the decoding rule $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ can only lead to a decoding error if $(k-l)H(\tilde{V}^{k-l}|P^{k-l}) + (n-k+1)H(\tilde{P}^{n-k} \times \tilde{V}^{n-k}) < (k-l)H(V^{k-l}|P^{k-l}) + (n-k+1)H(P^{n-k} \times V^{n-k})$. In (77) we apply the polynomial bound on the number of types.

We substitute (77) into (72) and pull out the polynomial term, giving

$$\begin{aligned}
p_n(l, k) &\leq (n-l+2)^{2^{|\mathcal{X}||\mathcal{Y}|}} \sum_{P^{n-k}, P^{k-l}} \sum_{V^{n-k}, V^{k-l}} \sum_{\substack{y_l^{k-1} \in \mathcal{T}_{P^{k-l}}, \\ y_k^n \in \mathcal{T}_{P^{n-k}}} } \sum_{\substack{x_l^{k-1} \in \mathcal{T}_{V^{k-l}}(y_l^{k-1}), \\ x_k^n \in \mathcal{T}_{V^{n-k}}(y_k^n)}} \\
&\min \left[1, \exp\{-(k-l)[R_x - H(V^{k-l}|P^{k-l})] - (n-k+1)[R_x + R_y - H(V^{n-k} \times P^{n-k})]\} \right] p_{x_l^n, y_l^n}(x_l^n, y_l^n) \\
&\leq (n-l+2)^{2^{|\mathcal{X}||\mathcal{Y}|}} \sum_{P^{n-k}, P^{k-l}} \sum_{V^{n-k}, V^{k-l}} \\
&\exp \left\{ \max \left[0, -(k-l)[R_x - H(V^{k-l}|P^{k-l})] - (n-k+1)[R_x + R_y - H(V^{n-k} \times P^{n-k})] \right] \right\} \\
&\exp \left\{ -(k-l)D(V^{k-l} \times P^{k-l} \| p_{x,y}) - (n-k+1)D(V^{n-k} \times P^{n-k} \| p_{x,y}) \right\} \tag{78} \\
&\leq (n-l+2)^{2^{|\mathcal{X}||\mathcal{Y}|}} \sum_{P^{n-k}, P^{k-l}} \sum_{V^{n-k}, V^{k-l}} \exp \left\{ -(n-l+1) \left[\lambda D(V^{k-l} \times P^{k-l} \| p_{x,y}) + \bar{\lambda} D(V^{n-k} \times P^{n-k} \| p_{x,y}) \right. \right. \\
&\quad \left. \left. + \left| \lambda [R_x - H(V^{k-l}|P^{k-l})] + \bar{\lambda} [R_x + R_y - H(V^{n-k} \times P^{n-k})] \right|^+ \right] \right\} \tag{79} \\
&\leq (n-l+2)^{2^{|\mathcal{X}||\mathcal{Y}|}} \sum_{P^{n-k}, P^{k-l}} \sum_{V^{n-k}, V^{k-l}} \exp \left\{ -(n-l+1) \inf_{\tilde{x}, \tilde{y}, \bar{x}, \bar{y}} \left[\lambda D(p_{\tilde{x}, \tilde{y}} \| p_{x,y}) + \bar{\lambda} D(p_{\bar{x}, \bar{y}} \| p_{x,y}) \right. \right. \\
&\quad \left. \left. + \left| \lambda [R_x - H(\tilde{x}|\tilde{y})] + \bar{\lambda} [R_x + R_y - H(\bar{x}, \bar{y})] \right|^+ \right] \right\} \tag{80} \\
&\leq (n-l+2)^{4^{|\mathcal{X}||\mathcal{Y}|}} \exp\{-(n-l+1)E_x(R_x, R_y, \lambda)\} \leq K_1 \exp\{-(n-l+1)[E_x(R_x, R_y, \lambda) - \eta]\} \tag{81} \\
&\tag{82}
\end{aligned}$$

In (78) we use the memoryless property of the source, and exponential bounds on the probability of observing (x_l^{k-1}, y_l^{k-1}) and (x_k^n, y_k^n) . In (79) we pull out $(n-l+1)$ from all terms, noticing that $\lambda = (k-l)/(n-l+1) \in [0, 1]$ and $\bar{\lambda} \triangleq 1 - \lambda = (n-k+1)/(n-l+1)$. In (80) we minimize the exponent over all choices of distributions $p_{\tilde{x}, \tilde{y}}$ and $p_{\bar{x}, \bar{y}}$. In (81) we define the universal random coding exponent $E_x(R_x, R_y, \lambda) \triangleq \inf_{\tilde{x}, \tilde{y}, \bar{x}, \bar{y}} \{ \lambda D(p_{\tilde{x}, \tilde{y}} \| p_{x,y}) + \bar{\lambda} D(p_{\bar{x}, \bar{y}} \| p_{x,y}) + |\lambda [R_x - H(\tilde{x}|\tilde{y})] + \bar{\lambda} [R_x + R_y - H(\bar{x}, \bar{y})]|^+ \}$ where $0 \leq \lambda \leq 1$ and $\bar{\lambda} = 1 - \lambda$. We also incorporate the number of conditional and marginal types into the polynomial bound, as well as the sum over k , and then push the polynomial into the exponent since for any polynomial F , $\forall E, \epsilon > 0$, there exists $C > 0$, s.t. $F(\Delta)e^{-\Delta E} \leq Ce^{-\Delta(E-\epsilon)}$. ■

A similar derivation yields a bound on $p_n(l, k)$ for $l \geq k$.

Using Lemma 4 in (52) and following the same steps as in the derivation for ML decoding of Section 6.3 yields (23).

7 Future Directions

7.1 Stationary-ergodic sources and universality

In [4] the block-coding proofs of the Slepian-Wolf problem are extended to stationary-ergodic sources using AEP arguments. To have a similar extension to the streaming context it is possible that additional regularity conditions will be required so that error exponents can be achieved. To additionally achieve universality over non-memoryless sources further technical restrictions will be required. For the specific case of distributed Markov sources however, it seems quite clear that all the arguments in this paper will easily generalize by following an approach similar to that taken in [14]: the source can be “segmented” into small blocks and the endpoints (for a Markov source of known order k , the endpoint is just k successive symbols at the end of the block) of the blocks can be encoded perfectly at an arbitrarily small rate by making the “small” blocks long enough. Conditioned on these endpoints, the blocks are then i.i.d. with the endpoints representing a third stream of perfectly known side-information.

7.2 Upper bounds and demonstrating optimal delays

This paper dealt entirely with achievability of certain error exponents. Ideally, we would have corresponding upper bounds demonstrating that no higher exponents are possible. In the block-coding case, problem 3.7.1 in [6] provides a simple upper-bound. However, the nature of the error exponents in the streaming case might be more complicated. In [2] an upper bound and matching achievable scheme for point-to-point source-coding with delay is provided. This bound extends naturally to the case where side-information is known at both the encoder and the decoder. Additionally, in [3] an upper bound is provided for the case of side-information known only at the decoder. This bound is tight for certain symmetric cases. However, both of these papers extend single encoder arguments from [13] and do not immediately generalize to the case of multiple encoders.

7.3 Trading off error exponents for the different source terminals

For multiple terminal systems, different error exponents can be achieved for different users or different sources. For channel coding, the encoders can choose different distributions while generating the randomized code book to achieve an error exponent trade-off among different users. In [17], the error exponent region is studied for the Gaussian multiple access channel and the broadcast channel within the block-coding paradigm. It is unclear whether similar trade offs are possible within the streaming Slepian Wolf problems considered here since there is nothing immediately comparable to the flexibility we have in choosing the “input distribution” for channel coding problems.

7.4 Adaptation and limited feedback

A final interesting extension is to adaptive universal streaming Slepian Wolf encoders. The decoders we use in this paper are based on empirical statistics. Therefore they can be used even if source statistics are unknown. The current proposal will work regardless of source and side information statistics as long as the conditional entropy $H(x|y)$ is less than the encoding rate. Even if there is uncertainty in statistics, the sequential nature of the coding system should enable the system to adapt on-line to the unknown entropy rate if some feedback channel is available. The feedback

channel would be used to order increases (or decreases) in the binning rate. An increase (or decrease) could be triggered by examining the difference between two quantities: the minimal empirical joint entropy between the decoded sequence and observation, and the empirical joint entropy between the particular sequence and observation yielding the second-lowest joint entropy. If there is a large difference between these two entropies, we are using rate excessively, and the rate of communication can be reduced. If the difference is negligible, then it's likely we are not decoding correctly. Our target should be to keep this difference at roughly ϵ . In the current context, this is analogous to the rate margin by which we choose to exceed the known conditional entropy.

Acknowledgments

The authors wish to acknowledge a desire expressed by Zixiang Xiong and subsequent hallway discussions during ITW 2004 that helped precipitate the current line of research. This work was supported in part by NSF ITR Grant CNS-0326503 and CCF-0729122.

