Low-Complexity Approaches to Distributed Data Dissemination

by

Todd Prentice Coleman

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2006

Author .....
Department of Electrical Engineering and Computer Science
November 28, 2005

Certified by .....
Muriel Médard
Associate Professor
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Low-Complexity Approaches to Distributed Data Dissemination

by

Todd Prentice Coleman

Submitted to the Department of Electrical Engineering and Computer Science
on November 28, 2005, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

In this thesis we consider practical ways of disseminating information from multiple senders to multiple receivers in an optimal or provably close-to-optimal fashion. The basis for our discussion of optimal transmission of information is mostly information theoretic - but the methods that we apply to do so in a low-complexity fashion draw from a number of different engineering disciplines. The three canonical multiple-input, multiple-output problems we focus our attention upon are:

- The Slepian-Wolf problem where multiple correlated sources must be distributedly compressed and recovered with a common receiver.
- The discrete memoryless multiple access problem where multiple senders communicate across a common channel to a single receiver.
- The deterministic broadcast channel problem where multiple messages are sent from a common sender to multiple receivers through a deterministic medium.

Chapter 1 serves as an introduction and provides models, definitions, and a discussion of barriers between theory and practice for the three canonical data dissemination problems we will discuss. Here we also discuss how these three problems are all in different senses 'dual' to each other, and use this as a motivating force to attack them with unifying themes.

Chapter 2 discusses the Slepian-Wolf problem of distributed near-lossless compression of correlated sources. Here we consider embedding any achievable rate in an M-source problem to a corner point in a $2M-1$-source problem. This allows us to employ practical iterative decoding techniques and achieve rates near the boundary with legitimate empirical performance. Both synthetic data and real correlated data from sensors at the International Space Station are used to successfully test our approach.

Chapter 3 generalizes the investigation of practical and provably good decoding algorithms for multiterminal systems to the case where the statistical distribution

of the memoryless system is unknown. It has been well-established in the theoretical literature that such 'universal' decoders exist and do not suffer a performance penalty, but their proposed structure is highly nonlinear and therefore believed to be complex. For this reason, most discussion of such decoders has been limited to the realm of ontology and proof of existence. By exploiting recently derived results in other engineering disciplines (i.e. expander graphs, linear programming relaxations, etc), we discuss a code construction and two decoding algorithms that have polynomial complexity and admit provably good performance (exponential error probability decay). Because there is no need for a priori statistical knowledge in decoding (which in many settings - for instance a sensor network - might be difficult to repeatedly acquire without significant cost), this approach has very attractive robustness, energy efficiency, and stand-alone practical implications.

Finally, Chapter 4 walks away from the multiple-sender, single-receiver setting and steps into the single-sender-multiple receiver setting. We focus our attention here on the deterministic broadcast channel, which is dual to the Slepian-Wolf and multiple access problems in a number of ways - including how the difficulty of practical implementation lies in the encoding rather than decoding. Here we illustrate how again a splitting approach can be applied, and how the same properties from the Slepian-Wolf and multiple access splitting settings remain. We also discuss practical coding strategies for some problems motivated by wireless, and show how by properly 'dualizing' provably good decoding strategies for some channel coding problems, we admit provably good encoding for this setting.

Thesis Supervisor: Muriel Médard
Title: Associate Professor

Acknowledgments

I would first and foremost like to thank my family for being so supportive and patient while I have been a graduate student. I especially appreciate everyone's thoughtfulness and words of encouragement in my decision towards taking this type of career path. I would next like to thank my advisor, Muriel Médard, for her guidance and mentorship. Muriel, you are truly a special person and I feel fortunate to have been advised by you. I am very grateful for the opportunity to learn from you - on matters of both research and life. My other committee members, John Tsitsiklis, Ralf Koetter, Michelle Effros, and Andrea Goldsmith have also been great to work with. Luckily, I've had the opportunity to meet with each of you individually and discuss a whole lot more than research - it has been great to know you personally, and hopefully our interactions can continue.

I would also like to take this opportunity to recognize my close friend from my undergraduate days at the University of Michigan, Marcus Ash. Although you did not directly aid in my Ph.D. development, ever since my freshman year you have served as an ideal role model, mentor, and the big brother I never had. The perspective I have developed from your experiences is as much of a reason that I stand her today as is anything I read in a textbook. I would also like to thank another Michigan buddy of mine who did not come to Boston: Chris King. Chris, all those times we spent studying together at Michigan - along with your passionate love of knowledge and learning - has undoubtedly shaped my perspective on what it means to conduct good research. My Michigan friends who did end up in Boston, including the Aguirre brothers (Aaron, Andy and Derek), have brought a plethora of fond memories. Aaron, I especially thank you for stepping up in times of need - despite your business - and giving me honest constructive criticism that has strengthened all my endeavors.

My time in LIDS and Boston has been a whole lot more relaxing due to the friendships I developed with people in the lab, including: Muriel Médard's research group, Danielle Hinton, David Brown, Jun Sun, and Paul Njoroge. Of course I can't forget to acknowledge my other close MIT friends that I spent a good amount of time

socializing with: Ayanna Samuels, Eric Wade, Lincoln Chandler, Corey Harwell, Sean Bradshaw, and Fritz Pierre.

Lastly, I would like to thank the National Science Foundation and the National National Reconnaissance Office for their financial support of my graduate studies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this thesis we consider practical ways of disseminating information from multiple senders to multiple receivers in an optimal or provably close-to-optimal fashion. The basis for which we discuss optimal transmission of information is mostly information theoretic - but the methods that we apply to do so in low-complexity fashions draw from a number of different engineering disciplines. The three canonical multiple-input, multiple-output problems we focus our attention upon are

- The Slepian-Wolf problem where multiple correlated sources must be distributively compressed and recovered with a common receiver.
- The discrete memoryless multiple access problem where multiple senders communicate across a common channel to a single receiver.
- The deterministic broadcast channel problem where multiple messages are sent from a common sender to multiple receivers through a deterministic medium.

Chapter 2 discusses the Slepian-Wolf problem of distributed near-lossless compression of correlated sources. As in the case of the multiple access channel, it has been known for decades that simple encoding strategies combined with an optimal decoder result in no performance loss. The optimal decoder, except in certain 'corner' cases, however is highly complex and has served as the major barrier between theory and practice. Motivated by an analogous technique in the multiple access literature, we

introduce practical new tools for communicating at *all* rates in the achievable region by means of a 'source-splitting' strategy that allows for parallelized encoding and pipelined decoding at the speed of a single-user decoder. Here we also discuss how using low-complexity iterative decoding techniques with this approach leads to a significant simplification in part of the decoding process, and illustrate the achievability of non-vertex rates near the theoretical boundary with empirical performance.

Chapter 3 continues the investigation of practical and provably good decoding algorithms for multiterminal systems - but when the statistical knowledge of the memoryless system is unknown. It has been well-established in the theoretical literature that such 'universal' decoders exist and do not suffer a performance penalty, but their structure is highly nonlinear and complex. For this reason, most discussion of such decoders has been limited to the realm of ontology and proof of existence. By exploiting recently derived results in other engineering disciplines, we construct polynomial-complexity algorithms and discuss constructing codes that together admit provably good performance. Because there is no need for a priori statistical knowledge in decoding (which in many settings - for instance a sensor network - might be difficult to repeatedly acquire without significant cost), this approach has very attractive robustness, energy efficiency, and stand-alone practical implications.

Chapter 4 walks away from the multiple-sender, single-receiver setting and steps into the single-sender-multiple receiver setting. We focus our attention here on the deterministic broadcast channel, which is dual to the Slepian-Wolf problem in a number of ways. As opposed to the Slepian-Wolf and multiple access problems, the deterministic broadcast problem manifests its practical difficulties in the encoding operation. Here we illustrate how again a splitting approach can be applied, and how the same properties from the Slepian-Wolf and multiple access splitting settings readily apply. We also discuss practical coding strategies for some problems motivated by wireless, and show how by properly 'dualizing' provably good decoding strategies for some channel coding problems, we admit provably good encoding for this setting.

1.1 Multiple Senders, One Receiver

Here we will discuss two canonical information theory problems where there are multiple senders and one receiver. The receiver must takes its observations and recover the information sent by all senders with arbitrarily small probability of error. In these settings, it is usually the case that constructing good low-complexity encoding mechanisms is far simpler than constructing good low-complexity decoders.

1.1.1 The Discrete Multiple Access Channel

In the discrete multiple access problem, senders $X^1, X^2, \ldots, X^M$ transmit messages to a common receiver Y under channel uncertainty, here modeled as a memoryless conditional probability distribution $P\left(Y|X^1, \ldots, X^M\right)$. The capacity region is the closure of a union of polyhedra [Ahl71, Lia72]:

$$\mathrm{cl}\left[\bigcup_{P(X^1)\cdots P(X^M)} \left\{\underline{R} \in \mathbb{R}_+^M \mid \sum_{i\in S} R_i < I\left(X(S); Y|X(S^c)\right) \ \forall\, S \subseteq \{1, 2, \ldots, M\}\right\}\right]$$

where $X(S) = \{X^j\}_{j\in S}$ and $\mathrm{cl}(\cdot)$ denotes closure.

Certain 'corner' points have intuitive interpretations. For instance, when M = 2, the rate pair $(I(X^1; Y|X^2), I(X^2; Y))$ may be achieved as follows. The receiver first treats X^1 as noise, and then reliably decodes X^2 for any $R_2 \leq I(X^2; Y)$. The receiver then uses its knowledge of X^2 in Y and reliably decodes X^1 for any $R^1 \leq I(X^1; Y|X^2)$. Similarly, by reversing the roles of X^1 and X^2, we see that the pair $(I(X^1; Y), I(X^2; Y|X^1))$ also lies in the capacity region. By using time-sharing, any convex combination of the corner points may be achieved as well.

It has also been found recently [GRUW01] that performing 'rate-splitting' (whereby one user splits its rate into virtual users who contain codebooks that appear to be noise to each other) may achieve any point in the M-user capacity region, using no more than two virtual users per physical user, with a maximum of $2M - 1$ virtual users. For each user, the virtual users split rate to code for a single-user point-to-point channel with rate that accounts for the other users' presence.

1.1.2 The Slepian-Wolf Problem

The Slepian-Wolf problem of distributed near-lossless compression of correlated sources (see (L) of Figure 1-1) has been understood theoretically for many years [SW73]. It has received a lot of attention recently due to its relevance as a sub-component of numerous distributed data dissemination systems. Practical techniques, however, have remained elusive for quite a long time. The challenges include: finding provably good codes, low-complexity decoding, and choosing source coding rates. Recently, proper application of channel coding developments to this setting has been successful at addressing some of these challenges. However, explicit practical solutions that apply to all instantiations of the problem have not yet been constructed. This thesis applies channel coding developments to broaden the class of problems with low complexity solutions. Indeed, any instance of the problem can be addressed practically with our approach.

The achievable rate region $\mathcal{R}\left[P\left(u^1,\ldots,u^M\right)\right]$ for M memoryless sources $(U^1,\ldots,U^M)$ with joint probability distribution $P\left(u^1,\ldots,u^M\right)$ is given by [SW73]:

$$\mathcal{R}\left[P\left(u^1,\ldots,u^M\right)\right]=\left\{\underline{R}\in\mathbb{R}_+^M \;\middle|\; \sum_{i\in S} R_i \geq H\left(U(S)|U(S^c)\right)\ \forall\ S\subseteq\{1,2,\ldots,M\}\right\} \quad (1.1)$$

where $U(S)=\{U^j\}_{j\in S}$. (See (R) of Figure 1-1.) In [Cov75], Cover simplified the proof by proposing a code design strategy whereby each encoder randomly places all possible source sequences into bins and gives the bin index to the decoder. Linear block codes can be used to perform binning practically and with no loss in either the achievable rate region or the error exponent [Csi82]. In code operation, the decoder receives a single bin index from each transmitter and then searches for a collection $\left(\hat{\underline{U}}^1,\ldots,\hat{\underline{U}}^M\right)$ of 'jointly typical' sequences [CT91, pp. 194-197] lying in the described bins. This can be done with high probability provided that the rates lie within the achievable region. At certain rate points, which we call 'vertices' or 'corner points', this joint search over all codebooks for 'jointly typical' sequences can be done successively. The corner points are the rate tuples $(R_1,\ldots,R_M)$ that are obtained by

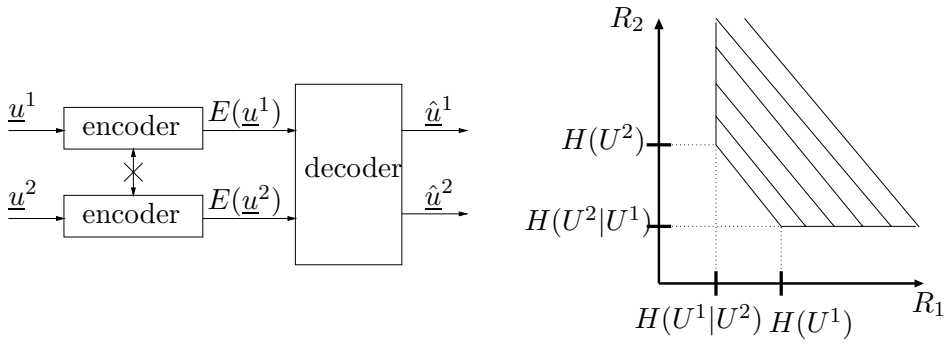


Figure 1-1: The Slepian-Wolf problem: (L) model (R) achievable rate region

expanding $H(U^1, \ldots, U^M)$ by M successive applications of the chain rule and assigning to each rate the unique corresponding term in the expansion. For instance, if users would like to communicate at the rate $(R_1, R_2) = (H(U^1), H(U^2|U^1))$, then we describe the source U^1 at rate $H(U^1)$ by entropy-encoding $\underline{U}^1$. (We can do this by using either a variable-rate lossless code or a fixed-rate near-lossless code.) After successful decoding, U^1 can be used as side information to help decode U^2 at rate $H(U^2|U^1)$. By exchanging the roles of U^1 and U^2, it follows that the same approach applies to encoding at rate $(R_1, R_2) = (H(U^1|U^2), H(U^2))$. Thus, in this case, the decoding process can be decomposed into a pipelined approach that operates at the speed of a single-user decoder. Recently, a lot of attention has been paid to the construction of low-complexity decoders to achieve rates of R_2 very close to $H(U^2|U^1)$. These attempts, which include iterative techniques for turbo-code [BGT93] constructions [GFZ01, AG02, BM01, LXG03a] and low-density parity check code (LDPC) [Gal62] constructions [TGFZ03, SPR02, LXG03b, GFZ03, LLN$^+$03], have met much success when U^1 and U^2 are binary random variables.

While these codes can be combined using time-sharing to achieve non-vertex rates, time-sharing has practical drawbacks. Rate fluctuations arise at different points of the encoding process, and the delay required to communicate near a target rate can be prohibitively long. Furthermore, as we will see in Section 2.3, significant error exponent reduction can ensue.

We consider in Chapter 2 a practical method to perform '*source-splitting*', which

transforms all points in the Slepian-Wolf achievable region into vertices in a Slepian-Wolf achievable region with more sources. Once the rate point becomes a vertex, we can parallelize encoding and pipeline decoding. Source-splitting was introduced in [RU97], but that approach required shared randomness at the encoders and decoder, and the outputs of the splitting operation had alphabets larger than the original source. Another approach that allows parallelized encoding and pipelined decoding is [Wil88], but this also requires common randomness at the encoder and decoder *and* involves searching for jointly typical sequences at the *encoder*. Our splitting technique involves a simple thresholding operation followed by specifying a bin index, reduces the alphabet size of the outputs of the splitter, and does not require common randomness.

We also illustrate via the 'method of types' [Csi98] and reasoning similar to [GRUW01] that performing the proposed splitting strategy at most once per user can achieve any rate in the Slepian-Wolf achievable rate region with parallelized encoding and pipelined decoding. We also discuss how the splitting strategy may be combined with iterative decoding in a practical setting. Our splitting technique has an important simplification in part of the decoding process. Simulation results confirm the practicality and effectiveness of this approach.

1.1.3 Slepian-Wolf, Multiple Access Duality

We would like to briefly mention that just as in the case of point-to-point communication, there is a precise duality between any instance of a Slepian-Wolf problem and any instance of a particular class of multiple-access problems. This is succinctly stated in [CT91, pp. 416-418] and precisely quantified in [Csi82]. In the point-to-point case, it is well known that any instance of a fixed-rate near-lossless data compression problem is precisely dual to any instance of a particular class of channel coding problems. The Slepian-Wolf, multiple access duality can be thought of as a multiterminal extension of this point-to-point case. The duality is as follows.