A Proof of Lemma 1

In this section we provide the proof of Lemma 1.

$$\begin{aligned}
p_n(l) &= \sum_{x^n} \Pr [\exists \tilde{x}^n \in \mathcal{B}_x(x^n) \cap \mathcal{F}_n(l, x^n) \text{ s.t. } p_{\mathbf{x}}(\tilde{x}_l^n) \geq p_{\mathbf{x}}(x_l^n)] p_{\mathbf{x}}(x^n) \\
&\leq \sum_{x^n} \min \left[1, \sum_{\substack{\tilde{x}^n \in \mathcal{F}_n(l, x^n) \text{ s.t.} \\ p_{\mathbf{x}}(x_l^n) \leq p_{\mathbf{x}}(\tilde{x}_l^n)}} \Pr[\tilde{x}^n \in \mathcal{B}_x(x^n)] \right] p_{\mathbf{x}}(x^n) \tag{83}
\end{aligned}$$

$$\begin{aligned}
&= \sum_{x^{l-1}, x_l^n} \min \left[1, \sum_{\substack{\tilde{x}_l^n \text{ s.t.} \\ p_{\mathbf{x}}(x_l^n) < p_{\mathbf{x}}(\tilde{x}_l^n)}} \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x^{l-1}) p_{\mathbf{x}}(x_l^n) \tag{84}
\end{aligned}$$

$$\begin{aligned}
&= \sum_{x_l^n} \min \left[1, \sum_{\substack{\tilde{x}_l^n \text{ s.t.} \\ p_{\mathbf{x}}(x_l^n) < p_{\mathbf{x}}(\tilde{x}_l^n)}} \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x_l^n) \\
&= \sum_{x_l^n} \min \left[1, \sum_{\tilde{x}_l^n} I[p_{\mathbf{x}}(\tilde{x}_l^n) > p_{\mathbf{x}}(x_l^n)] \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x_l^n) \tag{85}
\end{aligned}$$

$$\begin{aligned}
&\leq \sum_{x_l^n} \min \left[1, \sum_{\tilde{x}_l^n} \min \left[1, \frac{p_{\mathbf{x}}(\tilde{x}_l^n)}{p_{\mathbf{x}}(x_l^n)} \right] \exp\{-(n-l+1)R\} \right] p_{\mathbf{x}}(x_l^n) \\
&\leq \sum_{x_l^n} \left[\sum_{\tilde{x}_l^n} \left[\frac{p_{\mathbf{x}}(\tilde{x}_l^n)}{p_{\mathbf{x}}(x_l^n)} \right]^{\frac{1}{1+\rho}} \exp\{-(n-l+1)R\} \right]^{\rho} p_{\mathbf{x}}(x_l^n) \tag{86}
\end{aligned}$$

$$\begin{aligned}
&= \sum_{x_l^n} p_{\mathbf{x}}(x_l^n)^{\frac{1}{1+\rho}} \left[\sum_{\tilde{x}_l^n} [p_{\mathbf{x}}(\tilde{x}_l^n)]^{\frac{1}{1+\rho}} \right]^{\rho} \exp\{-(n-l+1)\rho R\} \\
&= \left[\sum_x p_{\mathbf{x}}(x)^{\frac{1}{1+\rho}} \right]^{(n-l+1)} \left[\sum_x p_{\mathbf{x}}(x)^{\frac{1}{1+\rho}} \right]^{(n-l+1)\rho} \exp\{-(n-l+1)\rho R\} \tag{87}
\end{aligned}$$

$$\begin{aligned}
&= \left[\sum_x p_{\mathbf{x}}(x)^{\frac{1}{1+\rho}} \right]^{(n-l+1)(1+\rho)} \exp\{-(n-l+1)\rho R\} \\
&= \exp \left\{ -(n-l+1) \left[\rho R - (1+\rho) \ln \left(\sum_x p_{\mathbf{x}}(x)^{\frac{1}{1+\rho}} \right) \right] \right\}. \tag{88}
\end{aligned}$$

In (83) the union bound is applied. In (84) we use the fact that after the first symbol in which two sequences differ, the remaining parity bits are independent. In (85) $1(\cdot)$ is the indicator function, taking the value one if the argument is true, and zero if it is false. We get (86) by limiting ρ to the range $0 \leq \rho \leq 1$ since the arguments of the minimization are both positive and upper-bounded by one. We use the i.i.d. property of the source, exchanging sums and products to get (87). The bound in (88) is true for all ρ in the range $0 \leq \rho \leq 1$. Maximizing (88) over ρ gives $p_n(l) \leq \exp\{-(n-l+1)E_{pt,x}(R)\}$ where $E_{pt,x}(R)$ is defined in Theorem 2, in particular in (15).

B Proof of Lemma 3

In this section we provide the proof of Lemma 3. The bound depends on whether $l \leq k$ or $l \geq k$. Consider the case $l \leq k$,

$$p_n(l, k) = \sum_{x^n, y^n} p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \Pr[\exists (\tilde{x}^n, \tilde{y}^n) \in \mathcal{B}_x(x^n) \times \mathcal{B}_y(y^n) \cap \mathcal{F}_n(l, k, x^n, y^n) \text{ s.t. } p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n) < p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^n, \tilde{y}_l^n)]$$

$$\leq \sum_{x^n, y^n} \min \left[1, \sum_{\substack{(\tilde{x}^n, \tilde{y}^n) \in \mathcal{F}_n(l, k, x^n, y^n) \\ p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n) < p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^n, \tilde{y}_l^n)}} \Pr[\tilde{x}^n \in \mathcal{B}_x(x^n), \tilde{y}^n \in \mathcal{B}_y(y^n)] \right] p_{\mathbf{x}, \mathbf{y}}(x^n, y^n) \tag{89}$$

$$\leq \sum_{x_l^n, y_l^n} \min \left[1, \sum_{\substack{(\tilde{x}_l^n, \tilde{y}_l^n) \text{ s.t. } \tilde{y}_l^{k-1} = y_l^{k-1} \\ p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n) < p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^n, \tilde{y}_l^n)}} \exp\{-(n-l+1)R_x - (n-k+1)R_y\} \right] p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n) \tag{90}$$

$$= \sum_{x_l^n, y_l^n} \min \left[1, \sum_{\tilde{x}_l^n, \tilde{y}_k^n} \exp\{-(n-l+1)R_x - (n-k+1)R_y\} \right. \\ \left. 1[p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^{k-1}, y_l^{k-1}) p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_k^n, \tilde{y}_k^n) > p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n)] \right] p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n)$$

$$\leq \sum_{x_l^n, y_l^n} \min \left[1, \sum_{\tilde{x}_l^n, \tilde{y}_k^n} \exp\{-(n-l+1)R_x - (n-k+1)R_y\} \right. \\ \left. \min \left[1, \frac{p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^{k-1}, y_l^{k-1}) p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_k^n, \tilde{y}_k^n)}{p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n)} \right] \right] p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n)$$

$$\leq \sum_{x_l^n, y_l^n} \left[\sum_{\tilde{x}_l^n, \tilde{y}_k^n} e^{-(n-l+1)R_x - (n-k+1)R_y} \left[\frac{p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^{k-1}, y_l^{k-1}) p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_k^n, \tilde{y}_k^n)}{p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n)} \right]^{\frac{1}{1+\rho}} \right]^\rho p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n) \tag{91}$$

$$= e^{-(n-l+1)\rho R_x - (n-k+1)\rho R_y} \sum_{x_l^n, y_l^n} \left[\sum_{\tilde{x}_l^n, \tilde{y}_k^n} [p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_l^{k-1}, y_l^{k-1}) p_{\mathbf{x}, \mathbf{y}}(\tilde{x}_k^n, \tilde{y}_k^n)]^{\frac{1}{1+\rho}} \right]^\rho p_{\mathbf{x}, \mathbf{y}}(x_l^n, y_l^n)^{\frac{1}{1+\rho}}$$

$$\begin{aligned}
&= e^{-(n-l+1)\rho R_x - (n-k+1)\rho R_y} \sum_{y_l^{k-1}} \left[\sum_{x_l^{k-1}} p_{\mathbf{x},\mathbf{y}}(x_l^{k-1}, y_l^{k-1})^{\frac{1}{1+\rho}} \right] \left[\sum_{\tilde{x}_l^{k-1}} p_{\mathbf{x},\mathbf{y}}(\tilde{x}_l^{k-1}, y_l^{k-1})^{\frac{1}{1+\rho}} \right]^\rho \\
&\quad \left[\sum_{\tilde{x}_k^n, \tilde{y}_k^n} p_{\mathbf{x},\mathbf{y}}(\tilde{x}_k^n, \tilde{y}_k^n)^{\frac{1}{1+\rho}} \right]^\rho \sum_{x_k^n, y_k^n} p_{\mathbf{x},\mathbf{y}}(x_k^n, y_k^n)^{\frac{1}{1+\rho}} \\
&= e^{-(n-l+1)\rho R_x - (n-k+1)\rho R_y} \left[\sum_{y_l^{k-1}} \left[\sum_{x_l^{k-1}} p_{\mathbf{x},\mathbf{y}}(x_l^{k-1}, y_l^{k-1})^{\frac{1}{1+\rho}} \right]^{1+\rho} \right] \left[\sum_{x_k^n, y_k^n} p_{\mathbf{x},\mathbf{y}}(x_k^n, y_k^n)^{\frac{1}{1+\rho}} \right]^{1+\rho} \\
&= e^{-(n-l+1)\rho R_x - (n-k+1)\rho R_y} \left[\sum_y \left[\sum_x p_{\mathbf{x},\mathbf{y}}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right]^{k-l} \left[\sum_{x,y} p_{\mathbf{x},\mathbf{y}}(x, y)^{\frac{1}{1+\rho}} \right]^{(1+\rho)(n-k+1)}
\end{aligned} \tag{92}$$

$$\begin{aligned}
&= \exp \left\{ -(k-l) \left[\rho R_x - \log \left[\sum_y \left[\sum_x p_{\mathbf{x},\mathbf{y}}(x, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} \right] \right] \right\} \\
&\quad \exp \left\{ -(n-k+1) \left[\rho(R_x + R_y) - (1+\rho) \log \left[\sum_{x,y} p_{\mathbf{x},\mathbf{y}}(x, y)^{\frac{1}{1+\rho}} \right] \right] \right\} \\
&= \exp \left\{ -(k-l)E_{x|y}(R_x, \rho) - (n-k+1)E_{xy}(R_x, R_y, \rho) \right\}
\end{aligned} \tag{93}$$

$$= \exp \left\{ -(n-l+1) \left[\frac{k-l}{n-l+1} E_{x|y}(R_x, \rho) + \frac{n-k+1}{n-l+1} E_{xy}(R_x, R_y, \rho) \right] \right\} \tag{94}$$

$$\leq \exp \left\{ -(n-l+1) \sup_{\rho \in [0,1]} \left[\frac{k-l}{n-l+1} E_{x|y}(R_x, \rho) + \frac{n-k+1}{n-l+1} E_{xy}(R_x, R_y, \rho) \right] \right\} \tag{95}$$

$$= \exp \left\{ -(n-l+1) E_x \left(R_x, R_y, \frac{k-l}{n-l+1} \right) \right\}. \tag{96}$$

In (89) we explicitly indicate the three conditions that a suffix pair $(\tilde{x}_k^n, \tilde{y}_k^n)$ must satisfy to result in a decoding error. In (90) we sum out over the common prefixes (x_l^{l-1}, y_l^{l-1}) , and use the fact that the random binning is done independently at each encoder, see Definition. 2. We get (91) by limiting ρ to the interval $0 \leq \rho \leq 1$, as in (86). Getting (92) from (91) follows by a number of basic manipulations. In (92) we get the single letter expression by again using the memoryless property of the sources. In (93) we use the definitions of $E_{x|y}$ and E_{xy} from (7) and (8) of Theorem 4. Noting that the bound holds for all $\rho \in [0, 1]$ optimizing over ρ results in (95). Finally, using the definition of (22) gives (96). The bound on $p_n(l, k)$ when $l > k$, is developed in an analogous fashion. ■

C Equivalence of the two forms of the error exponent for streaming Slepian-Wolf

In this section we prove the following lemma.

Lemma 5

$$E_x(R_x, R_y, \gamma) = \sup_{\rho \in [0,1]} \left\{ \gamma E_{x|y}(R_x, \rho) + (1 - \gamma) E_{xy}(R_x, R_y, \rho) \right\} \quad (97)$$

$$= \inf_{q_{xy}, o_{xy}} \left\{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma) D(o_{xy} \| p_{xy}) \right. \\ \left. + \max\{0, \gamma(R_x - H(q_{x|y})) + (1 - \gamma)(R_x + R_y - H(o_{xy}))\} \right\}, \quad (98)$$

where $E_{x|y}(\cdot)$ and $E_{xy}(\cdot)$ are defined in (7) and (8). For notational simplicity, we write q_{xy} and o_{xy} as two arbitrary joint distributions on $\mathcal{X} \times \mathcal{Y}$ (instead of $p_{\bar{x}\bar{y}}$ and $p_{\bar{x}\bar{y}}$). We retain p_{xy} to indicate the joint distribution of the source.

The equivalence between the forms of $E_y(R_x, R_y, \gamma)$ in (22) and (23) can be proved using the same approach.

The proof of Lemma 5 is given in Section C.2. We start by giving some preliminary definitions in Section C.1. The proofs of a number of technical lemmas are deferred to Section C.3.

C.1 Preliminaries

We recall that the first form of the exponent specified in (97) resulted from the analysis of ML decoding. To help establish the lemma we define the function $E_{ml,x}(R_x, R_y, \gamma, \rho)$ as

$$E_{ml,x}(R_x, R_y, \gamma, \rho) = \gamma E_{x|y}(R_x, \rho) + (1 - \gamma) E_{xy}(R_x, R_y, \rho) \\ = \rho R^{(\gamma)} - \gamma \log \left(\sum_y \left(\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right)^{1+\rho} \right) - (1 - \gamma)(1 + \rho) \log \left(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right),$$

and define $E_{ml,x}(R_x, R_y, \gamma) = \sup_{\rho \in [0,1]} E_{ml,x}(R_x, R_y, \gamma, \rho)$.

In addition, we use $E_{un,x}(R_x, R_y, \gamma)$ to denote the ‘‘universal’’ form (98) of the exponent, i.e.,

$$E_{un,x}(R_x, R_y, \gamma) = \inf_{q_{xy}, o_{xy}} \left\{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma) D(o_{xy} \| p_{xy}) \right. \\ \left. + \max\{0, \gamma(R_x - H(q_{x|y})) + (1 - \gamma)(R_x + R_y - H(o_{xy}))\} \right\} \\ = \inf_{q_{xy}, o_{xy}} \left\{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma) D(o_{xy} \| p_{xy}) + \max\{0, R^{(\gamma)} - \gamma H(q_{x|y}) - (1 - \gamma) H(o_{xy})\} \right\}.$$

To increase compactness we have defined

$$R^{(\gamma)} = \gamma R_x + (1 - \gamma)(R_x + R_y).$$

We note that in the achievable rate region we have the relation

$$R^{(\gamma)} > \gamma H(p_{x|y}) + (1 - \gamma) H(p_{x,y}).$$

Finally, before starting the proof, we define a pair of distributions that we will need.

Definition 5 Tilted distribution of p_{xy} : p_{xy}^ρ , for all $\rho \in [-1, \infty)$

$$p_{xy}^\rho(x, y) = \frac{p_{xy}(x, y)^{\frac{1}{1+\rho}}}{\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}}}. \quad (99)$$

The entropy of the tilted distribution is written as $H(p_{xy}^\rho)$. Obviously $p_{xy}^0 = p_{xy}$.

Definition 6 $x - y$ tilted distribution of p_{xy} : \bar{p}_{xy}^ρ , for all $\rho \in [-1, +\infty)$

$$\begin{aligned} \bar{p}_{xy}^\rho(x, y) &= \frac{\left[\sum_s p_{xy}(s, y)^{\frac{1}{1+\rho}} \right]^{1+\rho}}{\sum_t \left[\sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}} \right]^{1+\rho}} \times \frac{p_{xy}(x, y)^{\frac{1}{1+\rho}}}{\sum_s p_{xy}(s, y)^{\frac{1}{1+\rho}}} \\ &= \frac{A(y, \rho)}{B(\rho)} \times \frac{C(x, y, \rho)}{D(y, \rho)} \end{aligned}$$

Where

$$\begin{aligned} A(y, \rho) &= \left[\sum_s p_{xy}(s, y)^{\frac{1}{1+\rho}} \right]^{1+\rho} = D(y, \rho)^{1+\rho} \\ B(\rho) &= \sum_s \left[\sum_t p_{xy}(s, t)^{\frac{1}{1+\rho}} \right]^{1+\rho} = \sum_y A(y, \rho) \\ C(x, y, \rho) &= p_{xy}(x, y)^{\frac{1}{1+\rho}} \\ D(y, \rho) &= \sum_s p_{xy}(s, y)^{\frac{1}{1+\rho}} = \sum_x C(x, y, \rho) \end{aligned}$$

The marginal distribution for y is $\frac{A(y, \rho)}{B(\rho)}$. Obviously $\bar{p}_{xy}^0 = p_{xy}$. Write the conditional distribution of x given y under distribution \bar{p}_{xy}^ρ as $\bar{p}_{x|y}^\rho$, where $\bar{p}_{x|y}^\rho(x, y) = \frac{C(x, y, \rho)}{D(y, \rho)}$, and the conditional entropy of x given y under distribution \bar{p}_{xy}^ρ as $H(\bar{p}_{x|y}^\rho)$. Obviously $H(\bar{p}_{x|y}^0) = H(p_{x|y})$. The conditional entropy of x given y for the $x - y$ tilted distribution is

$$H(\bar{p}_{x|y=y}^\rho) = - \sum_x \frac{C(x, y, \rho)}{D(y, \rho)} \log \left(\frac{C(x, y, \rho)}{D(y, \rho)} \right) \quad (100)$$

We introduce $A(y, \rho)$, $B(\rho)$, $C(x, y, \rho)$, $D(y, \rho)$ to simplify the notations. Some of their properties are shown in Lemma 9.