Point-to-point

Consider a discrete memoryless source U with probability distribution W. Without loss of generality, we assume that $\mathcal{U} = \{0, 1, \ldots, Q-1\}$ where $Q = 2^t$ for some integer $t \geq 1$. Thus we may assume that U takes on values in $\mathbb{F}_{2^t}$. We consider the case where a linear mapping

$$H = \begin{bmatrix} -H_1'- \\ -H_2'- \\ \vdots \\ -H_M'- \end{bmatrix} : \mathcal{U}^N \to \mathcal{U}^M$$

is used to map $\underline{u} \in \mathcal{U}^N$ to $\underline{s} \in \mathcal{U}^M$ via $\underline{s} = H\underline{u}$ where $M < N$ and U is memoryless with probability distribution $W \in \mathcal{P}(\mathcal{U})$. We will denote the rate R as

$$R = t\frac{M}{N} \tag{1.2}$$

and note that this corresponds to rate in a data compression sense and *not* in a channel coding sense (which would correspond to $t - R$). The decoder knows that $\underline{u}$ must be consistent with $\underline{s}$, in other words it must lie in the coset

$$\mathrm{Co}(H, \underline{s}) = \{\underline{u} \mid H\underline{u} = \underline{s}\}, \tag{1.3}$$

and selects $\hat{\underline{u}}$ as the 'best' coset member. This encompasses two settings:

a) Fixed-to-fixed length near-lossless data compression, where $\underline{u}$ is identified as the sourceword and $\underline{s}$ is the syndrome, the output of the compression operation.

b) An additive noise channel $\underline{Y} = \underline{X} \oplus \underline{U}$. By using a linear code $\mathcal{C}$ for $\underline{x}$, and identifying the parity check matrix H with $\mathcal{C}$ as

$$\mathcal{C} = \{\underline{x} : H\underline{x} = \underline{0}\}, \tag{1.4}$$

then we have that a sufficient statistic for decoding is

$$H\underline{y} = H\underline{u} = \underline{s}.$$

Successfully decoding for the noise vector $\underline{u}$ is equivalent to successfully decoding for the transmitted codeword $\underline{x}$:

$$\underline{\hat{x}} = \underline{\hat{u}} \oplus \underline{y}.$$

Note that for the same matrix H used in (a) and (b), decoding error probability is identical. For both problems (a) and (b), it is known [Csi82] that linear codes attain all achievable rates (as well as the random coding error exponent).

Multiterminal

Consider a universal pair of discrete memoryless sources (U^1, U^2) drawn according to a joint probability distribution W. For $k \in \{1, 2\}$ we define $Q^k = |\mathcal{U}_k| = 2^{t_k}$ and without loss of generality assume $\mathcal{U}^k = \{0, 1, \ldots, Q^k - 1\}$. For $k \in \{1, 2\}$ we consider the case where a linear mapping H^k:

$$H^k = \begin{bmatrix} -H_1^{k\prime}- \\ -H_2^{k\prime}- \\ \vdots \\ -H_{M_k}^{k\ \prime}- \end{bmatrix} : {\mathcal{U}_k}^N \rightarrow {\mathcal{U}_k}^{M_k}$$

is used over $\mathbb{F}_{2^{t_k}}$ to map $\underline{u} \in \mathcal{U}^N$ to $\underline{s} \in \mathcal{U}^{M_k}$ via $\underline{s}^k = H^k\underline{u}$ where $M_k < N$. We will denote the rates as

$$R_1 = t_1 \frac{M_1}{N} \tag{1.5}$$

$$R_2 = t_2 \frac{M_2}{N} \tag{1.6}$$

The decoder knows that $\left\{\underline{u}^k\right\}_{k=1}^2$ must be consistent with $\left\{\underline{s}^k\right\}_{k=1}^2$, in other words each $\underline{u}^k$ must lie in the coset

$$\mathrm{Co}\left(H^k, \underline{s}^k\right) = \{\underline{u}^k \mid H^k \underline{u}^k = \underline{s}^k\}, \tag{1.7}$$

and selects $\left\{\hat{\underline{u}}^k\right\}_{k=1}^2$ as the 'best' coset member. This encompasses two settings:

a) Fixed-to-fixed length near-lossless Slepian-Wolf data compression, where $\left\{\underline{u}^k\right\}_{k=1}^2$ are identified as the sourcewords and $\left\{\underline{s}^k\right\}_{k=1}^2$ as the syndromes, the outputs of the compression operation.

b) A multiple access channel where $\underline{x}^1 \in \mathcal{U}_1$ and $\underline{x}^2 \in \mathcal{U}_2$ are mapped to

$$\underline{y} = \left((y_1^1, y_1^2), (y_2^1, y_2^2) \ldots (y_N^1, y_N^2)\right) \in \{\mathcal{U}_1 \times \mathcal{U}_2\}^N$$

according to

$$\left(\underline{y}^k = \underline{x}^k \oplus \underline{u}^k\right)_{k=1,2}$$

By using linear codes $\mathcal{C}^k$ for $\underline{x}^k$, and identifying the parity check matrix H^k with $\mathcal{C}^k$ as

$$\mathcal{C}^k = \left\{\underline{x} : H^k \underline{x} = \underline{0}\right\}, \tag{1.8}$$

then we have that a sufficient statistic for decoding is the pair

$$\left(H^k \underline{y}^k = H^k \underline{u}^k = \underline{s}^k\right)_{k=1,2}$$

Successfully decoding for $\left\{\underline{u}^k\right\}_{k=1}^2$ is equivalent to successfully decoding for the transmitted codewords $\left\{\underline{x}^k\right\}$ $k = 1, 2$:

$$\hat{\underline{x}^k} = \hat{\underline{u}}^k \oplus \underline{y}^k.$$

Note that for the same matrices $\{H^1, H^2\}$ used in (a) and (b), decoding error probability is identical. It was also shown in [Csi82] that for both problems (a) and (b),linear codes attain all achievable rates (as well as the random coding error exponent).

1.1.4 Universal Decoding

The techniques previously mentioned all require knowledge of the joint probability distribution of sources $(U^1, \ldots, U^M)$. We next pursue universal decoding algorithms that deliver the same performance (in terms of achievable rates and rate of probability of error decay) without that knowledge. Csiszár's *minimum-entropy* decoder is one such universal decoder for a class of discrete memoryless systems [CK82, Csi82, Csi98], including the Slepian-Wolf problem as well as the type of multiple access problem discussed in Section 1.1.3. However, those pieces of work do not consider decoder complexity. Universal decoding is potentially very useful if it can be implemented with low complexity, since limited feedback and rate loss can make it difficult to estimate unknown statistics. To date, as far as we know, there has not been much effort at trying to build practical universal decoders.

In Chapter 3 we address the universal coding problem, taking into account complexity. We consider both point-to-point and multiterminal coding problems, given by the Slepian-Wolf problem as well as their dual multiple access problems of the type given in Section 1.1.3. Our perspective takes techniques from the channel coding literature and applies them to the universal setting. We construct two polynomial-complexity encoding and decoding algorithms that exhibit an exponential decay in error probability. One of them relies on a linear programming relaxation to decoding, while the other is an iterative bit-flipping approach. It is our hope that the methodologies presented here will form the basis for significant advances in the practical universal coding domain.

1.2 One Sender, Multiple Receivers

In Chapter 4 we will discuss a canonical information theory problem, whose capacity region is known, where there is one sender and multiple receivers. In this setting, the encoder must take multiple messages and combine them into a single channel input so that each receiver can decode its message with arbitrarily small probability of error. The decoders work independently and are far easier to design.

The deterministic broadcast channel has one sender and multiple (M) receivers. The sender combines the messages $\{m_j\}_{j=1}^{M}$ into a single length-n channel input $\underline{X} = \{X_1, \ldots, X_n\}$, where $m_j \in \{1, \ldots, 2^{nR_j}\}$ is the message for receiver j. Receiver j receives a deterministic function of X, i.e. $Y_i^j = f_j(X_i)$, and from this it attempts to reconstruct m_j, $\hat{m}_j = d_j(\underline{Y}^i)$. The capacity region of the deterministic broadcast channel is given by

$$\text{cl}\left[\bigcup_{P(X)}\left\{\underline{R} \in \mathbb{R}_+^M \mid \sum_{i \in S} R_i < H\left(Y(S)\right) \ \forall\ S \subseteq \{1, 2, \ldots, M\}\right\}\right],$$

where $Y(S) = \{Y^j\}_{j \in S}$.

We start off by motivating consideration of such channels by considering interference effects of simple wireless networks. We next show that any instance of a deterministic broadcast problem with M receivers may be reduced, via a *rate-splitting* transformation, to another $(2M-1)$-receiver problem where a successive encoding approach suffices. Analogous to rate-splitting for the multiple access channel and source-splitting for the Slepian-Wolf problem, *all* achievable rates (including non-vertices) apply. This amounts to significant complexity reduction at the encoder. Here we also discuss practical schemes for first-stage coding at vertices of any deterministic broadcast problem, using Cover's 'enumerative source coding' [Cov73] technique.

We then consider specific classes of deterministic broadcast channels and provide complete practical solutions. For all *degraded* deterministic broadcast channels, we show that the 'enumerative source coding' technique can be applied with zero error

for all stages of vertex coding. We also show that for classes of two-receiver channels where transmission to one user puts constraints on the alphabet available to transmission to the other, a strong duality exists with coding over erasure channels. We then illustrate how the capacity-achieving 'LT codes' erasure-correcting codes framework of [Lub02] can be 'dualized' to construct encoders with side information for our domain. We show that we can attain certain rates on the boundary of the capacity region using this practical approach. Such problems include our wireless interference management examples, as well the Blackwell channel - the simplest non-trivial deterministic broadcast channel.

Chapter 2

Practical Approaches to Slepian-Wolf Data Compression Problem

Background: Previous attempts to practically address the Slepian-Wolf problem have generally only been able to attain corner points of the region. Time-sharing between different corner points can attain all achievable rates, but this has its practical drawbacks. A source-splitting technique to attain all achievable was previously proposed, but this assumed common randomness at encoders and the decoder.

Our contribution: We propose a source-splitting technique for the Slepian-Wolf problem that requires no sources of common randomness. The technique is efficient - involving thresholding operations, and significantly reduces complexity in estimation when using iterative decoding. We use low-density parity check codes and iterative decoding to confirm the effectiveness of this approach - from simulations using both synthetic data and real data from the International Space Station. In this chapter we also compare source-splitting and time-sharing from a theoretical perspective. We demonstrate that source-splitting also has its benefits over time-sharing - from an error exponent perspective.

In this chapter we discuss a splitting transformation for the Slepian-Wolf problem that allows for *any* achievable rate to be encoded and decoded. Our approach uses parallelized encoding and pipelined decoding that operates at the speed of a singe-user decoder. Following the splitting transformation previously used for discrete multiple

access channel coding, this approach further manifests the duality discussed in [CT91, pp. 416-418], [Csi82]. The decoding methods in this chapter use a priori knowledge of the joint probability distribution between the correlated sources at the decoder. The practical effectiveness of this approach is illustrated with empirical performance using good channel codes and an iterative decoder.

2.1 Model and Definitions

In this paper, we will consider a set of M discrete memoryless sources $U^1, U^2, ..., U^M$ drawn according to $P\left(u^1, u^2, ..., u^M\right)$ with alphabets $\mathcal{U}^1, \mathcal{U}^2, ..., \mathcal{U}^M$. We denote U_j^i as the jth symbol from process U^i. We use the following notation:

$$\begin{aligned}
[r] &\triangleq \{1, 2, \ldots, r\} \\
R(\mathcal{S}) &\triangleq \sum_{i \in \mathcal{S}} R_i \text{ for any set } \mathcal{S} \text{ of integers, where } R_i \in \mathbb{R}_+ \\
\underline{U}^{\mathcal{S}} &\triangleq \left(U^i\right)_{i \in \mathcal{S}} \text{ for any set } \mathcal{S} \text{ of integers} \\
\underline{U}_{\mathcal{S}} &\triangleq \left(U_j\right)_{j \in \mathcal{S}} \text{ for any set } \mathcal{S} \text{ of integers} \\
\mathcal{S}^c &\triangleq [M] \backslash \mathcal{S} \\
\Pi(\mathcal{U}) &= \{\pi | \pi \text{ permutes } \mathcal{U}\} \\
H\left(U\right) &\triangleq \sum_{a \in \mathcal{U}} -P_U\left(a\right) \log_2\left(P_U\left(a\right)\right) \\
\mathcal{H}\left(U\right) &\triangleq \lim_{n \to \infty} \frac{1}{n} H\left(\underline{U}_{[n]}\right) \\
D\left(P \| Q\right) &\triangleq \sum_{a \in \mathcal{U}} P(a) \log_2 \left(\frac{P(a)}{Q(a)}\right)
\end{aligned}$$

2.1.1 Dominant Face

The *dominant face* $\mathcal{D}\left[\mathcal{R}\left[P(\underline{u}^{[M]})\right]\right]$ consists of all $R \in \mathcal{R}\left[P(\underline{u}^{[M]})\right]$ that satisfy

$$R([M]) = H(\underline{U}^{[M]}). \tag{2.1}$$

Note that any point in $\mathcal{R}$ is dominated (with respect to the standard partial order on $\mathbb{R}_+^M$) by a point in the dominant face.

Throughout the paper, we exploit the chain rule for entropy

$$H(\underline{U}^{\mathcal{T}}) = H(\underline{U}^{\mathcal{S}}) + H(\underline{U}_{\mathcal{T}\setminus\mathcal{S}}|\underline{U}^{\mathcal{S}}) \quad \forall \mathcal{S} \subseteq \mathcal{T} \subseteq [M]. \tag{2.2}$$

We may now apply the chain rule to derive an alternative description of the dominant face $\mathcal{D}$. By combining the chain rule with (1.1) and (2.1), we arrive at

$$R(\mathcal{S}) = R([M]) - R(\mathcal{S}^c) = H(\underline{U}^{[M]}) - R(\mathcal{S}^c) \leq H(\underline{U}^{[M]}) - H(\underline{U}^{\mathcal{S}^c}|\underline{U}^{\mathcal{S}}) = H(\underline{U}^{\mathcal{S}}).$$

So we see that achievability (1.1) and lying on the dominant face (2.1) imply that

$$H(\underline{U}^{\mathcal{S}}|\underline{U}^{\mathcal{S}^c}) \leq R(\mathcal{S}) \leq H(\underline{U}^{\mathcal{S}}) \quad \forall \mathcal{S} \subseteq [M]. \tag{2.3}$$

Conversely, we see that the leftmost inequality in (2.3) directly implies achievability (1.1) and setting $\mathcal{S} = [M]$ in (2.3) directly implies lying on the dominant face (2.1). Hence, we may alternatively characterize the dominant face as

$$\mathcal{D} = \left\{ R \in \mathbb{R}_+^M \;\middle|\; H(\underline{U}^{\mathcal{S}}|\underline{U}^{\mathcal{S}^c}) \leq R(\mathcal{S}) \leq H(\underline{U}^{\mathcal{S}}) \quad \forall \mathcal{S} \subseteq [M] \right\}. \tag{2.4}$$

Vertices are the rate tuples $\underline{R}_{[M]} \in \mathcal{D}$ that occur at the intersection of the bounding surfaces (for instance, they are the two 'corner points' of Figure 1-1). They are obtained by expanding $H(\underline{U}^{[M]})$ into M terms by $M-1$ successive applications of the chain rule, and assigning to R_i the value of the unique term in the expansion having the form $H(U^i|\underline{U}^{\mathcal{S}})$ for some set $\mathcal{S} \subseteq [M]$. Each unique vertex of the dominant face corresponds to a rate-tuple that is single-user decodable given side information of the previously decoded sources. Most of the practical methods [GFZ01, AG02, BM01, LXG03a, TGFZ03, SPR02, LXG03b, GFZ03, LLN+03] to achieve rates near the Slepian-Wolf achievable rate region boundary are only applicable to vertices.

2.2 Source-Splitting for Slepian-Wolf

Let us now consider taking each symbol of a DMS $U = (U_1, U_2 \ldots)$ where $U_i \in \mathcal{U} = \{0, \ldots, |\mathcal{U}| - 1\}$ and splitting it into a collection of random variables of smaller cardinality. We write $U_i \leftrightarrow (U_i^a, U_i^b)$ if there is a bijection between the random variables U_i and (U_i^a, U_i^b). We consider the following way to perform source-splitting:

$$U_i \;\mapsto\; \left(\begin{array}{l} U_i^a = \min(\pi(U_i), T) \\ U_i^b = \max(\pi(U_i), T) - T \end{array} \right) \mapsto U_i = \pi^{-1}\left(U_i^a + U_i^b\right), \tag{2.5}$$

where $T \in \mathcal{U}$ operates as a thresholder and $\pi \in \Pi(\mathcal{U})$ is a permutation operator.

Definition (2.5) gives many possible splits, since there are many possible values of $\pi \in \Pi(\mathcal{U})$ and $T \in \mathcal{U}$. For a nontrivial splitting threshold ($T \in \mathcal{U} \setminus \{0, |\mathcal{U}| - 1\}$), $U_i^a \in \{0, \ldots, T\}$,$U_i^b \in \{0, \ldots, |\mathcal{U}| - 1 - T\}$, and there are $\binom{|\mathcal{U}|}{T}$ distinct ways to map the $|\mathcal{U}|$ symbols to the splitting sets in (2.5). This provides a total of

$$\sum_{i=1}^{|\mathcal{U}|-2} \binom{|\mathcal{U}|}{i} = 2^{|\mathcal{U}|} - \binom{|\mathcal{U}|}{0} - \binom{|\mathcal{U}|}{|\mathcal{U}| - 1} - \binom{|\mathcal{U}|}{|\mathcal{U}|} = 2^{|\mathcal{U}|} - |\mathcal{U}| - 2 = O(2^{|\mathcal{U}|})$$

distinct ways to perform the splitting mechanism and form the bijection $U_i \leftrightarrow (U_i^a, U_i^b)$.