While tilted distributions are common optimal distributions in large deviation theory, it is useful to contemplate why we need to introduce these *two* tilted distributions. In the proof of Lemma 5 we show through a Lagrange multiplier argument that $\{p_{xy}^\rho : \rho \in [-1, +\infty)\}$ is the family of distributions that minimize the Kullback-Leibler distance to p_{xy} with fixed *entropy* and $\{\bar{p}_{xy}^\rho : \rho \in [-1, +\infty)\}$ is the family of distributions that minimize the Kullback-Leibler distance to p_{xy} with fixed *conditional entropy*. Using a Lagrange multiplier argument, we parametrize the universal error exponent $E_{un,x}(R_x, R_y, \gamma)$ in terms of ρ and show the equivalence of the universal and maximum likelihood error exponents.

C.2 Proof of Lemma 5

The proof of the lemma splits into two cases. Case 1 is when $\gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy}) < R^{(\gamma)} < \gamma H(\bar{p}_{x|y}^1) + (1 - \gamma)H(p_{xy}^1)$. Case 2 is when $R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^1) + (1 - \gamma)H(p_{xy}^1)$.

Proof:

Case 1: First, from Lemma 15 and Lemma 16:

$$\frac{\partial E_{ml,x}(R_x, R_y, \gamma, \rho)}{\partial \rho} = R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^\rho) - (1 - \gamma)H(p_{xy}^\rho) \quad (101)$$

Then, using Lemma 6 and Lemma 10, we have:

$$\frac{\partial^2 E_{ml,x}(R_x, R_y, \gamma, \rho)}{\partial \rho} \leq 0. \quad (102)$$

So ρ maximize $E_{ml,x}(R_x, R_y, \gamma, \rho)$, if and only if:

$$0 = \frac{\partial E_{ml,x}(R_x, R_y, \gamma, \rho)}{\partial \rho} = R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^\rho) - (1 - \gamma)H(p_{xy}^\rho) \quad (103)$$

Because $R^{(\gamma)}$ is in the interval $[\gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy}), \gamma H(\bar{p}_{x|y}^1) + (1 - \gamma)H(p_{xy}^1)]$ and the entropy functions monotonically-increase over ρ , we can find $\rho^* \in (0, 1)$, s.t.

$$\gamma H(\bar{p}_{x|y}^{\rho^*}) + (1 - \gamma)H(p_{xy}^{\rho^*}) = R^{(\gamma)}. \quad (104)$$

Using Lemma 13 and Lemma 14 we get:

$$E_{ml,x}(R_x, R_y, \gamma) = \gamma D(\bar{p}_{xy}^{\rho^*} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho^*} \| p_{xy}) \quad (105)$$

Where $\gamma H(\bar{p}_{x|y}^{\rho^*}) + (1 - \gamma)H(p_{xy}^{\rho^*}) = R^{(\gamma)}$, ρ^* is generally unique because both $H(\bar{p}_{x|y}^\rho)$ and $H(p_{xy}^\rho)$ are strictly increasing with ρ .

Secondly

$$\begin{aligned} & E_{un,x}(R_x, R_y, \gamma) \\ &= \inf_{q_{xy}, o_{xy}} \{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma)D(o_{xy} \| p_{xy}) + \max\{0, R^{(\gamma)} - \gamma H(q_{x|y}) - (1 - \gamma)H(o_{xy})\} \} \\ &= \inf_b \{ \inf_{q_{xy}, o_{xy}: \gamma H(q_{x|y}) + (1 - \gamma)H(o_{xy}) = b} \{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma)D(o_{xy} \| p_{xy}) + \max(0, R^{(\gamma)} - b) \} \} \\ &= \inf_{b \geq \gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy})} \{ \inf_{q_{xy}, o_{xy}: \gamma H(q_{x|y}) + (1 - \gamma)H(o_{xy}) = b} \{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma)D(o_{xy} \| p_{xy}) \\ & \quad + \max(0, R^{(\gamma)} - b) \} \} \end{aligned} \quad (106)$$

The last equality is true because, for $b < \gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy}) < R^{(\gamma)}$,

$$\begin{aligned}
& \inf_{q_{xy}, o_{xy}: \gamma H(q_{x|y}) + (1-\gamma)H(o_{xy}) = b} \{\gamma D(q_{xy}||p_{xy}) + (1-\gamma)D(o_{xy}||p_{xy}) + \max(0, R^{(\gamma)} - b)\} \\
& \geq 0 + R^{(\gamma)} - b \\
& = \inf_{q_{xy}, o_{xy}: H(q_{x|y}) = H(p_{x|y}), H(o_{xy}) = H(p_{xy})} \{\gamma D(q_{xy}||p_{xy}) + (1-\gamma)D(o_{xy}||p_{xy}) + \max(0, R^{(\gamma)} - b)\} \\
& \geq \inf_{q_{xy}, o_{xy}: H(q_{x|y}) = H(p_{x|y}), H(o_{xy}) = H(p_{xy})} \{\gamma D(q_{xy}||p_{xy}) + (1-\gamma)D(o_{xy}||p_{xy}) \\
& \quad + \max(0, R^{(\gamma)} - \gamma H(p_{x|y}) + (1-\gamma)H(p_{xy}))\} \\
& \geq \inf_{q_{xy}, o_{xy}: \gamma H(q_{x|y}) + (1-\gamma)H(o_{xy}) = \gamma H(p_{x|y}) + (1-\gamma)H(p_{xy})} \{\gamma D(q_{xy}||p_{xy}) + (1-\gamma)D(o_{xy}||p_{xy}) \\
& \quad + \max(0, R^{(\gamma)} - \gamma H(p_{x|y}) + (1-\gamma)H(p_{xy}))\}
\end{aligned}$$

Fixing $b \geq \gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy})$, the inner infimum in (106) is an optimization problem on q_{xy}, o_{xy} with equality constraints $\sum_x \sum_y q_{xy}(x, y) = 1$, $\sum_x \sum_y o_{xy}(x, y) = 1$ and $\gamma H(q_{x|y}) + (1 - \gamma)H(o_{xy}) = b$ and the obvious inequality constraints $0 \leq q_{xy}(x, y) \leq 1, 0 \leq o_{xy}(x, y) \leq 1, \forall x, y$. In the following formulation of the optimization problem, we relax one equality constraint to an inequality constraint $\gamma H(q_{x|y}) + (1 - \gamma)H(o_{xy}) \geq b$ to make the optimization problem *convex*. It turns out later that the optimal solution to the relaxed problem is also the optimal solution to the original problem because $b \geq \gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy})$. The resulting optimization problem is:

$$\begin{aligned}
& \inf_{q_{xy}, o_{xy}} \{\gamma D(q_{xy}||p_{xy}) + (1-\gamma)D(o_{xy}||p_{xy})\} \\
& \text{s.t. } \sum_x \sum_y q_{xy}(x, y) = 1 \\
& \quad \sum_x \sum_y o_{xy}(x, y) = 1 \\
& \quad b - \gamma H(q_{x|y}) - (1-\gamma)H(o_{xy}) \leq 0 \\
& \quad 0 \leq q_{xy}(x, y) \leq 1, \quad \forall (x, y) \in \mathcal{X} \times \mathcal{Y} \\
& \quad 0 \leq o_{xy}(x, y) \leq 1, \quad \forall (x, y) \in \mathcal{X} \times \mathcal{Y}
\end{aligned} \tag{107}$$

The above optimization problem is *convex* because the objective function and the inequality constraint functions are convex and the equality constraint functions are affine[1]. The Lagrange multiplier function for this convex optimization problem is:

$$\begin{aligned}
& L(q_{xy}, o_{xy}, \rho, \mu_1, \mu_2, \nu_1, \nu_2, \nu_3, \nu_4) \\
& = \gamma D(q_{xy}||p_{xy}) + (1-\gamma)D(o_{xy}||p_{xy}) \\
& \quad + \mu_1(\sum_x \sum_y q_{xy}(x, y) - 1) + \mu_2(\sum_x \sum_y o_{xy}(x, y) - 1) \\
& \quad + \rho(b - \gamma H(q_{x|y}) - (1-\gamma)H(o_{xy})) \\
& \quad + \sum_x \sum_y \{\nu_1(x, y)(-q_{xy}(x, y)) + \nu_2(x, y)(1 - q_{xy}(x, y)) + \nu_3(x, y)(-o_{xy}(x, y)) + \nu_4(x, y)(1 - o_{xy}(x, y))\}
\end{aligned} \tag{108}$$

Where ρ, μ_1, μ_2 are real numbers and $\nu_i \in R^{|\mathcal{X}||\mathcal{Y}|}$, $i = 1, 2, 3, 4$.