If we have two discrete memoryless sources (U^1, U^2) drawn according to $P(u^1, u^2)$, then we can split U^1 to form (U^{1a}, U^{1b}) as shown in (2.5). At this point, we have three sources, each of which can be encoded separately at rates R_{1a}, R_{1b}, R_2. We note that because $U \leftrightarrow (U^{1a}, U^{1b})$, $H\left(U^1, U^2\right) = H\left(U^{1a}, U^{1b}, U^2\right)$. Through the chain rule for entropy, we consider the rates

$$\begin{aligned} R_{1a} &= H\left(U^{1a}\right) && \text{(2.6a)} \\ R_2 &= H\left(U^2 | U^{1a}\right) && \text{(2.6b)} \\ R_{1b} &= H\left(U^{1b} | U^2, U^{1a}\right) && \text{(2.6c)} \\ R_1 &= R_{1a} + R_{1b}. && \text{(2.6d)} \end{aligned}$$

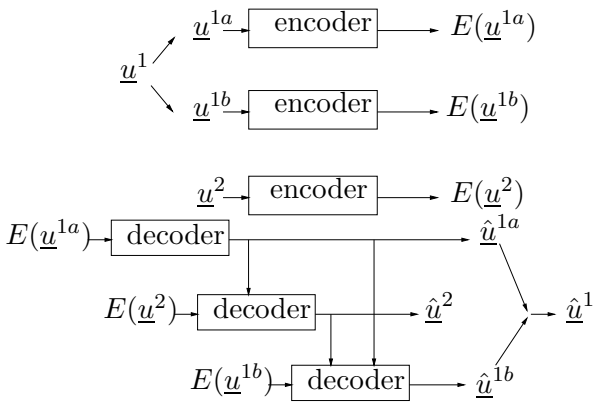


Figure 2-1: Source splitting and decoding for a two-source Slepian-Wolf problem

For any nontrivial split, (R_1, R_2) is not a vertex in $\mathcal{R}\left[P(u^1, u^2)\right]$, but (R_{1a}, R_2, R_{1b}) *is* a vertex in $\mathcal{R}\left[P\left(u^{1a}, u^2, u^{1b}\right)\right]$. This directly implies a parallelizable encoding strategy and pipelined single-user decoding strategy that operates with the complexity of a smaller-alphabet decoder. By varying across the different values of the threshold $T \in \mathcal{U}$ and $\pi \in \Pi(\mathcal{U})$, we may sweep across $O(2^{|\mathcal{U}|})$ distinct non-vertex points on the dominant face $\mathcal{D}\left[\mathcal{R}\left[P(u^1, u^2)\right]\right]$. Figure 2-1 illustrates the proposed encoding and decoding strategy.

Source-splitting may be performed to transform a source U of cardinality $|\mathcal{U}|$ into $|\mathcal{U}| - 1$ binary random variables:

$$U_i \;\mapsto\; \left(\begin{array}{l} U_i^1 = 1_{\{\pi(U_i)=1\}} \\ U_i^2 = 1_{\{\pi(U_i)=2\}} \\ \vdots \\ U_i^{|\mathcal{U}|-1} = 1_{\{\pi(U_i)=|\mathcal{U}|-1\}} \end{array}\right) \mapsto U_i = \pi^{-1}\left(\sum_{k=1}^{|\mathcal{U}|-1} k U_i^k\right) \tag{2.7}$$

where $\pi \in \Pi(\mathcal{U})$ and $1_{\{A\}} = 1$ if event A occurs and 0 otherwise. Each $\pi \in \Pi(\mathcal{U})$ yields new splits and thus there are $|\mathcal{U}|!$ splits.

The motivation for binary splitting is the reduction in complexity of near-lossless block-compression of high-rate sources: the splitting approach allows for parallelized encoding and pipelined single-user decoding of low-rate binary sources.

In the next section we show that although this method generates a *finite* number of distinct splits, we may group consecutive symbols together and interpret them as a single outcome of a source of larger alphabet. Because of the exponential growth in the number of splits as a function of the source alphabet size, it follows that long super-symbols lengths are not required. We also discuss in the next section a controlled way to map super-symbols to a desired rate point. Moreover we arrive at similar details about the required number of splits per source, as in case of multiple access [RU97].

2.2.1 Two Sources: At Most One Split Per Source Required

We consider a DMS U drawn according to pmf Q over alphabet $\mathcal{U} = \{0, 1, \ldots, |\mathcal{U}|-1\}$ and assume without loss of generality that $Q(a) > 0$ for each $a \in \mathcal{U}$. We treat the first n outcomes $\underline{U}_{[n]}$ of the source U as the single outcome of a DMS with alphabet $\{0, \ldots, |\mathcal{U}|^n - 1\}$ through the standard integral representation

$$\mathrm{sr}\left(\underline{u}_{[n]}\right) = \sum_{j=1}^{n} u_j |\mathcal{U}|^{j-1}. \tag{2.8}$$

Splitting $\mathrm{sr}\left(\underline{U}_{[n]}\right)$ according to (2.5) on $\mathrm{sr}\left(\underline{U}_{[n]}\right)$ yields $(\underline{U}^a_{[n]}, \underline{U}^b_{[n]})$ and a total of $2^{|\mathcal{U}|^n} - |\mathcal{U}|^n - 2 = O(2^{|\mathcal{U}|^n})$ non-trivial splits. We use the 'method of types' [Csi98] to take a subset of all $\pi \in \Pi(\mathrm{sr}\,(\mathcal{U}^n))$ and $T \in |\mathcal{U}|^n - 1$, parametrize them according to $\epsilon \in [0, 1]$, and demonstrate that $P_{\underline{U}^a_{[n]}(\epsilon)}(\cdot)$ tends to a continuous function of ϵ and $\frac{1}{n} H\left(\underline{U}^a_{[n]}(\epsilon)\right)$ tends to $\epsilon H(U)$. Moreover, we illustrate in Theorem 2.2.5 that any point on the dominant face of the two-user Slepian-Wolf achievable rate region can be transformed to a vertex in a three-user problem via source-splitting. Since the number of nontrivial splits grows as $O(2^{|\mathcal{U}|^n})$, operating near any target rate does not require long super-symbol lengths. We introduce some intermediate lemmas that are useful in the proof of Theorem 2.2.5.

We denote the set of all probability distributions on $\mathcal{U}$ by $\mathcal{P}(\mathcal{U})$. For a length-n sequence $\underline{u} = (u_1, u_2, \ldots, u_n) \in \mathcal{U}^n$, the type $P_{\underline{u}} \in \mathcal{P}(\mathcal{U})$ is the probability distri-

bution defined by $P_{\underline{u}}(a) = \frac{1}{n}\sum_{i=1}^{n} 1_{\{u_i=a\}}$, for all $a \in \mathcal{U}$. We denote by Q^n the pmf induced on $\mathcal{U}^n$ by n independent drawings according to Q. We denote by $\mathcal{P}_n(\mathcal{U}) = \{P^{0,n}, P^{1,n}, \ldots\}$ the subset of $\mathcal{P}(\mathcal{U})$ consisting of the possible types of sequences $\underline{u} \in \mathcal{U}^n$. For any type $P^{j,n} \in \mathcal{P}_n(\mathcal{U})$, the type class $T(P^{j,n})$ is the set of all $\underline{u} \in \mathcal{U}^n$ such that $P_{\underline{u}} = P^{j,n}$. From [Csi98] we note thAT:

$$\begin{aligned} |\mathcal{P}_n(\mathcal{U})| &= \binom{n+|\mathcal{U}|-1}{|\mathcal{U}|-1} \\ &\leq (n+1)^{|\mathcal{U}|} \qquad (2.9) \\ Q^n(\underline{u}) &= 2^{-n\left(H\left(P_{\underline{u}}\right)+D\left(P_{\underline{u}}\|Q\right)\right)} \ \forall\, \underline{u} \in \mathcal{U}^n. \qquad (2.10) \end{aligned}$$

Define:

$$\begin{aligned} \mathcal{J}(n) &= \{0,1,\ldots,|\mathcal{P}_n(\mathcal{U})|-1\} \qquad (2.11) \\ \mathcal{K}(j,n) &= \{0,1,\ldots,|T\left(P^{j,n}\right)|-1\},\ j \in \mathcal{J}(n) \qquad (2.12) \\ A(j,\epsilon,n) &= \left\lceil \epsilon \left|T\left(P^{j,n}\right)\right| \right\rceil,\ j \in \mathcal{J}(n). \qquad (2.13) \end{aligned}$$

We now construct the set of permutations $\Pi_{\epsilon,n}(\mathrm{sr}(\mathcal{U}^n)) \subset \Pi(\mathrm{sr}(\mathcal{U}^n))$. For each $j \in \mathcal{J}(n)$, order the members of $T(P^{j,n})$ lexicographically. Then any $\underline{u}_{[n]} \in \mathcal{U}^n$ can be uniquely specified by $\left(j(\underline{u}_{[n]}), k(\underline{u}_{[n]})\right)$ where $j(\underline{u}_{[n]}) \in \mathcal{J}(n)$ satisfies $\underline{u}_{[n]} \in T\left(P^{j(\underline{u}_{[n]}),n}\right)$ and $k(\underline{u}_{[n]}) \in \mathcal{K}(j(\underline{u}_{[n]}),n)$ denotes the lexicographically ordered position of $\underline{u}_{[n]}$ in $T\left(P^{j(\underline{u}_{[n]}),n}\right)$. Conversely, we define $\underline{u}_{[n]}^{j,k}$ to be the $(k+1)$st member of $T(P^{j,n})$.

We define the type class integral representation parametrized by ϵ as

$$\tau_\epsilon\left(\underline{u}_{[n]}\right) = \left(\sum_{i=0}^{j(\underline{u}_{[n]})-1} A(i,\epsilon,n)\right) + k(\underline{u}_{[n]}). \qquad (2.14)$$

We then construct a set $\Pi_{\epsilon,n}(\mathrm{sr}(\mathcal{U}^n))$ of permutations on $\mathrm{sr}(\mathcal{U}^n)$ so that any $\pi_{\epsilon,n} \in \Pi_{\epsilon,n}(\mathrm{sr}(\mathcal{U}^n))$ satisfies

$$\forall\, \underline{u}_{[n]}\ s.t.\ k(\underline{u}_{[n]}) < A\left(j(\underline{u}_{[n]}),\epsilon,n\right):\ \ \pi_{\epsilon,n}(\mathrm{sr}\left(\underline{u}_{[n]}\right)) = \tau_\epsilon\left(\underline{u}_{[n]}\right). \qquad (2.15)$$

Finally, we define the threshold

$$T_{\epsilon,n} \;=\; \sum_{j\in\mathcal{J}(n)} A\left(j,\epsilon,n\right) = \sum_{j\in\mathcal{J}(n)} \left\lceil \epsilon \left| T\left(P^{j,n}\right)\right|\right\rceil . \tag{2.16}$$

Intuitively, any $\pi_{\epsilon,n} \in \Pi_{\epsilon,n}(\mathrm{sr}\left(\mathcal{U}^n\right))$ maps approximately a fraction ϵ of the members of each type class $P^{j,n}$ to values below the threshold $T_{\epsilon,n}$, and the remaining ones to values at or above $T_{\epsilon,n}$. As n grows, this approximation becomes more exact. The set $\Pi_{\epsilon,n}(\mathrm{sr}\left(\mathcal{U}^n\right))$ contains more than one permutation since the definition given by (2.15) does not specify the order for strings $\underline{u}_{[n]}$ that satisfy $k(\underline{u}_{[n]}) \geq A\left(j(\underline{u}_{[n]}),\epsilon,n\right)$.

We now split $\underline{U}_{[n]}$ into $\underline{U}^a_{[n]}(\epsilon)$ and $\underline{U}^b_{[n]}(\epsilon)$

$$\underline{U}_{[n]} \mapsto \left(\begin{array}{l} \underline{U}^a_{[n]}(\epsilon) = \min(\pi_{\epsilon,n}(\mathrm{sr}\left(\underline{U}_{[n]}\right)), T_{\epsilon,n}) \\ \underline{U}^b_{[n]}(\epsilon) = \max(\pi_{\epsilon,n}(\mathrm{sr}\left(\underline{U}_{[n]}\right)), T_{\epsilon,n}) - T_{\epsilon,n} \end{array} \right) \tag{2.17}$$

where $\pi_{\epsilon,n} \in \Pi_{\epsilon,n}(\mathrm{sr}\left(\mathcal{U}^n\right))$ and $T_{\epsilon,n}$ is given by (2.16). Note that $\underline{U}^a_{[n]}(\epsilon)$ has cardinality $T_{\epsilon,n}+1$ and all $\pi_{\epsilon,n} \in \Pi_{\epsilon,n}(\mathrm{sr}\left(\mathcal{U}^n\right))$ lead to the same random variable $\underline{U}^a_{[n]}(\epsilon)$.

We next demonstrate the asymptotic continuity of the distribution of $\underline{U}^a_{[n]}(\epsilon)$ with respect to ϵ. The given property is not obvious because for $0 \leq \epsilon' < \epsilon \leq 1$ and large enough n, $T_{\epsilon,n} > T_{\epsilon',n}$. Moreover, for the same value of $r < T_{\epsilon',n} < T_{\epsilon,n}$, the event $\{\underline{U}^a_{[n]}(\epsilon) = r\}$ does not necessarily correspond in any sense to the event $\{\underline{U}^a_{[n]}(\epsilon') = r\}$. Nonetheless, Lemma 2.2.1, proved in Appendix A.2, shows that asymptotic Lipschitz continuity of $P_{\underline{U}^a_{[n]}(\epsilon)}(\cdot)$ essentially holds. Lemma 2.2.2, proved in Appendix A.3, shows the corresponding property for the joint distribution $P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\cdot,\cdot)$.

Lemma 2.2.1. *For any $\epsilon,\epsilon' \in [0,1]$, $\underline{U}^a_{[n]}(\epsilon)$ forms a bijection with another random variable $\underline{\tilde{U}}^a_{[n]}(\epsilon)$ that satisfies*

$$\lim_{n\to\infty} \left| P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\cdot) - P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(\cdot) \right|_1 = 2|\epsilon-\epsilon'|.$$

Lemma 2.2.2. *Let $(\underline{U}^a_{[n]}(\epsilon), \underline{U}^b_{[n]}(\epsilon))$ be a split of the discrete memoryless source U, and let $\underline{U}^{\mathcal{S}}$ be another set of discrete memoryless sources. Then for any $\epsilon,\epsilon' \in [0,1]$,*

$\underline{U}^a_{[n]}(\epsilon)$ *forms a bijection with another random variable* $\tilde{\underline{U}}^a_{[n]}(\epsilon)$ *that satisfies*

$$\lim_{n\to\infty} \left| P_{\underline{U}^{\mathcal{S}}_{[n]},\tilde{\underline{U}}^a_{[n]}(\epsilon)}(\cdot,\cdot) - P_{\underline{U}^{\mathcal{S}}_{[n]},\tilde{\underline{U}}^a_{[n]}(\epsilon')}(\cdot,\cdot) \right|_1 = 2|\epsilon - \epsilon'|.$$

Lemma 2.2.3, proved in Appendix A.4, demonstrates the relationship between the entropy rate $\mathcal{H}(U^a(\epsilon))$ and $H(U)$. Lemma 2.2.4, proved in Appendix A.5, shows the corresponding continuity for the conditional entropy.

Lemma 2.2.3. *For* $\epsilon \in [0,1]$*, the random variable* $\underline{U}^a_{[n]}(\epsilon)$ *defined in (2.17) satisfies* $\mathcal{H}(U^a(\epsilon)) = \epsilon H(U)$*.*

Lemma 2.2.4 (Range Lemma). *Let* $(\underline{U}^a_{[n]}(\epsilon), \underline{U}^b_{[n]}(\epsilon))$ *be a split of the discrete memoryless source* U*. Then* $g(\epsilon) = \mathcal{H}\left(\underline{U}^{\mathcal{S}}|U^a(\epsilon)\right)$ *defines a continuous function from* $[0,1]$ *onto the interval* $[H(\underline{U}^{\mathcal{S}}|U), H(\underline{U}^{\mathcal{S}})]$*.*

Together, these results prove that any point on the dominant face of the achievable rate region can be approximated to arbitrary accuracy using the given approach, as shown in Theorem 2.2.5.

Theorem 2.2.5. *For two sources* U^1, U^2 *with joint distribution* $P(u^1,u^2)$*, any point on the dominant face* $\mathcal{D}$ *of* $\mathcal{R}[P(u^1,u^2)]$ *can be transformed via source-splitting* U^1 *according to (2.17) to a vertex in* $\mathcal{R}\left[P\left(u^{1a},u^2,u^{1b}\right)\right]$*.*

Proof. Using the chain rule for entropy and the fact that $U^1 \leftrightarrow (U^{1a}(\epsilon), U^{1b}(\epsilon))$, we have that

$$\begin{aligned} R_1 + R_2 &= H\left(U^1,U^2\right) = \mathcal{H}\left(U^1,U^2\right) \\ &= \mathcal{H}\left(U^{1a}(\epsilon), U^{1b}(\epsilon), U^2\right) \\ &= \mathcal{H}\left(U^{1a}(\epsilon)\right) + \mathcal{H}\left(U^2|U^{1a}(\epsilon)\right) + \mathcal{H}\left(U^{1b}(\epsilon)|U^{1a}(\epsilon),U^2\right). \end{aligned}$$

By the Range Lemma we can set ϵ so that $R_2 = \mathcal{H}(U^2|U^{1a}(\epsilon))$. We may then define $R_a = \mathcal{H}(U^{1a}(\epsilon))$ and $R_b = \mathcal{H}\left(U^{1b}(\epsilon)|U^{1a}(\epsilon),U^2\right)$ where $R_a + R_b = R_1$. Then we note from the Slepian-Wolf theorem that the rate-tuple (R_a, R_2, R_b) is achievable, and furthermore, it is a vertex of the region $\mathcal{R}\left[P\left(u^{1a},u^2,u^{1b}\right)\right]$. □

2.2.2 M Sources: At Most One Split Per Source Required

We now apply the source-splitting procedure for the Slepian-Wolf problem with $M > 2$ users and show that $2M-1$ virtual sources are sufficient. The argument is based upon a recursive generalization of Theorem 2.2.5. The technique employed to show this is analogous to Section II of [GRUW01]. From there it follows from direct manipulation of the arguments in Section III of [GRUW01] that at most one split per source is required.

Theorem 2.2.6. *Consider M correlated sources $\underline{U}^{[M]}$ with product distribution $P_{\underline{U}^{[M]}}\left(\underline{u}^{[M]}\right)$, and let $\mathcal{R}\left[P_{\underline{U}^{[M]}}\left(\underline{u}^{[M]}\right)\right]$ and $\mathcal{D}$ be the corresponding Slepian-Wolf achievable rate region and dominant face. Any $R_{[M]} \in \mathcal{D}$ may be transformed to a vertex in a $2M-1$ source Slepian-Wolf achievable rate region by splitting each source at most once using (2.17).*

Proof. Suppose $R_{[M]} \in \mathcal{D}$. Apply the split (2.17) to source U^M to arrive at $\left(U^a(\epsilon), U^b(\epsilon)\right)$. For each $\mathcal{S} \subseteq [M-1]$ the inequality

$$R(\mathcal{S}) \leq \mathcal{H}\left(\underline{U}^{\mathcal{S}} | U^a(\epsilon)\right) \tag{2.18}$$

is valid for all sufficiently small $\epsilon \in [0,1]$ by the following argument. For $\epsilon = 0$ it is valid, since

$$R(\mathcal{S}) \leq H\left(\underline{U}^{\mathcal{S}}\right) = H\left(\underline{U}^{\mathcal{S}} | U^a(0)\right).$$

Since $\mathcal{H}\left(\underline{U}^{\mathcal{S}} | U^a(\epsilon)\right)$ is continuous in ϵ, there exists a largest interval $J_{\mathcal{S}} = [0, \epsilon_{\mathcal{S}}] \subset [0,1]$ such that (2.18) is fulfilled for all $\epsilon \in J_{\mathcal{S}}$.

Hence, for any $\mathcal{S} \subseteq [M-1]$ we have from (2.18) that

$$R(\mathcal{S}) \leq \mathcal{H}\left(\underline{U}^{\mathcal{S}} | U^a(\epsilon_{\mathcal{S}})\right) \tag{2.19}$$

and from the definition of $\epsilon_{\mathcal{S}}$ it follows that

$$R(\mathcal{S}) = \mathcal{H}\left(\underline{U}^{\mathcal{S}} | U^a(\epsilon_{\mathcal{S}})\right). \tag{2.20}$$

Choose

$$\epsilon' = \min_{\mathcal{S}\subseteq[M-1]} \epsilon_{\mathcal{S}} \tag{2.21}$$

and let $\mathcal{T} \subseteq [M-1]$ be the largest subset of $[M-1]$ that satisfies $\epsilon_{\mathcal{T}} = \epsilon'$. From (2.20) with $\mathcal{S} = \mathcal{T}$ we have

$$R(\mathcal{T}) = \mathcal{H}\left(\underline{U}^{\mathcal{T}} | U^a(\epsilon')\right). \tag{2.22}$$

Define a virtual $(M+1)$-source $\left(U^{1\prime}, \ldots, U^{M+1\prime}\right) = \left(U^1, \ldots, U^{M-1}, U^b(\epsilon'), U^a(\epsilon')\right)$. Let $(R'_1, \ldots, R'_{M+1})$ be the $(M+1)$-tuple defined by $R'_i = R_i, i \in [M-1]$ and

$$\begin{aligned} R'_{M+1} &= \mathcal{H}\left(U^a(\epsilon')\right) \\ R'_M &= R_M - R'_{M+1}. \end{aligned} \tag{2.23}$$

We next show that $(R'_1, R'_2, \ldots, R'_{M+1}) \in \mathcal{D}'$ where $\mathcal{D}'$ is the dominant face of the Slepian-Wolf achievable rate region corresponding to the $M+1$ sources. We first illustrate that (2.1) holds and then show achievability (1.1).

Note that by the definition of $(R'_1, R'_2, \ldots, R'_{M+1})$ and since the splits form a bijection we have that

$$R'([M+1]) = R([M]) = H\left(\underline{U}^{[M]}\right) = \mathcal{H}\left(\underline{U}^{[M+1]'}\right). \tag{2.24}$$

It remains to be shown that the rate tuple $(R'_1, R'_2, \ldots, R'_{M+1})$ is achievable, i.e.

$$R'(\mathcal{S}) \geq \mathcal{H}\left(\underline{U'}^{\mathcal{S}} | \underline{U'}^{\mathcal{S}^c}\right), \ \forall\, \mathcal{S} \subseteq [M+1]. \tag{2.25}$$

We note from (2.24) and the chain rule for entropy that $R'(\mathcal{S}^c) \leq \mathcal{H}\left(\underline{U'}^{\mathcal{S}^c}\right)$ would imply

$$R'(\mathcal{S}) \geq \mathcal{H}\left(\underline{U'}^{[M+1]}\right) - \mathcal{H}\left(\underline{U'}^{\mathcal{S}^c}\right) = \mathcal{H}\left(\underline{U'}^{\mathcal{S}} | \underline{U'}^{\mathcal{S}^c}\right). \tag{2.26}$$

Therefore it suffices to show that $R'(\mathcal{S}^c) \leq \mathcal{H}\left(\underline{U'}^{\mathcal{S}^c}\right)$ for each $\mathcal{S}^c \subseteq [M+1]$. We

enumerate the cases:

- $\{M, M+1\} \subset \mathcal{S}$ or $\{M, M+1\} \subset \mathcal{S}^c$: this holds by (2.4).

- $M+1 \in \mathcal{S}^c$ and $M \in \mathcal{S}$:

$$\begin{aligned} R'(\mathcal{S}^c) &= R'_{M+1} + R'(\mathcal{S}^c \backslash \{M+1\}) \\ &= R'_{M+1} + R(\mathcal{S}^c \backslash \{M+1\}) \\ &\leq \mathcal{H}\left(U^a(\epsilon')\right) + \mathcal{H}\left(\underline{U}^{\mathcal{S}^c \backslash \{M+1\}} | U^a(\epsilon')\right) \qquad (2.27) \\ &= \mathcal{H}\left(U^a(\epsilon'), \underline{U}^{\mathcal{S}^c \backslash \{M+1\}}\right) \\ &= \mathcal{H}\left(\underline{U'}^{\mathcal{S}^c}\right) \end{aligned}$$

  where (2.27) holds by (2.23), (2.21), and (2.19).

- $M \in \mathcal{S}^c$ and $M+1 \in \mathcal{S}$:

$$\begin{aligned} R'(\mathcal{S}^c) &= R(\mathcal{S}^c \setminus \{M\}) + \left(R_M - R'_{M+1}\right) \\ &= R(\mathcal{S}^c) - R'_{M+1} \\ &\leq H\left(\underline{U}^{\mathcal{S}^c}\right) - \mathcal{H}\left(U^a(\epsilon')\right) \qquad (2.28) \\ &= H\left(\underline{U}^{\mathcal{S}^c \backslash \{M\}}, U^M\right) - \mathcal{H}\left(U^a(\epsilon')\right) \\ &= \mathcal{H}\left(\underline{U}^{\mathcal{S}^c \backslash \{M\}}, U^a(\epsilon'), U^b(\epsilon')\right) - \mathcal{H}\left(U^a(\epsilon')\right) \\ &= \mathcal{H}\left(\underline{U}^{\mathcal{S}^c \backslash \{M\}}, U^b(\epsilon') | U^a(\epsilon')\right) \qquad (2.29) \\ &\leq \mathcal{H}\left(\underline{U}^{\mathcal{S}^c \backslash \{M\}}, U^b(\epsilon')\right) \\ &= \mathcal{H}\left(\underline{U'}^{\mathcal{S}^c}\right) \end{aligned}$$

  where (2.28) holds by (2.4) and (2.23).

Thus we have that $(R'_1, R'_2, \ldots, R'_{M+1}) \in \mathcal{D}'$. Note further that by our choice of ϵ' there exists a $\mathcal{T} \subseteq [M-1]$ such that

$$R'(\mathcal{T}) = R(\mathcal{T}) = \mathcal{H}\left(\underline{U}^{\mathcal{T}} | U^a(\epsilon')\right) = \mathcal{H}\left(\underline{U'}^{\mathcal{T}} | U^a(\epsilon')\right). \qquad (2.30)$$

It follows that, in addition to (2.23), we also have that for any $\mathcal{S} \subseteq \mathcal{T}$,

$$\begin{aligned} R'(\mathcal{S}) &= R'(\mathcal{T}) + R'(\{M+1\}) - R'\left((\mathcal{T}\backslash\mathcal{S}) \cup \{M+1\}\right) \\ &\geq \mathcal{H}\left(\underline{U'}^{\mathcal{T}} | U^a(\epsilon')\right) + \mathcal{H}\left(U^a(\epsilon')\right) - \mathcal{H}\left(\underline{U'}^{\mathcal{T}\backslash\mathcal{S}}, U^a(\epsilon')\right) \\ &= \mathcal{H}\left(\underline{U'}^{\mathcal{T}}, U^a(\epsilon')\right) - \mathcal{H}\left(\underline{U'}^{\mathcal{T}\backslash\mathcal{S}}, U^a(\epsilon')\right) \\ &= \mathcal{H}\left(\underline{U'}^{\mathcal{S}} | \underline{U'}^{\mathcal{T}\backslash\mathcal{S}}, U^a(\epsilon')\right) \quad \forall \mathcal{S} \subseteq \mathcal{T}. \end{aligned} \tag{2.31}$$

Finally, for all $\mathcal{S} \subseteq [M] \setminus \mathcal{T}$,

$$R'(\mathcal{S}) \geq \mathcal{H}\left(\underline{U'}^{\mathcal{S}} | \underline{U}^{\mathcal{S}^c\prime}\right) \tag{2.32}$$

by (2.25).

This suggests the following parallelizable way of decoding $(R'_1, R'_2, \ldots, R'_{M+1})$. First note that from (2.23), we can entropy encode and decode $U^a(\epsilon')$ at rate $\mathcal{H}\left(U^a(\epsilon')\right)$. Knowledge of $U^a(\epsilon')$ can be kept at the decoder and we see that the group $\underline{U'}^{\mathcal{T}}$ can be encoded and decoded according to (2.30). This follows from (2.31) and the Slepian-Wolf coding theorem. Finally, it follows from (2.32) that with knowledge of $U^a(\epsilon')$ and $\underline{U'}^{\mathcal{T}}$ at the decoder, we may decode the remaining group of users. Each of these three groups has size at most $M-1$. From the $M=2$ case, we know that every rate point on the dominant face can be achieved by rate-splitting with at most $2M-1=3$ virtual sources. Let us assume by induction that for the $M-1$ user case, every rate tuple may be achieved with rate-splitting using at most $2(M-1)-1$ virtual sources. We just saw that for the M-user case, we can decompose it into a single-source encoding problem, and two Slepian-Wolf encoding problems of size m and $M-m$, respectively, where $1 \leq m < M$. By applying the induction hypothesis on these two smaller Slepian-Wolf encoding problems, we see that any rate-tuple in the M-user region can be achieved by rate-splitting with at most

$$1 + (2m-1) + (2(M-m)-1) = 2M-1$$

virtual sources.

Finally we observe that each user needs to split at most once to achieve any rate point on the dominant face. Algebraic topology techniques used to prove the analogous result in the discrete multiple access setting ([GRUW01], sec. III) directly apply in this setting. □

2.2.3 M Sources: The Boundary of the Dominant Face

Now we show that rate tuples on the *boundary* of the dominant face can be divided into two sets of sources that may be decoded successively *but otherwise independently.*

We can express the dominant face $\mathcal{D}\left[\mathcal{R}\left[P(\underline{u}^{[M]})\right]\right]$ in three ways:

$$\mathcal{D} = \mathcal{D}_1 = \left\{\underline{R} \in \mathbb{R}_+^M \;\middle|\; H(\underline{U}^{\mathcal{S}}|\underline{U}^{\mathcal{S}^c}) \leq R(\mathcal{S})\ \forall \mathcal{S} \subseteq [M],\ \text{with equality for } \mathcal{S} = [M]\right\} \tag{2.33}$$

$$= \mathcal{D}_2 = \left\{\underline{R} \in \mathbb{R}_+^M \;\middle|\; H(\underline{U}^{\mathcal{S}}|\underline{U}^{\mathcal{S}^c}) \leq R(\mathcal{S}) \leq H(\underline{U}^{\mathcal{S}})\ \forall \mathcal{S} \subseteq [M]\right\} \tag{2.34}$$

$$= \mathcal{D}_3 = \left\{\underline{R} \in \mathbb{R}_+^M \;\middle|\; R(\mathcal{S}) \leq H(\underline{U}^{\mathcal{S}})\ \forall \mathcal{S} \subseteq [M],\ \text{with equality for } \mathcal{S} = [M]\right\} \tag{2.35}$$

where (2.33) is a restatement of (1.1),(2.1); (2.34) is a restatement of (2.4); and (2.35) follows because $\mathcal{D}_3 \supseteq \mathcal{D}_2$ holds directly and $\mathcal{D}_3 \subseteq \mathcal{D}_1$ holds by exchanging $\mathcal{S}$ in $\mathcal{D}_3$ with $\mathcal{S}^c$ in $\mathcal{D}_1$ and applying the chain rule for entropy.

We say a rate tuple $R \in \mathcal{D}$ lies on the *boundary* of $\mathcal{D}$ if there exists a proper subset $\mathcal{A} \subset [M]$ such that

$$R(\mathcal{A}) = H\left(\underline{U}^{\mathcal{A}}\right). \tag{2.36}$$

Rates that are on the boundary of $\mathcal{D}$ have the desirable property that they allow serial, but otherwise independent, decoding of sets of sources and their complements. More specifically, if R is on the boundary of $\mathcal{D}$ and $\mathcal{A}$ satisfies (2.36), then we can jointly decode the subset of inputs with index in $\mathcal{A}$ and subsequently jointly decode the subset of inputs with index in $\mathcal{A}^c = [M]\backslash\mathcal{A}$. The proof is as follows.

By definition, for a point on the boundary there is at least one $\mathcal{A} \subset [M]$ such that

(2.36) holds. Now note that for any $\mathcal{L} \subset \mathcal{A}$,

$$\begin{aligned} R(\mathcal{L}) &= H\left(\underline{U}^{\mathcal{A}}\right) - R(\mathcal{A}\backslash\mathcal{L}) \\ &\geq H\left(\underline{U}^{\mathcal{A}}\right) - H\left(\underline{U}^{\mathcal{A}\backslash\mathcal{L}}\right) \qquad (2.37) \\ &= H\left(\underline{U}^{\mathcal{L}}|\underline{U}^{\mathcal{A}\backslash\mathcal{L}}\right) \qquad (2.38) \end{aligned}$$

where (2.37) follows from (2.35). From (2.33) and (2.38), (2.36) we now have

$$R_{\mathcal{A}} \in \mathcal{D}\left[\mathcal{R}\left[P\left(\underline{u}^{\mathcal{A}}\right)\right]\right] \qquad (2.39)$$

where $R_{\mathcal{A}} = (R_i)_{i\in\mathcal{A}}$. Thus $\underline{U}^{\mathcal{A}}$ can be decoded *independently* of $\underline{U}^{\mathcal{A}^c}$. Finally, since $R \in \mathcal{D}\left[\mathcal{R}\left[P(\underline{u}^{[M]})\right]\right]$, (2.33) allows for $\underline{U}^{\mathcal{A}^c}$ to be decoded successfully by using a successive decoder with $\underline{U}^{\mathcal{A}}$ as side information.

2.3 Time-Sharing versus Source-Splitting: an Error Exponent Analysis

In this setting we discuss alternative approaches to attain achievable rates on the dominant face of the Slepian-Wolf region. Here we will focus on the two-user setting, but this can naturally be generalized. Consider two sources (U^1, U^2) with joint probability distribution $W(u^1, u^2)$.

Generally speaking, the decoder must find a pair of 'jointly typical' sequences [CT91, pp. 194-197] consistent with what is observed. This is in general a computationally difficult task. At *vertex* rate points, the joint search over both codebooks for a pair of 'jointly typical' sequences can be done successively. For instance, if users would like to communicate at the rate of $(R_1, R_2) = (H(U^1), H(U^2|U^1))$, then we note that communicating at a rate of $H(U^1)$ can be done by simply entropy-encoding either a variable-rate lossless fashion or a near-lossless fixed-rate fashion. After successful decoding, U^1 can be passed as side information to help decode U^2 at a rate of $H(U^2|U^1)$. By exchanging the roles of U^1 and U^2, it follows that the same approach

applies to encoding at the vertex rate $(R_1, R_2) = (H(U^1|U^2), H(U^2))$. Recently, a lot of attention has been paid to the construction of low-complexity decoders to achieve rates of R_2 very close to $H(U^2|U^1)$.

A more interesting question concerns communicating at *any* rate in the achievable rate region - not necessarily vertices. The most efficient communication schemes minimize sum rate and thus attain rates lying on the *dominant face*, $\mathcal{D}\{\mathcal{R}[W]\}$, given by

$$(R_1, R_2) \in \mathcal{R}[W] \ : \ R_1 + R_2 = H\left(U^1, U^2\right).$$

Two candidate approaches of using decoding strategies that rely upon vertex decoding are:

- time-sharing, where coding for a non-vertex point is done by coding a certain fraction $\alpha \in [0, 1]$ of the time at one vertex, and the remaining fraction $1 - \alpha$ of the time at the other vertex

- source-splitting (see Section 2.2), where coding for a non-vertex point in a two-source problem is done by splitting one of the sources and coding at a vertex rate in the corresponding three-source problem

We would like to understand here the performance of the two candidate approaches at rates near the joint entropy boundary, in terms of error probability. We illustrate below that the source-splitting approach is more robust for decoding at arbitrary rates on the dominant face as compared to time-sharing, which can have significant error exponent penalty at rates close to vertices. As a by-product of our analysis, we show an interesting connection between information theory and estimation theory: the error exponent of vertex decoding in an arbitrary instance of the Slepian-Wolf problem depends on the inverse of the Fisher information of Gallager's ρ-parametrized tilted distribution.