According to the KKT conditions for convex optimization[1], q_{xy}, o_{xy} minimize the convex optimization problem in (107) if and only if the following conditions are simultaneously satisfied for some $q_{xy}, o_{xy}, \mu_1, \mu_2, \nu_1, \nu_2, \nu_3, \nu_4$ and ρ :

$$\begin{aligned}
0 &= \frac{\partial L(q_{xy}, o_{xy}, \rho, \mu_1, \mu_2, \nu_1, \nu_2, \nu_3, \nu_4)}{\partial q_{xy}(x, y)} \\
&= \gamma[-\log(p_{xy}(x, y)) + (1 + \rho)(1 + \log(q_{xy}(x, y))) + \rho \log(\sum_s q_{xy}(s, y))] + \mu_1 - \nu_1(x, y) - \nu_2(x, y) \\
0 &= \frac{\partial L(q_{xy}, o_{xy}, \rho, \mu_1, \mu_2, \nu_1, \nu_2, \nu_3, \nu_4)}{\partial o_{xy}(x, y)} \\
&= (1 - \gamma)[- \log(p_{xy}(x, y)) + (1 + \rho)(1 + \log(o_{xy}(x, y)))] + \mu_2 - \nu_3(x, y) - \nu_4(x, y)
\end{aligned} \tag{109}$$

For all x, y and

$$\begin{aligned}
\sum_x \sum_y q_{xy}(x, y) &= 1 \\
\sum_x \sum_y o_{xy}(x, y) &= 1 \\
\rho(\gamma H(q_{x|y}) + (1 - \gamma)H(o_{xy}) - b) &= 0 \\
\rho &\geq 0 \\
\nu_1(x, y)(-q_{xy}(x, y)) &= 0, \quad \nu_2(x, y)(1 - q_{xy}(x, y)) = 0 \quad \forall x, y \\
\nu_3(x, y)(-o_{xy}(x, y)) &= 0, \quad \nu_4(x, y)(1 - o_{xy}(x, y)) = 0 \quad \forall x, y \\
\nu_i(x, y) &\geq 0, \quad \forall x, y, i = 1, 2, 3, 4
\end{aligned} \tag{110}$$

Solving the above standard Lagrange multiplier equations (109) and (110), we have:

$$\begin{aligned}
q_{xy}(x, y) &= \frac{[\sum_s p_{xy}(s, y)^{\frac{1}{1+\rho_b}}]^{1+\rho_b} p_{xy}(x, y)^{\frac{1}{1+\rho_b}}}{\sum_t [\sum_s p_{xy}(s, t)^{\frac{1}{1+\rho_b}}]^{1+\rho_b} \sum_s p_{xy}(s, y)^{\frac{1}{1+\rho_b}}} \\
&= \bar{p}_{xy}^{\rho_b}(x, y) \\
o_{xy}(x, y) &= \frac{p_{xy}(x, y)^{\frac{1}{1+\rho_b}}}{\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho_b}}} \\
&= p_{xy}^{\rho_b}(x, y) \\
\nu_i(x, y) &= 0 \quad \forall x, y, i = 1, 2, 3, 4 \\
\rho &= \rho_b
\end{aligned} \tag{111}$$

Where ρ_b satisfies the following condition

$$\gamma H(\bar{p}_{x|y}^{\rho_b}) + (1 - \gamma)H(p_{xy}^{\rho_b}) = b \geq \gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy}) \tag{112}$$

and thus $\rho_b \geq 0$ because both $H(\bar{p}_{x|y}^{\rho})$ and $H(p_{xy}^{\rho})$ are monotonically increasing with ρ as shown in Lemma 6 and Lemma 10.

Notice that all the KKT conditions are simultaneously satisfied with the inequality constraint $\gamma H(q_{x|y}) + (1 - \gamma)H(o_{xy}) \geq b$ being met with equality. Thus, the relaxed optimization problem has

the same optimal solution as the original problem as promised. The optimal q_{xy} and o_{xy} are the $x - y$ tilted distribution $\bar{p}_{xy}^{\rho_b}$ and standard tilted distribution $p_{xy}^{\rho_b}$ of p_{xy} with the same parameter $\rho_b \geq 0$. chosen s.t.

$$\gamma H(\bar{p}_{x|y}^{\rho_b}) + (1 - \gamma)H(p_{xy}^{\rho_b}) = b \quad (113)$$

Now we have :

$$\begin{aligned} & E_{un,x}(R_x, R_y, \gamma) \\ &= \inf_{b \geq \gamma H(p_{x|y}) + (1-\gamma)H(p_{xy})} \left\{ \inf_{q_{xy}, o_{xy}: \gamma H(q_{x|y}) + (1-\gamma)H(o_{xy}) = b} \{ \gamma D(q_{xy} \| p_{xy}) + (1 - \gamma)D(o_{xy} \| p_{xy}) + \max(0, R^{(\gamma)} - b) \} \right\} \\ &= \inf_{b \geq \gamma H(p_{x|y}) + (1-\gamma)H(p_{xy})} \{ \gamma D(\bar{p}_{xy}^{\rho_b} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho_b} \| p_{xy}) + \max(0, R^{(\gamma)} - b) \} \\ &= \min_{\rho \geq 0: R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \left[\inf_{\rho \geq 0: R^{(\gamma)} \leq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{ \gamma D(\bar{p}_{xy}^{\rho} \| p_{xy}) + (1 - \gamma)D(p_{xy\rho} \| p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^{\rho}) - (1 - \gamma)H(p_{xy}^{\rho}) \}, \right. \\ & \quad \left. \inf_{\rho \geq 0: R^{(\gamma)} \leq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{ \gamma D(\bar{p}_{xy}^{\rho} \| p_{xy}) + (1 - \gamma)D(p_{xy\rho} \| p_{xy}) \} \right] \quad (114) \end{aligned}$$

Notice that $H(p_{xy}^{\rho})$, $H(\bar{p}_{x|y}^{\rho})$, $D(\bar{p}_{xy}^{\rho} \| p_{xy})$ and $D(p_{xy}^{\rho} \| p_{xy})$ are all strictly increasing with $\rho > 0$ as shown in Lemma 10, Lemma 11, Lemma 6 and Lemma 7 later in this appendix. We have:

$$\begin{aligned} & \inf_{\rho \geq 0: R^{(\gamma)} \leq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{ \gamma D(\bar{p}_{xy}^{\rho} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} \| p_{xy}) \} \\ &= \gamma D(\bar{p}_{xy}^{\rho^*} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho^*} \| p_{xy}) \quad (115) \end{aligned}$$

where $R^{(\gamma)} = \gamma H(\bar{p}_{x|y}^{\rho^*}) + (1 - \gamma)H(p_{xy}^{\rho^*})$. Applying the results in Lemma 12 and Lemma 8, we get:

$$\begin{aligned} & \inf_{\rho \geq 0: R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{ \gamma D(\bar{p}_{xy}^{\rho} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} \| p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^{\rho}) - (1 - \gamma)H(p_{xy}^{\rho}) \} \\ &= \gamma D(\bar{p}_{xy}^{\rho} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} \| p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^{\rho}) - (1 - \gamma)H(p_{xy}^{\rho})|_{\rho=\rho^*} \\ &= \gamma D(\bar{p}_{xy}^{\rho^*} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho^*} \| p_{xy}) \quad (116) \end{aligned}$$

This is true because for $\rho : R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^{\rho}) + (1 - \gamma)H(p_{xy}^{\rho})$, we know $\rho \leq 1$ because of the range of $R^{(\gamma)}$: $R^{(\gamma)} < \gamma H(\bar{p}_{x|y}^1) + (1 - \gamma)H(p_{xy}^1)$. Substituting (115) and (116) into (114), we get

$$\begin{aligned} E_{un,x}(R_x, R_y, \gamma) &= \gamma D(\bar{p}_{xy}^{\rho^*} \| p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho^*} \| p_{xy}) \\ & \quad \text{where } R^{(\gamma)} = \gamma H(\bar{p}_{x|y}^{\rho^*}) + (1 - \gamma)H(p_{xy}^{\rho^*}) \quad (117) \end{aligned}$$

So for $\gamma H(p_{x|y}) + (1 - \gamma)H(p_{xy}) \leq R^{(\gamma)} \leq \gamma H(\bar{p}_{x|y}^1) + (1 - \gamma)H(p_{xy}^1)$, from (105) we have the desired property:

$$E_{ml,x}(R_x, R_y, \gamma) = E_{un,x}(R_x, R_y, \gamma) \quad (118)$$

Case 2: Recall that this is the case where $R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^1) + (1 - \gamma)H(p_{xy}^1)$. In this case, for all $0 \leq \rho \leq 1$

$$\frac{\partial E_{ml,x}(R_x, R_y, \gamma, \rho)}{\partial \rho} = R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^{\rho}) - (1 - \gamma)H(p_{xy}^{\rho}) \geq R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^1) - (1 - \gamma)H(p_{xy}^1) \geq 0. \quad (119)$$

So ρ takes value 1 to maximize the error exponent $E_{ml,x}(R_x, R_y, \gamma, \rho)$, thus

$$E_{ml,x}(R_x, R_y, \gamma) = R^{(\gamma)} - \gamma \log\left(\sum_y \left(\sum_x p_{xy}(x, y)^{\frac{1}{2}}\right)^2\right) - 2(1 - \gamma) \log\left(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{2}}\right) \quad (120)$$