2.4 Error Exponents

Here we discuss the near-lossless fixed-to-fixed distributed data compression setting where n samples of the memoryless source $\{(U_i^1, U_i^2)\}_{i=1}^n$ are separately encoded. For each source j, the $\left\{U_i^j\right\}_{i=1}^n$ symbols will be mapped to 2^{nR_j} output symbols. The error exponent for a particular coding scheme k will be denoted by

$$E^k(R_1, R_2) \triangleq \liminf_{n\to\infty} -\frac{1}{n} \log P_e^k(R_1, R_2).$$

As illustrated in the appendix (A.45), for random variables X, Y with joint distribution W, the error exponent $E_{x|y}(R)$ for source coding X at rate R with side information Y has a flat slope at $R = H(X|Y)$:

$$\frac{d}{dR}\left\{E_{x|y}(R)\right\}_{R=H(X|Y)} = 0.$$

Thus to capture the behavior of the exponent at $R = H(X|Y) + \delta$, we must consider second order effects:

$$\begin{aligned} E_{x|y}(H(X|Y) + \delta) &= \frac{1}{2}\delta^2 \frac{d^2}{dR^2}\left\{E_{x|y}(R)\right\}_{R=H(X|Y)} + o(\delta^2) \\ &= \frac{1}{2}\delta^2 E''_{x|y}(H(X|Y)) + o(\delta^2). \end{aligned}$$

We will denote the error exponent for time-sharing as $E^t(R_1, R_2)$ and that for source-splitting as $E^s(R_1, R_2)$. We are interested in the behavior of the error exponent at rates near the dominant face.

2.4.1 Time-Sharing

Time-sharing is one approach to attain any rate on the dominant face. For $\alpha \in [0, 1]$, αn of the samples are encoded near the vertex

$$(R_1, R_2) = \left(H\left(U^1\right), H\left(U^2|U^1\right)\right)$$

and the remaining $(1-\alpha)n$ samples are encoded near the other vertex

$$(R_1, R_2) = \left(H\left(U^1|U^2\right), H\left(U^2\right)\right).$$

We will assume that decoding is done with the pipelined vertex decoding approach described above. Thus for the decoding of the αn symbol pairs at the rate $(H(U^1)+\delta, H(U^2|U^1)+\delta)$, we have

$$\begin{aligned} P_e^{t,\alpha} &\le P\left(\left[\hat{\underline{U}}^1\right]_1^{\alpha n} \neq \left[\underline{U}^1\right]_1^{\alpha n}\right) + P\left(\left[\hat{\underline{U}}^2\right]_1^{\alpha n} \neq \left[\underline{U}^2\right]_1^{\alpha n} \,\middle|\, \left[\underline{U}^1\right]_1^{\alpha n}\right) \\ &= 2^{-n\alpha\left[E_{u^1}\left(H\left(U^1\right)+\delta\right)-o(n)\right]} + 2^{-n\alpha\left[E_{u^2|u^1}\left(H\left(U^2|U^1\right)+\delta\right)-o(n)\right]} \end{aligned}$$

For the decoding of the $(1-\alpha n)$ symbol pairs at the rate $(H(U^1|U^2)+\delta, H(U^2)+\delta)$, we have

$$\begin{aligned} P_e^{t,1-\alpha} &\le P\left(\left[\hat{\underline{U}}^2\right]_{\alpha n+1}^{n} \neq \left[\underline{U}^2\right]_{\alpha n+1}^{n}\right) + P\left(\left[\hat{\underline{U}}^1\right]_{\alpha n+1}^{n} \neq \left[\underline{U}^1\right]_{\alpha n+1}^{n} \,\middle|\, \left[\underline{U}^2\right]_{\alpha n+1}^{n}\right) \\ &= 2^{-n(1-\alpha)\left[E_{u^2}\left(H\left(U^2\right)+\delta\right)-o(n)\right]} + 2^{-n(1-\alpha)\left[E_{u^1|u^2}\left(H\left(U^1|U^2\right)+\delta\right)-o(n)\right]} \end{aligned}$$

Thus it follows that for $(R_1, R_2) \in \mathcal{D}$,

$$\begin{aligned} E^t(R_1+\delta, R_2+\delta) &= \min\Big[\alpha E_{u^1}\left(H\left(U^1\right)+\delta\right), \alpha E_{u^2|u^1}\left(H\left(U^2|U^1\right)+\delta\right), \\ &\qquad (1-\alpha)E_{u^2}\left(H\left(U^2\right)+\delta\right), (1-\alpha)E_{u^1|u^2}\left(H\left(U^1|U^2\right)+\delta\right)\Big] \\ &= \frac{1}{2}\delta^2 \min\Big[\alpha E''_{u^1}\left(H\left(U^1\right)\right), \alpha E''_{u^2|u^1}\left(H\left(U^2|U^1\right)\right), \\ &\qquad (1-\alpha)E''_{u^2}\left(H\left(U^2\right)\right), (1-\alpha)E''_{u^1|u^2}\left(H\left(U^1|U^2\right)\right)\Big] + o(\delta^2) \end{aligned}$$

where α satisfies

$$R_1 = \alpha H\left(U^1\right) + (1-\alpha)H\left(U^1|U^2\right). \tag{2.40}$$

2.4.2 Source-Splitting

As discussed in Section 2.2, *source-splitting* transforms a point on the dominant face of the two-source problem to a vertex point in a three-source problem:

$$U_i^1 \;\mapsto\; \left(\begin{array}{l} U_i^{1a} = f_a\left(U_i^1\right) \\ U_i^{1b} = f_b\left(U_i^1\right) \end{array}\right) \mapsto U_i^1 = f\left(U_i^{1a}, U_i^{1b}\right) \tag{2.41}$$

where $f_a : \mathcal{U}_1 \to \mathcal{U}_1$, $f_b : \mathcal{U}_1 \to \mathcal{U}_1$ and $f : \mathcal{U}_1 \to \mathcal{U}_1$ satisfy $f\left(f_a(u), f_b(u)\right) = u$ for each $u \in \mathcal{U}_1$. As an example of how to construct $\{f_a, f_b, f\}$, see (2.5). This corresponds to a vertex in the U^{1a}, U^{1b}, U^2 problem by using the chain rule for entropy and encoding at rates given by (2.6). Also in this scheme, coding to attain a non-vertex is mapped to coding at a vertex. Here we also assume that decoding is done with the pipelined vertex decoding approach. Thus for the decoding, we have

$$\begin{aligned} P_e^s &\le P\left(\hat{\underline{U}}^{1a} \neq \underline{U}^{1a}\right) + P\left(\hat{\underline{U}}^{2} \neq \underline{U}^{2} | \underline{U}^{1a}\right) + P\left(\hat{\underline{U}}^{1b} \neq \underline{U}^{1b} | \underline{U}^{1a}, \underline{U}^{2}\right) \\ &= 2^{-n\left[E_{u^{1a}}\left(H\left(U^{1a}\right)+\frac{1}{2}\delta\right)-o(n)\right]} + 2^{-n\left[E_{u^2|u^{1a}}\left(H\left(U^2|U^{1a}\right)+\delta\right)-o(n)\right]} \\ &+ 2^{-n\left[E_{u^{1b}|u^2,u^{1a}}\left(H\left(U^{1b}|U^2,U^{1a}\right)+\frac{1}{2}\delta\right)-o(n)\right]} \end{aligned}$$

Thus it follows that for $(R_1, R_2) \in \mathcal{D}$,

$$\begin{aligned} E^s(R_1+\delta, R_2+\delta) = \min \Bigg[& E_{u^{1a}}\left(H\left(U^{1a}\right)+\frac{1}{2}\delta\right), E_{u^2|u^{1a}}\left(H\left(U^2|U^{1a}\right)+\delta\right), \\ & E_{u^{1b}|u^2,u^{1a}}\left(H\left(U^{1b}|U^2,U^{1a}\right)+\frac{1}{2}\delta\right)\Bigg] \\ = \frac{1}{2}\delta^2 \min \Bigg[& \frac{1}{4}E''_{u^{1a}}\left(H\left(U^{1a}\right)\right), E''_{u^2|u^{1a}}\left(H\left(U^2|U^{1a}\right)\right), \\ & \frac{1}{4}E''_{u^{1b}|u^2,u^{1a}}\left(H\left(U^{1b}|U^2,U^{1a}\right)\right)\Bigg] + o(\delta^2). \end{aligned}$$

Note that to attain a rate of $(R_1+\delta, R_2+\delta)$ we have to allocate $\frac{1}{2}\delta$ extra rate to U^{1a} and $\frac{1}{2}\delta$ to U^{1b}, as compared to the usual δ to U^1 in the time-sharing case.

2.4.3 Comparison

It is the purpose of this discussion to observe how the error exponents behave for the two approaches when coding at rates $(R_1 + \delta, R_2 + \delta)$ where $(R_1, R_2) \in \mathcal{D}$. From the onset it is not clear which approach has better exponents - both cases exhibit error exponent degradation. In the case of time-sharing, error exponent degradation is caused by a reduction in the effective block length by factors of α and $1 - \alpha$. In the source-splitting scenario, error exponent degradation arises because of error propagation in decoding three sources rather than two, along with the reduction by a factor of $\frac{1}{4}$ due to the splitting operation. Furthermore, the comparison is not straightforward because the source-splitting operation creates a new joint distribution on the three sources, as compared to the original joint distribution on the two sources. Although these distributions are in some sense equivalent because (U_i^1, U_i^2) and $(U_i^{1a}, U_i^2, U_i^{1b})$ form a bijection, the behavior of the error exponent's second derivative involves more complicated functions of the distribution than just entropy:

Lemma 2.4.1.

$$E''_{x|y}\left(H(X|Y)\right) = \frac{1}{-H(X|Y)^2 + \sum_{x,y} Q(x,y)\log^2[Q(x|y)]}.$$

Proof details are in the appendix.

What is interesting about this denominator is that it can be characterized in terms of the Fisher information of Gallager's ρ-tilted distribution [Gal76]. In particular, if we define $Q_\rho(x, y)$ as the product of $Q_\rho(y)$, given by (A.39), and $Q_\rho(x|y)$, given by (A.40), then we can calculate the Fisher information of this parametrized probability distribution:

$$F(\rho) = \sum_{x,y} \left(\frac{d \log Q_\rho(x,y)}{d\rho}\right)^2 Q_\rho(x,y) \tag{2.42}$$

Then we can characterize the error exponent's second derivative in terms of the inverse of the above Fisher information quantity:

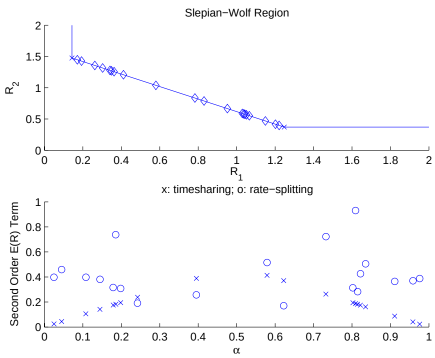
Lemma 2.4.2.

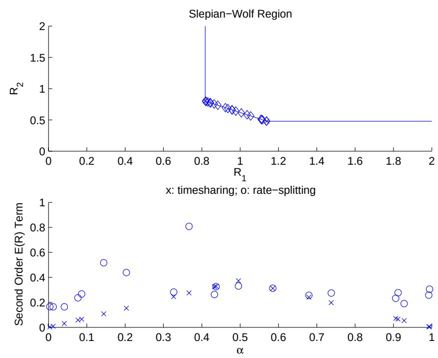
$$E''_{x|y}\left(H(X|Y)\right) = \frac{1}{F(\rho)}\Big|_{\rho=0}.$$

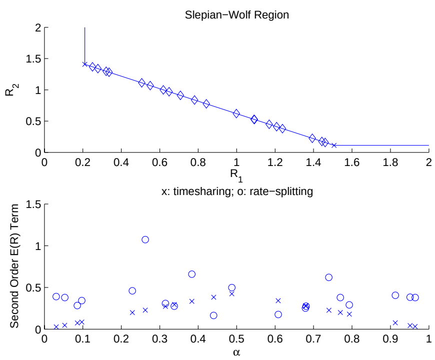
Proof details are in the appendix. This leads to another interesting connection between information-theoretic quantities (error exponents) and estimation theoretic ones (MMSE and Fisher Information). In particular, the connection relates the error exponent's second derivative to the inverse of a Fisher information - which is a bound on minimum mean-squared error. The common thread appears to lie in the information geometry [AN00] interpretation of the Kullback-Leibler distance. However in our opinion, an in-depth understanding of this relation remains to be found.

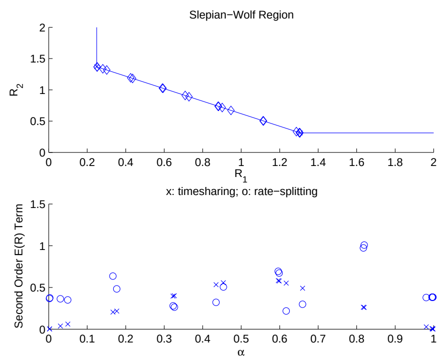
2.5 Examples on Randomly Constructed Joint Probability Distributions

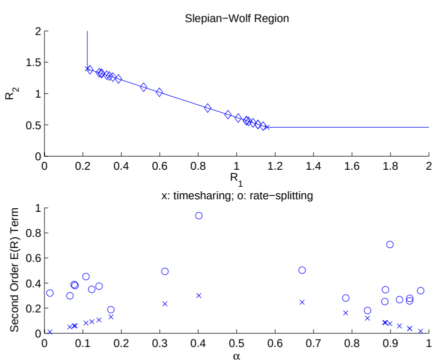
Here we randomly construct joint probability distributions W on (U^1, U^2) and compare $E^s(R_1+\delta, R_2+\delta)$ with $E^t(R_1+\delta, R_2+\delta)$. In the figure pairs below, the top figure in each pair shows the Slepian-Wolf achievable rate region and the target rate points on the dominant face. The bottom figure in each pair shows $E^s(R_1+\delta, R_2+\delta)$ and $E^t(R_1+\delta, R_2+\delta)$ as a function of α, where α satisfies (2.40). For the splitting case, splitting is done according to (2.5). The takeaway theme from all these examples is that the minimum $E^s(R_1+\delta, R_2+\delta)$ for points $(R_1, R_2) \in \mathcal{D}$ is bounded away from 0 whereas $E^t(R_1+\delta, R_2+\delta)$ decays to 0 linearly as α approaches 0 or 1. Consequently at rates close to vertices, the second order source-splitting exponent significantly dominates that of time-sharing. At rates halfway between vertices, in some cases source-splitting wins, and in other cases time-sharing does. We were not able to find many cases where the second-order time-sharing exponent significantly dominates that of source-splitting. Thus in terms of error exponents, source-splitting appears to be more robust across various rates than time-sharing.

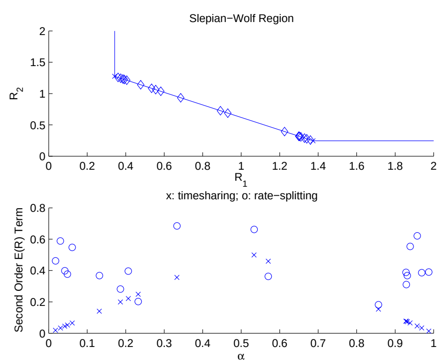

Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α

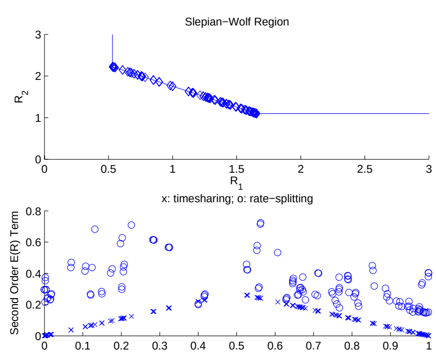

Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α

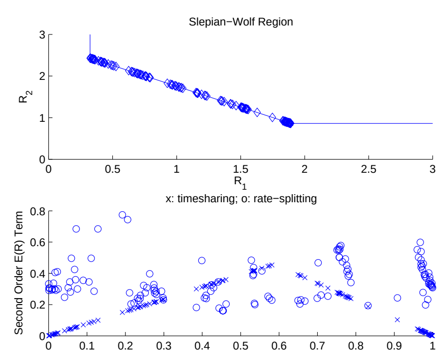

Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α

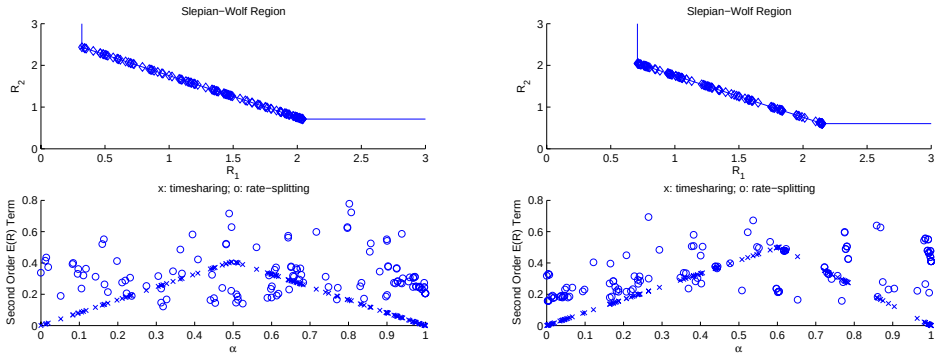

Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α


Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α


Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α


Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α


Slepian−Wolf Region
R2
R1
x: timesharing; o: rate−splitting
Second Order E(R) Term
α



2.6 Source-Splitting and Iterative Decoding for Slepian-Wolf

We discuss in this section how we can combine iterative decoding methods with source-splitting and point out how the splitting strategies defined in (2.5) and (2.7) significantly facilitate part of the decoding process. We conclude by showing simulation results.

Using the successive decoding approach of Section 2.2 we can near-losslessly compress a pair of sources (U^1, U^2) drawn according to $P(u^1, u^2)$ at any rate (R_1, R_2) on the dominant face $\mathcal{D}$ of $\mathcal{R}[P(u^1, u^2)]$. The strategy performs the splitting operation (2.5) and allocates rates according to (2.6a)-(2.6d).

Good binning strategies exist to perform successing decoding at rates that are vertices of the Slepian-Wolf region. Iterative decoding using 'syndrome-former' LDPC encoders [TGFZ03, SPR02, LXG03b, GFZ03, LLN$^+$03] and punctured turbo code encoders [GFZ01, AG02, BM01, LXG03a] have been extremely successful.

The iterative decoding technique applied here is the sum-product algorithm [KFL01], which operates on the graphical structure of the code. For example, Figure 2-2 illustrates a normal graph representation [For01] of an LDPC used as a syndrome-former encoder, where the syndrome $\underline{s}$ is the index of the bin in which input $\underline{u}$ lies. The sum-product algorithm produces symbol-wise a posteriori probabilities (APPs), which are approximate on graphs with cycles. We use carefully constructed graphical representations that allow for the approximate APPs to give credible empirical performance.

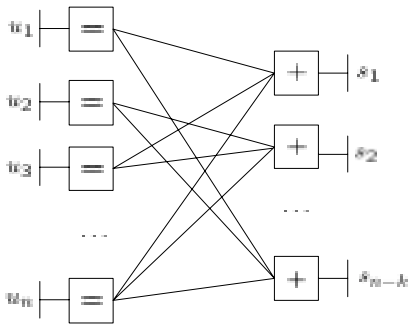


Figure 2-2: Normal syndrome-former encoding graph

In the context of our problem, the bin indices handed to the decoder for (U^{1a}, U^{1b}, U^2) are denoted as $(\underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2)$. At each level of the pipeline, the APP outputs of previously decoded users are used as inputs to the currently operating decoder. The outputs of the iterative decoders are the approximate APPs

$$\begin{aligned} P\left(U_i^{1a} = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) &\triangleq \operatorname{app}_i^{1a}(u), \\ P\left(U_i^{1b} = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) &\triangleq \operatorname{app}_i^{1b}(u), \\ P\left(U_i^{2} = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) &\triangleq \operatorname{app}_i^{2}(u). \end{aligned}$$

Let the outputs of the decoder be the estimate $(\underline{\hat{U}}^1, \underline{\hat{U}}^2)$, which may be constructed from the APPs of (U^1, U^2) by performing the symbol-based Maximum A Posteriori (MAP) decoding:

$$\hat{U}_i^j = \arg \max_{u \in \{0,1,\ldots|\mathcal{U}|-1\}} \operatorname{app}_i^j(u).$$

While $\operatorname{app}_i^2(u)$ is the direct output of one of the iterative decoders, $(\operatorname{app}_i^{1a}(u), \operatorname{app}_i^{1b}(u))$ must be combined to yield $\operatorname{app}_i^1(u)$. The splitting strategy (2.5) leads to the implication

$$j \neq T : U_i^{1a} = j \;\Rightarrow\; U^{1b} = 0 \tag{2.43}$$

$$j \neq 0 : U_i^{1b} = j \;\Rightarrow\; U^{1a} = T \tag{2.44}$$

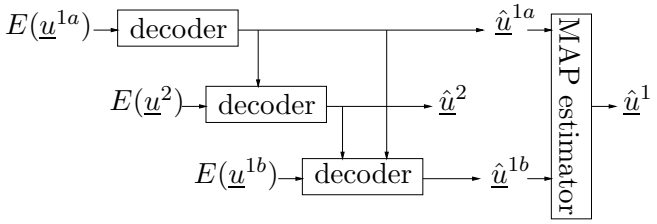


Figure 2-3: Combining iterative decoding with source-splitting

and thus $\mathrm{app}_i^1(u)$ can be constructed with very low complexity:

$$\begin{aligned}
u < T : P\left(U_i^1 = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) &= P\left(U_i^{1a} = u, U_i^2 = 0 | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= P\left(U_i^{1b} = 0 | U_i^1 = u, \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) P\left(U_i^{1a} = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= P\left(U_i^{1a} = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= \mathrm{app}_i^1(u) \quad \text{by (2.43)} \\
u > T : P\left(U_i^1 = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) &= P\left(U_i^{1a} = T, U_i^{1b} = u - T | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= P\left(U_i^{1a} = T | U_i^{1b} = u - T, \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) P\left(U_i^{1b} = u - T | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= P\left(U_i^{1b} = u - T | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= \mathrm{app}_i^{1b}(u - T) \quad \text{by (2.44)} \\
P\left(U_i^1 = T | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) &= 1 - \sum_{u \neq T} P\left(U_i^1 = u | \underline{s}^{1a}, \underline{s}^{1b}, \underline{s}^2\right) \\
&= 1 - \left(\sum_{u=0}^{T-1} \mathrm{app}_i^{1a}(u)\right) - \left(\sum_{u=T+1}^{|\mathcal{U}|-1} \mathrm{app}_i^{1b}(u - T)\right).
\end{aligned}$$

Figure 2-3 gives a schematic of the decoding process. In the case of binary splitting (2.7), the decoder observes bin indices $\underline{s}^1, \underline{s}^2, \ldots, \underline{s}^{|\mathcal{U}|-1}$ and the iterative successive decoder outputs will be the APPs

$$\begin{aligned}
P\left(U_i^1 = u | \underline{s}^1, \underline{s}^2, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) &\triangleq \mathrm{app}_i^1(u), \\
&\vdots \quad \vdots \\
P\left(U_i^{|\mathcal{U}|-1} = u | \underline{s}^1, \underline{s}^2, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) &\triangleq \mathrm{app}_i^{|\mathcal{U}|-1}(u).
\end{aligned}$$

In this case the implication

$$k \in \{1, \ldots, |\mathcal{U}| - 1\} \text{ and } U_i^k = 1 \Rightarrow U_i^r = 0, \ \forall r \neq k \tag{2.45}$$

holds and we can construct $\text{app}_i(u)$ again with very low complexity:

$$\begin{aligned}
k \neq 0 : P\left(U_i = k | \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) &= P\left(U_i^k = 1, U_i^r = 0, r \neq k | \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) \\
&= P\left(U_i^r = 0, r \neq k | U_i^k = 1, \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) \\
&\quad \cdot P\left(U_i^k = 1 | \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) \\
&= P\left(U_i^k = 1 | \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) \\
&= \text{app}_i^k(1) \text{ owing to (2.45)} \\
P\left(U_i = 0 | \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) &= 1 - \sum_{k=1}^{|\mathcal{U}|-1} P\left(U_i = k | \underline{s}^1, \ldots, \underline{s}^{|\mathcal{U}|-1}\right) \\
&= 1 - \sum_{k=1}^{|\mathcal{U}|-1} \text{app}_i^k(1).
\end{aligned}$$

2.6.1 Simulation Results

Synthetic Data

We now discuss simulation results that illustrate the promise of this splitting technique. The experiments begin with the random selection of a joint probability distribution for sources over $\mathcal{U}^1 = \mathcal{U}^2 = \mathcal{U} = GF(2^m)$ for some m. We then draw n independent samples and encode using an irregular LDPC with degree distribution drawn according to the density evolution results provided in [AU]. Once the non-zero components of the parity matrix are constructed, their values are selected randomly from $\{1, \ldots, 2^m - 1\}$. We perform the sum-product update rule in its dual form ([For01], section IX), which operates on the Fourier Transform of APPs. Also we note that in the case of $GF(2^m)$, the transormed APPs lie in $\mathbb{R}$ rather than $\mathcal{C}$. Thus the same gain in decoding complexity reduction is attained here as is in the binary case.

Figure 2-4 illustrates the achievability of non-vertices in the two source Slepian-

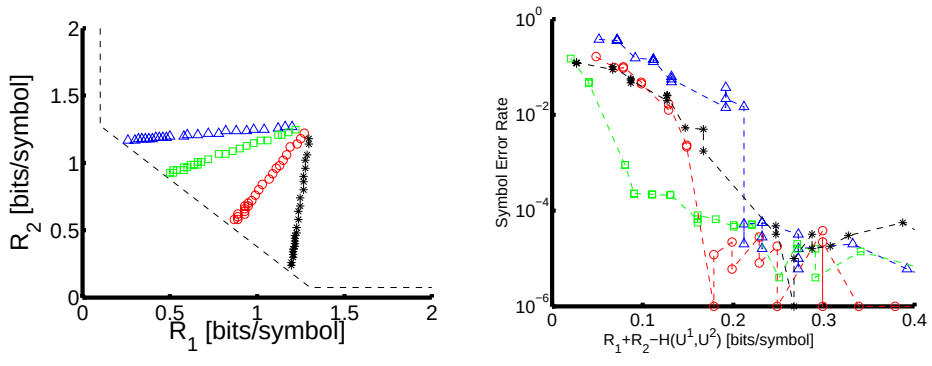


Figure 2-4: Symbol error rate for source-splitting to achieve non-vertex rate pairs.

Wolf problem using splitting and iterative decoding for $m = 2$ and $n = 5000$. The leftmost plot shows four non-vertex rate pairs on the boundary of the achievable region. We perform iterative decoding in their neighborhoods for a collection of points. The rightmost plot shows the symbol error rate as a function of the difference between the sum rate and the joint entropy. The given results show error probabilities of 10^{-4} at sum rate penalties between 0.1 and 0.25.

Quantized Data from International Space Station Sensors

Here we present the methods and results of applying the splitting methodology to actual data from the International Space Station. The Beta Gimbal Assembly (BGA) of the two solar array wings on the International Space Station (ISS) [FGH02] is responsible for rotating the solar panels in order to keep them directed at the sun. There is one BGA for each wing, referred to as BGA 4B and BGA 2B (see Figure 2-5). The data consists of various parameters such as Active Angle, Filtered Angle Error, and Motor Drive Current, represented as fixed point numbers at a typical sampling rate of 1Hz. BGA 4B was not functioning properly (spikes were observed in its current motor drive), so its data had to be analyzed to explore the problem. However, the time to download this data from the ISS to Earth is of the order of days. Clearly, there is a benefit to compressing the data before transmitting it, and to do so by utilizing the correlation among the data streams.

We found the Filtered Angle Error from each BGA to be highly correlated. To

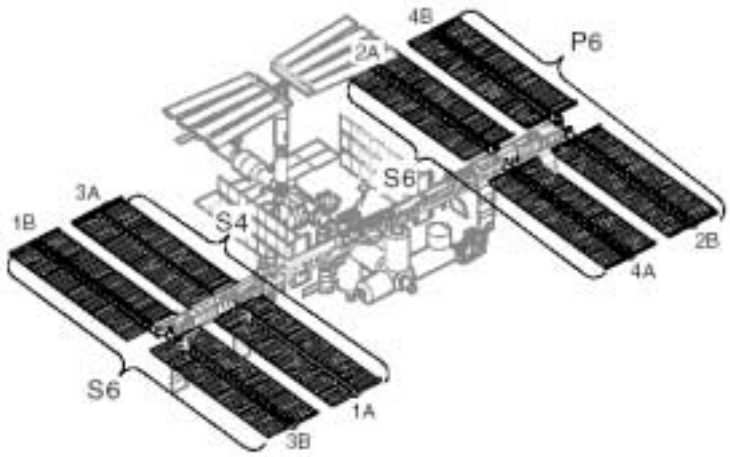


Figure 2-5: Description of the International Space Station

preprocess the data to better fit within the model of a discrete memoryless source, we took samples from each of the two data sequences (which we call U^1 and U^2), and created new sequences by taking the difference between consecutive values. These new sequences are shown in figure 2-6. The spikes in the data represent large jumps in the filtered angle error, and correlation between the location and sign of the spikes can be seen. These two sequences were further quantized into one of four values. Values less than 0.05 in magnitude were quantized to 0, values greater than 0.05 were quantized to 1, values less than -0.05 were quantized to 2, and values that were unknown owing to failures were quantized to 3. The end result was a sequence of symbols of cardinality 4, and one that more closely approached a memoryless source. We partitioned $20,000$ samples of the two data sequences into 4 blocks of length 5000. The decoding algorithm requires an a priori joint probability distribution for the two sources. We approximated this distribution by calculating the empirical distribution of the sample sequence, which is given in Table 2.1. Figure 2-7 provides the simulation results. The top plot shows the rate points that were tested for four of the splits, and the bottom plot shows the corresponding symbol error rates, as a function of the distance of the sum rate point from the joint entropy. For this particular source distribution, the splits and rate allocations given by (2.5) and (2.6)

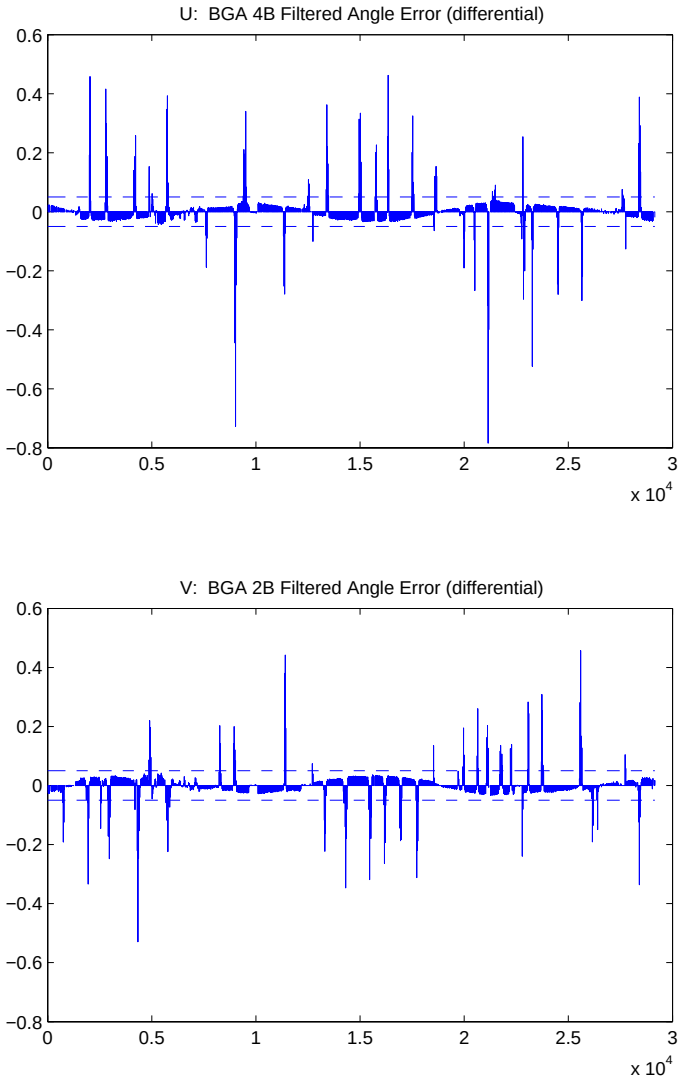


Figure 2-6: Differential of the two data sequences. The dotted lines indicate the threshold used to quantize the values.

$U^1 \backslash U^2$	0	1	2	3
0	0.9798	0.0015	0.0020	0
1	0.0020	0	0	0
2	0.0011	0	0	0
3	0.0005	0	0	0.0131

Table 2.1: Empirical joint distribution for U^1 and U^2.

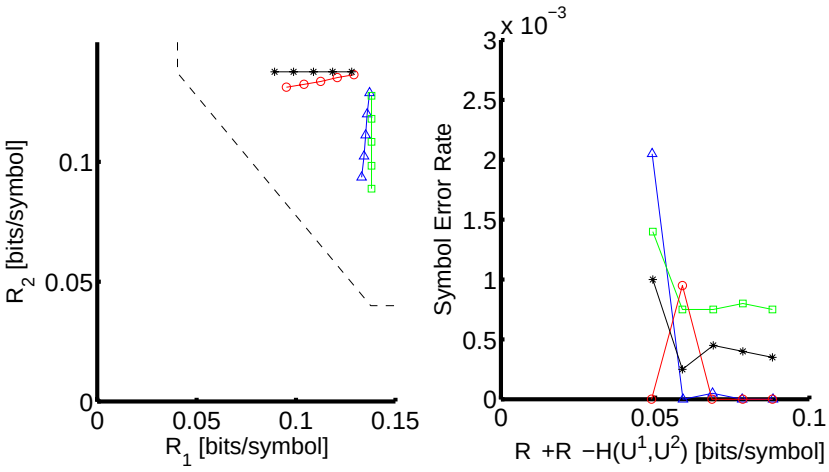


Figure 2-7: Symbol error rates for four distinct splits

did not yield a variety of uniformly spaced rate points at distinct locations across the diagonal rate boundary - instead, they were clustered. As expected, the symbol error rate generally decreased as the distance from the entropy boundary increased, and some of the splits approached zero error probability within 0.05 bits from the joint entropy bound.