Using the same convex optimization techniques as case 1, we notice the fact that $\rho^* \geq 1$ for $R^{(\gamma)} = \gamma H(\bar{p}_{x|y}^{\rho^*}) + (1 - \gamma)H(p_{xy}^{\rho^*})$. Then applying Lemma 12 and Lemma 8, we have:

$$\begin{aligned} & \inf_{\rho \geq 0: R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{\gamma D(\bar{p}_{xy}^{\rho} || p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} || p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^{\rho}) - (1 - \gamma)H(p_{xy}^{\rho})\}, \\ & = \gamma D(\bar{p}_{xy}^1 || p_{xy}) + (1 - \gamma)D(p_{xy}^1 || p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^1) - (1 - \gamma)H(p_{xy}^1) \end{aligned}$$

And

$$\begin{aligned} & \inf_{\rho \geq 0: R^{(\gamma)} \leq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{\gamma D(\bar{p}_{xy}^{\rho} || p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} || p_{xy})\} \\ & = \gamma D(\bar{p}_{xy}^* || p_{xy}) + (1 - \gamma)D(p_{xy}^* || p_{xy}) \\ & = \gamma D(\bar{p}_{xy}^* || p_{xy}) + (1 - \gamma)D(p_{xy}^* || p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^*) - (1 - \gamma)H(p_{xy}^*) \\ & \leq \gamma D(\bar{p}_{xy}^1 || p_{xy}) + (1 - \gamma)D(p_{xy}^1 || p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^1) - (1 - \gamma)H(p_{xy}^1) \end{aligned}$$

Finally:

$$\begin{aligned} & E_{un,x}(R_x, R_y, \gamma) \\ & = \inf_{b \geq \gamma H(\bar{p}_{x|y}) + (1-\gamma)H(p_{xy})} \left\{ \inf_{q_{xy}, o_{xy}: \gamma H(q_{x|y}) + (1-\gamma)H(o_{xy}) = b} \{\gamma D(q_{xy} || p_{xy}) + (1 - \gamma)D(o_{xy} || p_{xy}) + \max(0, R^{(\gamma)} - b)\} \right\} \\ & = \inf_{b \geq \gamma H(\bar{p}_{x|y}) + (1-\gamma)H(p_{xy})} \{\gamma D(\bar{p}_{xy}^b || p_{xy}) + (1 - \gamma)D(p_{xy}^b || p_{xy}) + \max(0, R^{(\gamma)} - b)\} \\ & = \min \left[\inf_{\rho \geq 0: R^{(\gamma)} \geq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{\gamma D(\bar{p}_{xy}^{\rho} || p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} || p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^{\rho}) - (1 - \gamma)H(p_{xy}^{\rho})\}, \right. \\ & \quad \left. \inf_{\rho \geq 0: R^{(\gamma)} \leq \gamma H(\bar{p}_{x|y}^{\rho}) + (1-\gamma)H(p_{xy}^{\rho})} \{\gamma D(\bar{p}_{xy}^{\rho} || p_{xy}) + (1 - \gamma)D(p_{xy}^{\rho} || p_{xy})\} \right] \\ & = \gamma D(\bar{p}_{xy}^1 || p_{xy}) + (1 - \gamma)D(p_{xy}^1 || p_{xy}) + R^{(\gamma)} - \gamma H(\bar{p}_{x|y}^1) - (1 - \gamma)H(p_{xy}^1) \\ & = R^{(\gamma)} - \gamma \log\left(\sum_y \left(\sum_x p_{xy}(x, y)^{\frac{1}{2}}\right)^2\right) - 2(1 - \gamma) \log\left(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{2}}\right) \quad (121) \end{aligned}$$

The last equality is true by setting $\rho = 1$ in Lemma 13 and Lemma 14.

Again, $E_{ml,x}(R_x, R_y, \gamma) = E_{un,x}(R_x, R_y, \gamma)$, thus we finish the proof. ■

C.3 Technical Lemmas

Some technical lemmas we used in the proof of Lemma 5 given above are now discussed.

Lemma 6 $\frac{\partial H(p_{xy}^{\rho})}{\partial \rho} \geq 0$

Proof: From the definition of the tilted distribution we have the following observation:

$$\log(p_{xy}^\rho(x_1, y_1)) - \log(p_{xy}^\rho(x_2, y_2)) = \log(p_{xy}(x_1, y_1)^{\frac{1}{1+\rho}}) - \log(p_{xy}(x_2, y_2)^{\frac{1}{1+\rho}})$$

Using the above equality, we first derive the derivative of the tilted distribution, for all x, y

$$\begin{aligned} \frac{\partial p_{xy}^\rho(x, y)}{\partial \rho} &= \frac{-1}{(1+\rho)^2} \frac{p_{xy}(x, y)^{\frac{1}{1+\rho}} \log(p_{xy}(x, y)) (\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}})}{(\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}})^2} \\ &\quad - \frac{-1}{(1+\rho)^2} \frac{p_{xy}(x, y)^{\frac{1}{1+\rho}} (\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}} \log(p_{xy}(s, t)))}{(\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}})^2} \\ &= \frac{-1}{1+\rho} p_{xy}^\rho(x, y) [\log(p_{xy}(x, y)^{\frac{1}{1+\rho}}) - \sum_t \sum_s p_{xy}^\rho(s, t) \log(p_{xy}(s, t)^{\frac{1}{1+\rho}})] \\ &= \frac{-1}{1+\rho} p_{xy}^\rho(x, y) [\log(p_{xy}^\rho(x, y)) - \sum_t \sum_s p_{xy}^\rho(s, t) \log(p_{xy}^\rho(s, t))] \\ &= -\frac{p_{xy}^\rho(x, y)}{1+\rho} [\log(p_{xy}^\rho(x, y)) + H(p_{xy}^\rho)] \end{aligned} \tag{122}$$

Then:

$$\begin{aligned} \frac{\partial H(p_{xy}^\rho)}{\partial \rho} &= -\frac{\partial \sum_{x,y} p_{xy}^\rho(x, y) \log(p_{xy}^\rho(x, y))}{\partial \rho} \\ &= -\sum_{x,y} (1 + \log(p_{xy}^\rho(x, y))) \frac{\partial p_{xy}^\rho(x, y)}{\partial \rho} \\ &= \sum_{x,y} (1 + \log(p_{xy}^\rho(x, y))) \frac{p_{xy}^\rho(x, y)}{1+\rho} (\log(p_{xy}^\rho(x, y)) + H(p_{xy}^\rho)) \\ &= \frac{1}{1+\rho} \sum_{x,y} p_{xy}^\rho(x, y) \log(p_{xy}^\rho(x, y)) (\log(p_{xy}^\rho(x, y)) + H(p_{xy}^\rho)) \\ &= \frac{1}{1+\rho} [\sum_{x,y} p_{xy}^\rho(x, y) (\log(p_{xy}^\rho(x, y)))^2 - H(p_{xy}^\rho)^2] \\ &= \frac{1}{1+\rho} [\sum_{x,y} p_{xy}^\rho(x, y) (\log(p_{xy}^\rho(x, y)))^2 \sum_{x,y} p_{xy}^\rho(x, y) - H(p_{xy}^\rho)^2] \\ &\stackrel{(a)}{\geq} \frac{1}{1+\rho} [(\sum_{x,y} p_{xy}^\rho(x, y) \log(p_{xy}^\rho(x, y)))^2 - H(p_{xy}^\rho)^2] \\ &= 0 \end{aligned} \tag{123}$$

where (a) is true by the Cauchy-Schwartz inequality. ■

Lemma 7 $\frac{\partial D(p_{xy}^\rho \| P)}{\partial \rho} = \rho \frac{\partial H(p_{xy}^\rho)}{\partial \rho}$

Proof: As shown in Lemma 13 and Lemma 15 respectively:

$$D(p_{xy}^\rho \| p_{xy}) = \rho H(p_{xy}^\rho) - (1+\rho) \log\left(\sum_{x,y} p_{xy}(x, y)^{\frac{1}{1+\rho}}\right) \tag{124}$$

$$H(p_{xy}^\rho) = \frac{\partial(1 + \rho) \log(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})}{\partial \rho} \quad (125)$$

We have:

$$\begin{aligned} \frac{\partial D(p_{xy}^\rho \| p_{xy})}{\partial \rho} &= H(p_{xy}^\rho) + \rho \frac{\partial H(p_{xy}^\rho)}{\partial \rho} - \frac{\partial(1 + \rho) \log(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})}{\partial \rho} \\ &= H(p_{xy}^\rho) + \rho \frac{\partial H(p_{xy}^\rho)}{\partial \rho} - H(p_{xy}^\rho) \\ &= \rho \frac{\partial H(p_{xy}^\rho)}{\partial \rho} \end{aligned} \quad (126)$$

■

Lemma 8 $\text{sign} \frac{\partial[D(p_{xy}^\rho \| p_{xy}) - H(p_{xy}^\rho)]}{\partial \rho} = \text{sign}(\rho - 1)$.