Chapter 3

Provably Good Strategies for Polynomial Complexity Universal Decoding

Background: Previous discussions on universal block coding for discrete memoryless settings either illustrated the existence of universal codes but neglected to take complexity considerations into account, or discussed the penalty in decoding according to an inaccurate probability distribution. In the context of decoding for general linear codes with knowledge of the probability distribution, it has been known that optimal decoding is provably computationally complex. However, others have developed techniques to guarantee exponential error probability decay for all achievable rates with polynomial complexity for these non-universal settings - usually with a 'codes on graphs' divide and conquer approach - including linear programming decoding and 'expander codes' iterative decoding.

Our contribution: In this chapter we show that optimal universal decoding for general linear codes is also provably computationally complex. We next illustrate the existence of good universal codes that have a sparse 'codes on graphs' structure. We then discuss two polynomial complexity universal decoding approaches that guarantee exponential error probability decay. One approach has a subcomponent that uses linear programming decoding, while the other has a similar theme to the 'expander codes' iterative decoding framework - with universal decoding subcomponents.

The previous chapter discussed low-complexity encoding and decoding techniques

for the Slepian-Wolf problem when the joint probability distribution associated with the sources is known, and helps bias the decoder to operate properly. In lots of practical settings, such as fading channels and sensor networks, the cost of tracking this information may hinder robustness, energy efficiency, and stand-alone practicality issues. For instance, in wireless settings, the sender transmits a 'pilot' sequence of symbols known to the receiver so that it can estimate the channel law by studying received signal statistics. Afterwards, the receiver usually decodes the remaining transmission with a maximum likelihood (ML) decoder tuned to the estimated channel law. However, it is well known that such a 'mismatched decoding' approach has its drawbacks [MKLSS94, Lap96, GLT00] - in terms of both error rates and capacity.

In the information theory literature there has been discussion on *universal* coding, where encoders and decoders are constructed that operate *without knowledge of the underlying probability distribution*. From an ontological perspective, there has been much success - it has been shown that for numerous settings [Gop75, Csi82, LZ97, FL98], there exist block encoders and decoders that can attain the same error exponent (exponential rate of decay in probability of error) as that of the random coding exponent corresponding to maximum-likelihood decoding. Such universal decoding algorithms have also served as subcomponents of other multiterminal communication systems - for instance statistical inference problems under data compression constraints [AH89, HA95, HA98, Jor01, JY02]. As in the typical channel coding case, the encoding situation is not nearly as difficult as the decoding situation. Indeed, the proposed universal decoding algorithms' nonlinearities and difficulty of implementation have obfuscated the desire for people to construct *practical* code constructions and decoding algorithms. This apparent intrinsic difficulty in universal decoding manifests itself in the consideration of other decoding algorithms [MKLSS94, Sec. I]: "Theoretically, one can employ universal decoding; however, in many applications, it is ruled out by complexity considerations."

However, we take a fresh perspective by looking back at how key advances manifested themselves in the traditional coding literature:

- Linear codes have been known to be sufficient for many channel coding problems

to attain all achievable rates and the random coding exponent. However, ML decoding for general linear codes has been shown [BMvT78] to be an intrinsically complex (NP-complete) problem.

- A 'divide-and-conquer' approach has been employed by coding theorists since the beginnings of information theory to construct large linear codes from smaller good components with polynomial complexity decoding algorithms whose performance is empirically good [Gal62] as well as provably good [For65, For66]. Indeed, Forney's seminal work "Concatenated Codes" [For65, For66] construction illustrated how to encode and decode in polynomial time at *all* achievable rates with exponential error probability decay.

- The 'divide-and-conquer' approach to error-correcting codes has been sharpened with help of a well-formalized 'codes on graphs' perspective [For01, KFL01] as well as new 'expansion' results in graph theory [LPS88] to construct *linear* complexity iterative decoding algorithms that achieve capacity from both an empirical [BGT93, CGFRU01] as well as theoretical [SS96, Z01, BZ02] perspective. Furthermore, the iterative algorithms with empirical performance guarantees have been shown [KV03, VK04] to be deeply connected to a linear programming relaxation decoder [FWK03, Fel03, FS05] that is easier to analyze theoretically.

and try to walk an analogous path to address practical code constructions and decoding algorithms for the universal setting:

- Linear codes have already been known to *universally* attain all achievable rates with exponential error probability decay. In this chapter we show that universal decoding general linear codes is also a provably complex (NP-complete) problem.

- We employ a 'divide-and-conquer' graph-based approach to show that large linear codes constructed from smaller 'universally good' component codes are aggregately provably good under optimal universal decoding.

- With this codes on graphs perspective, we use expander graphs to construct linear codes and two polynomial complexity decoding algorithms, both of which exhibit exponential error probability decay. One approach is a linear programming relaxation based decoder inspired by [FWK03, Fel03, FS05] and the other is a linear complexity iterative decoder inspired by the 'expander codes' framework of [SS96, Z01, BZ02].

We also extend these approaches to multiterminal settings including Slepian-Wolf distributed data compression and certain types of multiple-access channel coding. It is our hope that these results will form a foundation to bring forth further developments in *efficient*, practical code and decoding designs for the universal setting.

3.1 The General Universal Decoding Problem for a Single Source

3.1.1 Model and Definitions

Throughout this discussion we will consider a discrete memoryless source (DMS) U over $\mathcal{U} = \{0, 1, \ldots, Q-1\}$. The set of all probability distributions on $\mathcal{U}$ is given by $\mathcal{P}(\mathcal{U})$. For a length-N sequence $\underline{u} = (u_1, u_2, \ldots, u_N) \in \mathcal{U}^N$, the type $P_{\underline{u}} \in \mathcal{P}(\mathcal{U})$ is the probability distribution defined by $P_{\underline{u}}(a) = \frac{1}{N}\sum_{i=1}^{N} 1_{\{u_i=a\}}$, for all $a \in \mathcal{U}$. We denote by W^N the pmf induced on $\mathcal{U}^N$ by N independent drawings according to W. We denote by $\mathcal{P}_N(\mathcal{U})$ the subset of $\mathcal{P}(\mathcal{U})$ consisting of the possible types of sequences $\underline{u} \in \mathcal{U}^N$. For any type $\mathbb{P} \in \mathcal{P}_N(\mathcal{U})$, the type class $T(\mathbb{P})$ is the set of all $\underline{u} \in \mathcal{U}^N$ such

that $P_{\underline{u}} = \mathbb{P}$. To summarize, we have that

$$\begin{aligned}
\mathcal{P}(\mathcal{U}) &= \left\{ P = \left(\{P_a\}_{a\in\mathcal{U}}\right) : P \geq \underline{0}, \sum_{a\in\mathcal{U}} P_a = 1 \right\} \\
P_{\underline{u}} &= \left(\left\{ \frac{1}{N} \sum_{i=1}^{N} 1_{u_i=a} \right\}_{a\in\mathcal{U}} \right) \text{ for } \underline{u} \in \mathcal{U}^N \\
\mathcal{P}_N(\mathcal{U}) &= \left\{ P \in \mathcal{P}(\mathcal{U}) : P = P_{\underline{u}} \text{ for some } \underline{u} \in \mathcal{U}^N \right\} \\
T(\mathbb{P}) &= \left\{ \underline{u} \in \mathcal{U}^N \mid P_{\underline{u}} = \mathbb{P} \right\}.
\end{aligned} \tag{3.1}$$

For a random variable U with probability distribution W we will denote its entropy as $H(U)$ which is a function of W. When we instead want to explicitly speak of the entropy as a function of some $P \in \mathcal{P}_N(\mathcal{U})$, then we will denote this as $h(P)$. For two random variables with conditional and marginal distributions given by $P_{U|V}$ and P_V, we will denote the conditional entropy $H(U|V)$ explicitly interms of $P_{U|V}$ and P_V as $h\left(P_{U|V}|P_V\right)$.

From [Csi98] we note the following:

$$|\mathcal{P}_N(\mathcal{U})| \leq (N+1)^{|\mathcal{U}|} \tag{3.2}$$

$$|T(\mathbb{P})| \leq 2^{Nh(\mathbb{P})} \tag{3.3}$$

$$W^n(\underline{u}) = 2^{-N\left[h\left(P_{\underline{u}}\right)+D\left(P_{\underline{u}}\|W\right)\right]} \quad \forall\, \underline{u} \in \mathcal{U}^N \tag{3.4}$$

Thus ***the number of types is polynomial in*** N.

3.1.2 The General Problem

In this discussion we consider code constructions for fixed block length universal coding for the two dual settings of data compression and channel coding. The compression scenario mentioned could be relevant, for instance, in a wireless sensor network where the following two points apply:

1) Time-varying nature of field makes knowledge of field being sensed, the probability distribution on the data is not completely accurately modeled,

2) Complexity, memory, and energy constraints make a universal fixed-to-fixed length algebraic compression approach more viable than a universal fixed-to-variable length compression approach (such as Lempel-Ziv [LZ77, LZ78] or Burrows-Wheeler [EVKV02]) that requires dictionaries and table-lookups.

Similarly, due to the time-varying and multipath effects of the wireless channel, the universal channel coding scenario could be relevant where phase information cannot be accurately tracked.

More specifically, we take interest in universal decoding for discrete memoryless settings, where the decoder does not have knowledge of the probability distribution to aid in decoding. Consider a DMS U with probability distribution $W \in \mathcal{P}(\mathcal{U})$. Without loss of generality, we assume that $\mathcal{U} = \{0, 1, \ldots, Q-1\}$ where $Q = 2^t$ for some integer $t \geq 1$. Thus we may assume that U takes on values in $\mathbb{F}_{2^t}$. Our goal is to design a fixed-rate universal code that permits a decoding algorithm that is blind to W to have provably good performance. We consider the case where a linear mapping

$$H = \begin{bmatrix} -H_1'- \\ -H_2'- \\ \vdots \\ -H_M'- \end{bmatrix} : \mathcal{U}^N \to \mathcal{U}^M$$

is used to map $\underline{u} \in \mathcal{U}^N$ to $\underline{s} \in \mathcal{U}^M$ via

$$\underline{s} = H\underline{u} \tag{3.5}$$

where $M < N$ and U is memoryless with probability distribution $W \in \mathcal{P}(\mathcal{U})$. We will denote the rate R as

$$R = t\frac{M}{N} \tag{3.6}$$

and note that this corresponds to rate in a data compression sense and *not* in a channel coding sense (which would correspond to $t - R$). Throughout the rest of this chapter we will speak of rate in a data compression sense. The decoder knows that $\underline{u}$

must be consistent with $\underline{s}$, in other words it must lie in the coset

$$\mathrm{Co}\,(H, \underline{s}) = \{\underline{u} \;\big|\; H\underline{u} = \underline{s}\}, \tag{3.7}$$

and selects $\hat{\underline{u}}$ as the 'best' coset member (in a universal sense). This encompasses two settings:

a) Fixed-to-fixed length near-lossless data compression, where $\underline{u}$ is identified as the sourceword and $\underline{s}$ is the syndrome, the output of the compression operation.

b) An additive noise channel $\underline{y} = \underline{x} \oplus \underline{u}$. By using a linear code $\mathcal{C}$ for $\underline{x}$, and identifying the parity check matrix H with $\mathcal{C}$ as

$$\mathcal{C} = \{\underline{x} : H\underline{x} = \underline{0}\}\,, \tag{3.8}$$

then we have that a sufficient statistic for decoding is

$$H\underline{y} = H\underline{u} = \underline{s}.$$

Successfully decoding for the noise vector $\underline{u}$ is equivalent to successfully decoding for the transmitted codeword $\underline{x}$:

$$\hat{\underline{x}} = \hat{\underline{u}} \oplus \underline{y}.$$

We assume that the rate R is achievable (i.e. $t\frac{M}{N} > H\,(U)$). It has been known in the information theory literature for quite a while [Gop75, Csi82] that in the *universal* setting, *linear codes* still suffice to attain all achievable rates and can the same error exponent as the random coding exponent. Note that for any $\underline{u} \in \mathcal{U}^N$, we have that

$$P(\underline{u}) = 2^{-N\left[D\left(P_{\underline{u}} \| W\right) + h\left(P_{\underline{u}}\right)\right]}.$$

Thus an ML decoder with knowledge of W operates by selecting

$$\underline{\hat{u}} \in \arg \min_{\underline{u} \in \mathrm{Co}(H, \underline{s})} D\left(P_{\underline{u}} \| W\right) + h\left(P_{\underline{u}}\right)$$

It has been shown [Csi82] that there exist *linear* codes satisfying

$$\begin{aligned} \liminf_{N \to \infty} -\frac{1}{N} \log P_e^{\mathrm{ML}}(N) &\geq E_r\left(R, W\right), \\ E_r\left(R, W\right) &= \min_{\mathbb{P} \in \mathcal{P}(\mathcal{U})} D\left(\mathbb{P} \| W\right) + \left|R - h\left(\mathbb{P}\right)\right|^+ . \end{aligned} \tag{3.9}$$

Note that $E_r\left(R, W\right) > 0$ for all $R > H(U)$. Now we note that the only dependence on the distribution W in the above equation is in $D\left(P_{\underline{u}} \| W\right)$ and from the law of large numbers, with very high probability $P_{\underline{u}} \to W$. Since $D\left(\cdot \| W\right)$ operates in a distance-like manner, is continuous, and satisfies $D\left(W \| W\right) = 0$, dropping the term $D\left(P_{\underline{u}} \| W\right)$ for a universal decoder doesn't seem completely nonsensical.

Indeed, Csiszár's 'minimum-entropy' decoder selects as the source reconstruction the coset's entropy minimizer

$$\underline{\hat{u}} \in \arg \min_{\underline{u} \in \mathrm{Co}(H, \underline{s})} h\left(P_{\underline{u}}\right). \tag{3.10}$$

In [Csi82], Csiszár shows that not only do there exist linear codes such whose rates can be arbitrarily close to $H(U)$ when such a decoder is applied, but also that minimum entropy decoding achieves the same error exponent as the optimal maximum-likelihood (ML) decoder:

$$\liminf_{N \to \infty} -\frac{1}{N} \log P_e^{\mathrm{univ}}(N) \geq E_r\left(R, W\right).$$

Another interpretation of the universal decoding paradigm is that it is a manifestation of Occam's razor: "Find the explanation most easy to accept." Since the entropy function measures the inherent uncertainty or difficulty in explaining something, selecting the lowest entropy candidate consistent with observations is the same as selecting the easiest to accept candidate consistent with observations.

3.2 Universally Good Linear Codes

Csiszár's lemma specifying good encoders [Csi82, Sec. III] illustrates the existence of linear mappings $H: \mathcal{U}^N \to \mathcal{U}^M$ such for any joint type $\mathbb{P} \in \mathcal{P}_N(\mathcal{U}^2)$ with the definitions

$$\mathcal{N}_H(\mathbb{P}) \triangleq \left| \left\{ (\underline{u} \in \mathcal{U} \mid \begin{matrix} H\underline{u} = H\tilde{\underline{u}} \\ P_{\underline{u},\tilde{\underline{u}}} = \mathbb{P} \end{matrix} \text{ for some } \tilde{\underline{u}} \neq \underline{u} \right\} \right|, \tag{3.11}$$

every joint type $\mathbb{P} \in \mathcal{P}_N(\mathcal{U}^2)$ satisfies:

$$a) \qquad \mathcal{N}_H(\mathbb{P}) \leq 2^{-N(R-h(\mathbb{P})-\delta_N)} \tag{3.12}$$

$$b) \quad \text{if } h\left(\mathbb{P}_{U-\tilde{U}}\right) \leq R - \delta_N \text{ then } \mathcal{N}_H(\mathbb{P}) = 0 \tag{3.13}$$

where $\delta_N \to 0$ as $N \to \infty$. We will denote such codes as **universally good**. Note that the bound (3.12) can be strengthened to:

$$\begin{aligned} \mathcal{N}_H(\mathbb{P}) &\leq 2^{-N(R-h(\mathbb{P})-\delta_N)} \\ &= 2^{N\left(h(\mathbb{P}_U)-\left(R-h\left(\mathbb{P}_{\tilde{U}|U}|\mathbb{P}_U\right)-\delta_N\right)\right)} \\ \Rightarrow \mathcal{N}_H(\mathbb{P}) &\leq 2^{N\left(h(\mathbb{P}_U)-\left|R-h\left(\mathbb{P}_{\tilde{U}|U}|\mathbb{P}_U\right)-\delta_N\right|^+\right)} \\ &\leq 2^{N\left(h(\mathbb{P}_U)-\left|R-h\left(\mathbb{P}_{\tilde{U}}\right)-\delta_N\right|^+\right)} \end{aligned} \tag{3.14}$$

where (3.14) follows because by the definition of $\mathcal{N}_H(\mathbb{P})$, $\mathcal{N}_H(\mathbb{P}) \leq |T(\mathbb{P}_U)| \leq 2^{Nh(\mathbb{P}_U)}$.