Proof: Combining the results of the previous two lemmas, we have:

$$\frac{\partial D(p_{xy}^\rho \| p_{xy}) - H(p_{xy}^\rho)}{\partial \rho} = (\rho - 1) \frac{\partial H(p_{xy}^\rho)}{\partial \rho} = \text{sign}(\rho - 1)$$

■

Lemma 9 Properties of $\frac{\partial A(y, \rho)}{\partial \rho}$, $\frac{\partial B(\rho)}{\partial \rho}$, $\frac{\partial C(x, y, \rho)}{\partial \rho}$, $\frac{\partial D(y, \rho)}{\partial \rho}$ and $\frac{\partial H(\bar{p}_{x|y=y}^\rho)}{\partial \rho}$

First,

$$\begin{aligned} \frac{\partial C(x, y, \rho)}{\partial \rho} &= \frac{\partial p_{xy}(x, y)^{\frac{1}{1+\rho}}}{\partial \rho} = -\frac{1}{1+\rho} p_{xy}(x, y)^{\frac{1}{1+\rho}} \log(p_{xy}(x, y)^{\frac{1}{1+\rho}}) \\ &= -\frac{C(x, y, \rho)}{1+\rho} \log(C(x, y, \rho)) \\ \frac{\partial D(y, \rho)}{\partial \rho} &= \frac{\partial \sum_s p_{xy}(s, y)^{\frac{1}{1+\rho}}}{\partial \rho} = -\frac{1}{1+\rho} \sum_s p_{xy}(s, y)^{\frac{1}{1+\rho}} \log(p_{xy}(s, y)^{\frac{1}{1+\rho}}) \\ &= -\frac{\sum_x C(x, y, \rho) \log(C(x, y, \rho))}{1+\rho} \end{aligned} \quad (127)$$

For a differentiable function $f(\rho)$,

$$\frac{\partial f(\rho)^{1+\rho}}{\partial \rho} = f(\rho)^{1+\rho} \log(f(\rho)) + (1 + \rho) f(\rho)^\rho \frac{\partial f(\rho)}{\partial \rho} \quad (128)$$

So

$$\begin{aligned}
\frac{\partial A(y, \rho)}{\partial \rho} &= \frac{\partial D(y, \rho)^{1+\rho}}{\partial \rho} = D(y, \rho)^{1+\rho} \log(D(y, \rho)) + (1 + \rho)D(y, \rho)^\rho \frac{\partial D(y, \rho)}{\partial \rho} \\
&= D(y, \rho)^{1+\rho} (\log(D(y, \rho)) - \sum_x \frac{C(x, y, \rho)}{D(y, \rho)} \log(C(x, y, \rho))) \\
&= D(y, \rho)^{1+\rho} (-\sum_x \frac{C(x, y, \rho)}{D(y, \rho)} \log(\frac{C(x, y, \rho)}{D(y, \rho)})) \\
&= A(y, \rho) H(\bar{p}_{x|y=y}^\rho) \\
\frac{\partial B(\rho)}{\partial \rho} &= \sum_y \frac{\partial A(y, \rho)}{\partial \rho} = \sum_y A(y, \rho) H(\bar{p}_{x|y=y}^\rho) = B(\rho) \sum_y \frac{A(y, \rho)}{B(\rho)} H(\bar{p}_{x|y=y}^\rho) = B(\rho) H(\bar{p}_{x|y}^\rho)
\end{aligned}$$

And last:

$$\begin{aligned}
&\frac{\partial H(\bar{p}_{x|y=y}^\rho)}{\partial \rho} \\
&= -\sum_x \left[\frac{\frac{\partial C(x, y, \rho)}{\partial \rho}}{D(y, \rho)} - \frac{C(x, y, \rho) \frac{\partial D(y, \rho)}{\partial \rho}}{D(y, \rho)^2} \right] \left[1 + \log\left(\frac{C(x, y, \rho)}{D(y, \rho)}\right) \right] \\
&= -\sum_x \left[\frac{-\frac{C(x, y, \rho)}{1+\rho} \log(C(x, y, \rho))}{D(y, \rho)} + \frac{C(x, y, \rho) \sum_s \frac{C(s, y, \rho) \log(C(s, y, \rho))}{1+\rho}}{D(y, \rho)^2} \right] \left[1 + \log\left(\frac{C(x, y, \rho)}{D(y, \rho)}\right) \right] \\
&= \frac{1}{1+\rho} \sum_x \left[\bar{p}_{x|y}^\rho(x, y) \log(C(x, y, \rho)) - \bar{p}_{x|y}^\rho(x, y) \sum_s \bar{p}_{x|y}^\rho(s, y) \log(C(s, y, \rho)) \right] \left[1 + \log(\bar{p}_{x|y}^\rho(x, y)) \right] \\
&= \frac{1}{1+\rho} \sum_x \bar{p}_{x|y}^\rho(x, y) \left[\log(\bar{p}_{x|y}^\rho(x, y)) - \sum_s \bar{p}_{x|y}^\rho(s, y) \log(\bar{p}_{x|y}^\rho(s, y)) \right] \left[1 + \log(\bar{p}_{x|y}^\rho(x, y)) \right] \\
&= \frac{1}{1+\rho} \sum_x \bar{p}_{x|y}^\rho(x, y) \log(\bar{p}_{x|y}^\rho(x, y)) \left[\log(\bar{p}_{x|y}^\rho(x, y)) - \sum_s \bar{p}_{x|y}^\rho(s, y) \log(\bar{p}_{x|y}^\rho(s, y)) \right] \\
&= \frac{1}{1+\rho} \sum_x \bar{p}_{x|y}^\rho(x, y) \log(\bar{p}_{x|y}^\rho(x, y)) \log(\bar{p}_{x|y}^\rho(x, y)) - \frac{1}{1+\rho} \left[\sum_x \bar{p}_{x|y}^\rho(x, y) \log(\bar{p}_{x|y}^\rho(x, y)) \right]^2 \\
&\geq 0 \tag{129}
\end{aligned}$$

The inequality is true by the Cauchy-Schwartz inequality and by noticing that $\sum_x \bar{p}_{x|y}^\rho(x, y) = 1$.

■

These properties will again be used in the proofs in the following lemmas.

Lemma 10 $\frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho} \geq 0$

Proof:

$$\begin{aligned}
\frac{\partial \frac{A(y, \rho)}{B(\rho)}}{\partial \rho} &= \frac{1}{B(\rho)^2} \left(\frac{\partial A(y, \rho)}{\partial \rho} B(\rho) - \frac{\partial B(\rho)}{\partial \rho} A(y, \rho) \right) \\
&= \frac{1}{B(\rho)^2} (A(y, \rho) H(\bar{p}_{x|y=y}^\rho) B(\rho) - H(\bar{p}_{x|y}^\rho) B(\rho) A(y, \rho)) \\
&= \frac{A(y, \rho)}{B(\rho)} (H(\bar{p}_{x|y=y}^\rho) - H(\bar{p}_{x|y}^\rho))
\end{aligned}$$

Now,

$$\begin{aligned}
\frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho} &= \frac{\partial}{\partial \rho} \sum_y \frac{A(y, \rho)}{B(\rho)} \sum_x \frac{C(x, y, \rho)}{D(y, \rho)} [-\log(\frac{C(x, y, \rho)}{D(y, \rho)})] \\
&= \frac{\partial}{\partial \rho} \sum_y \frac{A(y, \rho)}{B(\rho)} H(\bar{p}_{x|y=y}^\rho) \\
&= \sum_y \frac{A(y, \rho)}{B(\rho)} \frac{\partial H(\bar{p}_{x|y=y}^\rho)}{\partial \rho} + \sum_y \frac{\partial \frac{A(y, \rho)}{B(\rho)}}{\partial \rho} H(\bar{p}_{x|y=y}^\rho) \\
&\geq \sum_y \frac{\partial \frac{A(y, \rho)}{B(\rho)}}{\partial \rho} H(\bar{p}_{x|y=y}^\rho) \\
&= \sum_y \frac{A(y, \rho)}{B(\rho)} (H(\bar{p}_{x|y=y}^\rho) - H(\bar{p}_{x|y}^\rho)) H(\bar{p}_{x|y=y}^\rho) \\
&= \sum_y \frac{A(y, \rho)}{B(\rho)} H(\bar{p}_{x|y=y}^\rho)^2 - H(\bar{p}_{x|y}^\rho)^2 \\
&= (\sum_y \frac{A(y, \rho)}{B(\rho)} H(\bar{p}_{x|y=y}^\rho)^2) (\sum_y \frac{A(y, \rho)}{B(\rho)}) - H(\bar{p}_{x|y}^\rho)^2 \\
&\geq_{(a)} (\sum_y \frac{A(y, \rho)}{B(\rho)} H(\bar{p}_{x|y=y}^\rho)^2) - H(\bar{p}_{x|y}^\rho)^2 \\
&= 0
\end{aligned} \tag{130}$$

where (a) is again true by the Cauchy-Schwartz inequality. ■

Lemma 11 $\frac{\partial D(\bar{p}_{xy}^\rho \| p_{xy})}{\partial \rho} = \rho \frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho}$

Proof: As shown in Lemma 14 and Lemma 16 respectively:

$$D(\bar{p}_{xy}^\rho \| p_{xy}) = \rho H(\bar{p}_{x|y}^\rho) - \log(\sum_y (\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})^{1+\rho}) \tag{131}$$

$$H(\bar{p}_{x|y}^\rho) = \frac{\partial \log(\sum_y (\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})^{1+\rho})}{\partial \rho} \tag{132}$$

We have:

$$\begin{aligned}
\frac{\partial D(\bar{p}_{xy}^\rho \| p_{xy})}{\partial \rho} &= H(\bar{p}_{x|y}^\rho) + \rho \frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho} - \frac{\partial \log(\sum_y (\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})^{1+\rho})}{\partial \rho} \\
&= H(\bar{p}_{x|y}^\rho) + \rho \frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho} - H(\bar{p}_{x|y}^\rho) \\
&= \rho \frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho}
\end{aligned} \tag{133}$$

■

Lemma 12 $\text{sign} \frac{\partial [D(\bar{p}_{xy}^\rho \| p_{xy}) - H(\bar{p}_{x|y}^\rho)]}{\partial \rho} = \text{sign}(\rho - 1)$.

Proof: Using the previous lemma, we get:

$$\frac{\partial D(\bar{p}_{xy}^\rho \| p_{xy}) - H(\bar{p}_{x|y}^\rho)}{\partial \rho} = (\rho - 1) \frac{\partial H(\bar{p}_{x|y}^\rho)}{\partial \rho}$$

Then by Lemma 10, we get the conclusion. ■

Lemma 13

$$\rho H(p_{xy}^\rho) - (1 + \rho) \log \left(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right) = D(p_{xy}^\rho \| p_{xy}) \quad (134)$$

Proof: By noticing that $\log(p_{xy}(x, y)) = (1 + \rho)[\log(p_{xy}^\rho(x, y)) + \log(\sum_{s,t} p_{xy}(s, t)^{\frac{1}{1+\rho}})]$. We have:

$$\begin{aligned} D(p_{xy}^\rho \| p_{xy}) &= -H(p_{xy}^\rho) - \sum_{x,y} p_{xy}^\rho(x, y) \log(p_{xy}(x, y)) \\ &= -H(p_{xy}^\rho) - \sum_{x,y} p_{xy}^\rho(x, y) (1 + \rho) [\log(p_{xy}^\rho(x, y)) + \log(\sum_{s,t} p_{xy}(s, t)^{\frac{1}{1+\rho}})] \\ &= -H(p_{xy}^\rho) + (1 + \rho) H(p_{xy}^\rho) - (1 + \rho) \sum_{x,y} p_{xy}^\rho(x, y) \log(\sum_{s,t} p_{xy}(s, t)^{\frac{1}{1+\rho}}) \\ &= \rho H(p_{xy}^\rho) - (1 + \rho) \log \left(\sum_{s,t} p_{xy}(s, t)^{\frac{1}{1+\rho}} \right) \end{aligned} \quad (135)$$

Lemma 14

$$\rho H(\bar{p}_{x|y}^\rho) - \log \left(\sum_y \left(\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right)^{1+\rho} \right) = D(\bar{p}_{x|y}^\rho \| p_{xy}) \quad (136)$$

Proof:

$$\begin{aligned} D(\bar{p}_{x|y}^\rho \| p_{xy}) &= \sum_y \sum_x \frac{A(y, \rho)}{B(\rho)} \frac{C(x, y, \rho)}{D(y, \rho)} \log \left(\frac{\frac{A(y, \rho)}{B(\rho)} \frac{C(x, y, \rho)}{D(y, \rho)}}{p_{xy}(x, y)} \right) \\ &= \sum_y \sum_x \frac{A(y, \rho)}{B(\rho)} \frac{C(x, y, \rho)}{D(y, \rho)} [\log \left(\frac{A(y, \rho)}{B(\rho)} \right) + \log \left(\frac{C(x, y, \rho)}{D(y, \rho)} \right) - \log(p_{xy}(x, y))] \\ &= -\log(B(\rho)) - H(\bar{p}_{x|y}^\rho) + \sum_y \sum_x \frac{A(y, \rho)}{B(\rho)} \frac{C(x, y, \rho)}{D(y, \rho)} [\log(D(y, \rho)^{1+\rho}) - \log(C(x, y, \rho)^{1+\rho})] \\ &= -\log(B(\rho)) - H(\bar{p}_{x|y}^\rho) + (1 + \rho) H(\bar{p}_{x|y}^\rho) \\ &= -\log \left(\sum_y \left(\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}} \right)^{1+\rho} \right) + \rho H(\bar{p}_{x|y}^\rho) \end{aligned}$$

Lemma 15

$$H(p_{xy}^\rho) = \frac{\partial(1 + \rho) \log(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})}{\partial \rho} \quad (137)$$

Proof:

$$\begin{aligned} & \frac{\partial(1 + \rho) \log(\sum_y \sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})}{\partial \rho} \\ = & \log\left(\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}}\right) - \sum_y \sum_x \frac{p_{xy}(x, y)^{\frac{1}{1+\rho}}}{\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}}} \log(p_{xy}(x, y)^{\frac{1}{1+\rho}}) \\ = & - \sum_y \sum_x \frac{p_{xy}(x, y)^{\frac{1}{1+\rho}}}{\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}}} \log\left(\frac{p_{xy}(x, y)^{\frac{1}{1+\rho}}}{\sum_t \sum_s p_{xy}(s, t)^{\frac{1}{1+\rho}}}\right) \\ = & H(p_{xy}^\rho) \end{aligned} \quad (138)$$

■

Lemma 16

$$H(\bar{p}_{x|y}^\rho) = \frac{\partial \log(\sum_y (\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})^{1+\rho})}{\partial \rho} \quad (139)$$

Proof: Notice that $B(\rho) = \sum_y (\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})^{1+\rho}$, and $\frac{\partial B(\rho)}{\partial \rho} = B(\rho)H(\bar{p}_{x|y}^\rho)$ as shown in Lemma 9. It is clear that:

$$\begin{aligned} \frac{\partial \log(\sum_y (\sum_x p_{xy}(x, y)^{\frac{1}{1+\rho}})^{1+\rho})}{\partial \rho} &= \frac{\partial \log(B(\rho))}{\partial \rho} \\ &= \frac{1}{B(\rho)} \frac{\partial B(\rho)}{\partial \rho} \\ &= H(\bar{p}_{x|y}^\rho) \end{aligned} \quad (140)$$

■

References

- [1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [2] C. Chang and A. Sahai. The error exponent with delay for lossless source coding. In *IEEE Inform. Theory Workshop*, pages 252–256, Punta del Este, Uruguay, March 2006.
- [3] C. Chang and A. Sahai. Upper bound on error exponents with delay for lossless source coding with side-information. In *Proc. Int. Symp. Inform. Theory*, pages 326–330, Seattle, WA, July 2006.

- [4] T. M. Cover. A proof of the data compression theorem of Slepian and Wolf for ergodic sources. *IEEE Trans. Inform. Theory*, 21:226–228, March 1975.
- [5] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley and Sons, 1991.
- [6] I. Csiszár and J. Körner. *Information Theory, Coding Theorems for Discrete Memoryless Systems*. Akadémiai Kiadó, 1981.
- [7] S. C. Draper. Universal incremental Slepian-Wolf coding. In *Proc. 42nd Allerton Conf. on Communication, Control and Computing*, pages 1332–1341, October 2004.
- [8] A. W. Eckford and W. Yu. Rateless Slepian-Wolf codes. In *39th Asilomar Conf. Signals, Systems, Comp.*, pages 1757–1761, October 2005.
- [9] G. Forney. Convolutional codes III. Sequential decoding. *Information and Control*, 25(3):267–297, 1974.
- [10] R. G. Gallager. Source coding with side information and universal coding. Technical Report LIDS-P-937, Mass. Instit. Tech., 1976.
- [11] V. N. Koshchev. On a problem of separate coding of two dependent sources. *Prob. Peredachi Informatsii*, 13(1):26–32, 1977.
- [12] P. Koulgi, E. Tuncel, S. Regunathan, and K. Rose. On zero-error coding of correlated sources. *IEEE Trans. Inform. Theory*, 49:2856–2873, November 2003.
- [13] A. Sahai. Why block-length and delay behave differently if feedback is present. *IEEE Trans. Inform. Theory*, 54(5):1860–1886, 2008.
- [14] A. Sahai and S. Mitter. Source coding and channel requirements for unstable processes. *Submitted to IEEE Trans. Inform. Theory*, 2006.
- [15] N. Shulman and M. Feder. Source broadcasting with an unknown amount of receiver side information. In *Proc. 2002 Inform. Theory Workshop, Bangalore, India*, pages 127–130, October 2002.
- [16] D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. *IEEE Trans. Inform. Theory*, 19:471–480, July 1973.
- [17] L. Weng, S. Pradhan, and A. Anastasopoulos. Error exponent regions for gaussian broadcast and multiple access channels. *IEEE Trans. Inform. Theory*, 54(7):2919–2942, July 2008.
- [18] E.-H. Yang and D.-K. He. Interactive encoding and decoding for one way learning: Near lossless recovery with side information at the decoder. *IEEE Trans. Inform. Theory*, 56(4):1808–1824, April 2010.