3.2.1 The Gilbert-Varshamov Distance

One important property of any linear code $\mathcal{C}$ with associated parity-check matrix H is its minimum distance

$$d_{\min}(H) = \min_{\underline{u} \in \mathrm{Co}(H,\underline{0}) \setminus \underline{0}} w_h(\underline{u})$$

where $w_h(\cdot)$ is the Hamming distance. It is well known that the larger the minimum distance of a code, the larger the number of errors it can guarantee to correct:

$$w_h(\underline{u}) < \frac{1}{2} d_{\min}(H) \Rightarrow w_h(\underline{u} + \underline{0}) < w_h(\underline{u} + \tilde{\underline{u}}) \ \forall \ \tilde{\underline{u}} \in \mathcal{C} \setminus \underline{0}. \tag{3.15}$$

Because of the linearity of the code $\mathcal{C}$ this can be generalized to

$$w_h(\underline{u} + \hat{\underline{u}}) < \frac{1}{2} d_{\min}(H) \Rightarrow w_h(\underline{u} + \hat{\underline{u}}) < w_h(\underline{u} + \tilde{\underline{u}}) \ \forall \ \tilde{\underline{u}} \in \mathrm{Co}(H, H\hat{\underline{u}}) \setminus \hat{\underline{u}}. \tag{3.16}$$

Here we briefly note how condition (3.13) of universally good codes relates to a standard bound on good linear codes. It has been well known that random linear codes with parity-check matrix H have minimum distance lying on the Q-ary Gilbert-Varshamov distance bound with high probability [WK03, p. 42-43]:

$$d_{\min}(H) \geq N\left(h_Q^{-1}(R) - \epsilon\right)$$

where $h_Q(\alpha)$ for $0 < \alpha \leq \frac{Q-1}{Q}$ is given by

$$h_Q(x) = x \log(Q-1) - x \log x - (1-x) \log(1-x).$$

Lemma 3.2.1. *Universally Good Linear Codes lie on the Gilbert-Varshamov bound.*

Proof. Setting $\tilde{\underline{u}} = \underline{0}$ we have from condition (3.13) of universally good codes that any $\underline{u} \in \mathrm{Co}(H, \underline{0}) \setminus \underline{0}$ satisfies $h(P_{\underline{u}}) \geq R - \epsilon_N$, where $\epsilon_N \to 0$ as $N \to \infty$. Now if we see what this means in terms of hamming distance, we can perform the following minimization:

$$\begin{aligned} \min \quad & w_h(\underline{u}) \\ s.t. \quad & h(P_{\underline{u}}) \geq R - \epsilon_N. \end{aligned}$$

From the concavity of the entropy function $h(\cdot)$, $\underline{u}^*$ will in $(1-\delta)N$ positions be 0

and in $\frac{\delta}{Q-1}N$ positions be a, for each $a \in \mathcal{U} \setminus 0$. Thus we have that

$$\begin{aligned} R - \epsilon_N &= h\left(P_{\underline{u}^*}\right) \\ &= -(1-\delta)\log(1-\delta) - (Q-1)\frac{\delta}{Q-1}\log\left(\frac{\delta}{Q-1}\right) \\ &= -(1-\delta)\log(1-\delta) - \delta\log\left(\frac{\delta}{Q-1}\right) \\ &= -(1-\delta)\log(1-\delta) - \delta\log\delta + \delta\log(Q-1) \\ &= h_Q(\delta). \quad \square \end{aligned}$$

3.2.2 Guarantees on Universal Decoding Success

In this subsection we discuss some conditions for which guaranteed universal decoding success applies. Although none of these discussions will be related to error probability analysis for our proposed decoding algorithms later in the chapter, we present them to fall in analogy with previously well-established conditions for minimum-distance decoding.

Decoding Success Guarantees for Binary Linear Codes: the Universal Distance

We now discuss the 'universal distance' of a binary linear code $\mathcal{C}$ with parity check matrix H, given by

$$d_{\text{univ}} \triangleq \min_{\underline{u} \in \mathcal{C}, \underline{u} \neq 0} \min\left(w_h\left(\underline{u}\right), w_h\left(\underline{1} \oplus \underline{u}\right)\right). \tag{3.17}$$

We illustrate the motivation for using the universal distance in code design using the following example. Consider any linear code with parity check matrix H for which $\underline{1}$ is a member of $\mathcal{C} = \text{Co}\left(H, \underline{0}\right)$. Then the minimum-entropy decoder has probability of error equal to $\frac{1}{2}$ by the following argument. For any $\underline{u} \in \text{Co}\left(H, \underline{s}\right)$, $\underline{u} \oplus \underline{1} \in \text{Co}\left(H, \underline{s}\right)$. Further, $h\left(P_{\underline{u}}\right) = h\left(P_{\underline{u} \oplus \underline{1}}\right)$. Thus $\underline{u}$ and $\underline{u} \oplus \underline{1}$ are indistinguishable to a minimum-entropy decoder. Note that the universal distance of any such linear code H is 0, which captures this undesirable effect.

We know that from (3.15) under minimum-distance decoding, if the error sequence has hamming weight less than half the minimum distance, then we can guarantee success. It is natural to consider an analogous statement regarding universal distance and minimum-entropy decoding:

Lemma 3.2.2. *Consider any M by N binary matrix H and its associated $d_{univ} = N\delta_{univ}$, given by (3.17). If $w_h(\underline{u}) < \frac{1}{2}d_{univ}$* ***or*** *$w_h(\underline{u} \oplus \underline{1}) < \frac{1}{2}d_{univ}$, then $\underline{u}$ is the unique solution to*

$$\min_{\underline{\hat{u}} \in Co(H,\underline{s})} h\left(P_{\underline{\hat{u}}}\right).$$

Stated alternatively, if $h(P_{\underline{u}}) < h_b\left(\frac{1}{2}\delta_{univ}\right)$ then $\underline{u}$ is the unique solution to

$$\min_{\underline{\hat{u}} \in Co(H,\underline{s})} h\left(P_{\underline{\hat{u}}}\right).$$

Proof: see Appendix.

Decoding Success Guarantees for Non-Binary Codes

Here we would like to speak to condition (3.13) of universally good codes. Define the *universal rate* R_{univ} associated with matrix H to be the largest R such that condition (3.13) holds for H with $\delta_N = 0$.

Lemma 3.2.3. *Consider any M by N $Q-ary$ matrix H over $\mathbb{F}_{2^t}$ and its associated R_{univ}. If $h(P_{\underline{u}}) < \frac{1}{2}R_{univ}$ then $\underline{u}$ is the unique solution to*

$$\min_{\underline{\hat{u}} \in Co(H,\underline{s})} h\left(P_{\underline{\hat{u}}}\right).$$

Proof. Note that

$$\begin{aligned} h\left(P_{\underline{u}-\underline{\tilde{u}},\underline{u},\underline{\tilde{u}}}\right) &= h\left(P_{\underline{u},\underline{\tilde{u}}}\right) + h\left(P_{\underline{u}-\underline{\tilde{u}}|\underline{u},\underline{\tilde{u}}}|P_{\underline{u},\underline{\tilde{u}}}\right) = h\left(P_{\underline{u},\underline{\tilde{u}}}\right) && (3.18)\\ h\left(P_{\underline{u}-\underline{\tilde{u}},\underline{u},\underline{\tilde{u}}}\right) &= h\left(P_{\underline{u}-\underline{\tilde{u}}}\right) + h\left(P_{\underline{u},\underline{\tilde{u}}|\underline{u}-\underline{\tilde{u}}}|P_{\underline{u}-\underline{\tilde{u}}}\right) && (3.19)\\ \Rightarrow h\left(P_{\underline{u}-\underline{\tilde{u}}}\right) &= h\left(P_{\underline{u},\underline{\tilde{u}}}\right) - h\left(P_{\underline{u},\underline{\tilde{u}}|\underline{u}-\underline{\tilde{u}}}|P_{\underline{u}-\underline{\tilde{u}}}\right) && (3.20)\\ &\leq h\left(P_{\underline{u},\underline{\tilde{u}}}\right) && (3.21)\\ &\leq h\left(P_{\underline{u}}\right) + h\left(P_{\underline{\tilde{u}}}\right). && (3.22) \end{aligned}$$

Now we proceed with a proof by contradiction. Suppose $H\underline{\tilde{u}} = H\underline{u}$ and $h\left(P_{\underline{\tilde{u}}}\right) \leq h\left(P_{\underline{u}}\right)$. Then note that

$$\begin{aligned} h\left(P_{\underline{u}-\underline{\tilde{u}}}\right) &\leq h\left(P_{\underline{u}}\right) + h\left(P_{\underline{\tilde{u}}}\right)\\ &\leq 2h\left(P_{\underline{u}}\right)\\ &< R_{\text{univ}} \end{aligned}$$

by definition there can be no such $\underline{\tilde{u}} \neq \underline{u}$ with $H\underline{\tilde{u}} = H\underline{u}$ and thus we have a contradiction. □

3.2.3 (β, E) universal robustness: Error Exponent Generalization

We now perform a generalization of the error exponent analysis of universally good codes exhibited by Csiszár[Csi82]. This generalization will be useful in the linear programming framework for universal decoding, to be introduced later in this chapter. Suppose that $\underline{U}$ has been transformed by H to $\underline{s}$ according to (3.5). Consider the event

$$\mathcal{E}_{\beta}^{\text{univ}} = \left\{\exists\, \underline{\tilde{u}} \neq \underline{U} \mid \underline{\tilde{u}} \in \text{Co}\left(H, \underline{s}\right), h\left(P_{\underline{\tilde{u}}}\right) \leq h\left(P_{\underline{U}}\right) + \beta\right\}$$

where $\beta \geq 0$. We say that a linear mapping H is (β, E) *universally robust* if

$$-\frac{1}{N}\log P\left(\mathcal{E}_{\beta}^{\text{univ}}\right) \geq E.$$

By defining the set

$$\mathcal{T}_{\beta}^{\text{univ}} \triangleq \left\{ \mathbb{P} \in \mathcal{P}_N\left(\mathcal{U}^2\right) \mid h\left(\mathbb{P}_{\tilde{U}}\right) \leq h\left(\mathbb{P}_U\right) + \beta \right\}$$

we have that universally good codes satisfy

$$\begin{aligned}
P\left(\mathcal{E}_{\beta}^{\text{univ}}\right) &\leq \sum_{\mathbb{P} \in \mathcal{T}_{\beta}^{\text{univ}}} \mathcal{N}_H\left(\mathbb{P}\right) 2^{-N[D(\mathbb{P}_U \| W) + h(\mathbb{P}_U)]} && (3.23)\\
&\leq \sum_{\mathbb{P} \in \mathcal{T}_{\beta}^{\text{univ}}} 2^{n\left(h(\mathbb{P}_U) - \left|R - h\left(\mathbb{P}_{\tilde{U}}\right) - \delta_N\right|^+\right)} 2^{-N[D(\mathbb{P}_U \| W) + h(\mathbb{P}_U)]} && (3.24)\\
&\leq (N+1)^{\mathcal{U}^2} \max_{\mathbb{P} \in \mathcal{T}_{\beta}^{\text{univ}}} \left[2^{-N\left[D(\mathbb{P}_U \| W) + \left|R - h\left(\mathbb{P}_{\tilde{U}}\right) - \delta_N\right|^+\right]} \right] && (3.25)\\
&\leq 2(N+1)^{\mathcal{U}^2} \max_{\mathbb{P} \in \mathcal{T}_{\beta}^{\text{univ}}} \left[2^{-N\left[D(\mathbb{P}_U \| W) + \left|R - h\left(\mathbb{P}_{\tilde{U}}\right)\right|^+\right]} \right] && (3.26)\\
&\leq 2(N+1)^{\mathcal{U}^2} \max_{\mathbb{P} \in \mathcal{T}_{\beta}^{\text{univ}}} \left[2^{-N\left[D(\mathbb{P}_U \| W) + |R - h(\mathbb{P}_U) - \beta|^+\right]} \right] && (3.27)\\
&\leq 2(N+1)^{\mathcal{U}^2} \max_{\mathbb{P} \in \mathcal{P}_n(\mathcal{U})} \left[2^{-N\left[D(\mathbb{P}_U \| W) + |R - h(\mathbb{P}_U) - \beta|^+\right]} \right] && (3.28)\\
&= 2(N+1)^{\mathcal{U}^2} 2^{-N E_r(R-\beta, W)}. && (3.29)
\end{aligned}$$

Thus universally good codes of rate R are also (β, E) universally robust whenever $0 \leq \beta > R - H(U)$ and $0 < E \leq E_r\left(R - \beta, W\right)$. Note that for $\beta = 0$ and $E = E_r\left(R, W\right)$ we get the typical random coding bound.

3.2.4 (β, E) robustness for ML decoding

Unsurprisingly, we now make the same claim about an ML decoder. In [FS05] a Gallager-style [Gal68] error exponent derivation approach such good codes in the binary case [FS05]. Here we use the method of types and show that for any alphabet, universally good codes are also (β, E)-robust with respect to ML decoding. Consider the event

$$\mathcal{E}_{\beta}^{\text{ML}} = \left\{ \exists\, \underline{\tilde{u}} \neq \underline{U} \mid \underline{\tilde{u}} \in \text{Co}\left(H, \underline{s}\right), D\left(P_{\underline{\tilde{u}}} \| W\right) + h\left(P_{\underline{\tilde{u}}}\right) \leq D\left(P_{\underline{U}} \| W\right) + h\left(P_{\underline{U}}\right) + \beta \right\}.$$

We say that a linear mapping H is (β, E) *ML robust* if

$$-\frac{1}{N}\log P\left(\mathcal{E}_{\beta}^{\mathrm{ML}}\right) \geq E.$$

By defining the set

$$\mathcal{T}_{\beta}^{\mathrm{ML}} = \left\{\mathbb{P} \in \mathcal{P}_N\left(\mathcal{U}^2\right) \mid D\left(\mathbb{P}_{\tilde{U}} \| W\right) + h\left(\mathbb{P}_{\tilde{U}}\right) \leq D\left(\mathbb{P}_U \| W\right) + h\left(\mathbb{P}_U\right) + \beta\right\}$$

we have that universally good codes also satisfy

$$\begin{aligned}
P\left(\mathcal{E}_{\beta}^{\mathrm{ML}}\right) &\leq \sum_{\mathbb{P} \in \mathcal{T}_{\beta}^{\mathrm{ML}}} \mathcal{N}_H\left(\mathbb{P}\right) 2^{-N\left[D\left(\mathbb{P}_U \| W\right) + h\left(\mathbb{P}_U\right)\right]} && (3.30)\\
&\leq \sum_{\mathbb{P} \in \mathcal{T}_{\beta}^{\mathrm{ML}}} 2^{n\left(h\left(\mathbb{P}_U\right) - \left|R - h\left(\mathbb{P}_{\tilde{U}}\right) - \delta_N\right|^+\right)} 2^{-N\left[D\left(\mathbb{P}_U \| W\right) + h\left(\mathbb{P}_U\right)\right]} && (3.31)\\
&\leq (N+1)^{\mathcal{U}^2} \max_{\mathbb{P} \in \mathcal{T}_{\beta}^{\mathrm{ML}}} \left[2^{-N\left[D\left(\mathbb{P}_U \| W\right) + \left|R - h\left(\mathbb{P}_{\tilde{U}}\right) - \delta_N\right|^+\right]}\right] && (3.32)\\
&\leq 2(N+1)^{\mathcal{U}^2} \max_{\mathbb{P} \in \mathcal{T}_{\beta}^{\mathrm{ML}}} \left[2^{-N\left[D\left(\mathbb{P}_U \| W\right) + \left|R - h\left(\mathbb{P}_{\tilde{U}}\right)\right|^+\right]}\right]. && (3.33)
\end{aligned}$$

Let us consider the constraint for any $\mathbb{P} \in \mathcal{T}_{\beta}^{\mathrm{ML}}$:

$$\begin{aligned}
D\left(P_{\underline{\tilde{u}}} \| W\right) + h\left(P_{\underline{\tilde{u}}}\right) &\leq \beta + D\left(P_{\underline{u}} \| W\right) + h\left(P_{\underline{u}}\right) && (3.34)\\
\Leftrightarrow D\left(P_{\underline{\tilde{u}}} \| W\right) - h\left(P_{\underline{u}}\right) - \beta &\leq D\left(P_{\underline{\tilde{u}}} \| W\right) - h\left(P_{\underline{\tilde{u}}}\right). && (3.35)
\end{aligned}$$

If $h\left(P_{\underline{u}}\right) > R - \beta$, then

$$\begin{aligned}
D\left(P_{\underline{u}} \| W\right) + \left|R - h\left(P_{\underline{\tilde{u}}}\right)\right|^+ &\geq D\left(P_{\underline{u}} \| W\right)\\
&= D\left(P_{\underline{u}} \| W\right) + \left|R - \beta - h\left(P_{\underline{u}}\right)\right|^+.
\end{aligned}$$

If on the other hand $h\left(P_{\underline{u}}\right) < R - \beta$:

$$\begin{aligned}
D\left(P_{\underline{\tilde{u}}}\|W\right) + \left|R - \beta - h\left(P_{\underline{u}}\right)\right|^{+} &= D\left(P_{\underline{\tilde{u}}}\|W\right) + \left(R - \beta - h\left(P_{\underline{u}}\right)\right) \\
&\leq D\left(P_{\underline{u}}\|W\right) + \left(R - h\left(P_{\underline{\tilde{u}}}\right)\right) \text{ via (3.35)} \\
&\leq D\left(P_{\underline{u}}\|W\right) + \left|R - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+} \\
\Rightarrow D\left(P_{\underline{u}}\|W\right) + \left|R - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+} &= \frac{1}{2}\left[D\left(P_{\underline{u}}\|W\right) + \left|R - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+}\right] \\
&+ \frac{1}{2}\left[D\left(P_{\underline{u}}\|W\right) + \left|R - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+}\right] \\
&\geq \frac{1}{2}\left[D\left(P_{\underline{\tilde{u}}}\|W\right) + \left|R - \beta - h\left(P_{\underline{u}}\right)\right|^{+}\right] \\
&+ \frac{1}{2}\left[D\left(P_{\underline{u}}\|W\right) + \left|R - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+}\right] \\
&\geq \frac{1}{2}\left[D\left(P_{\underline{\tilde{u}}}\|W\right) + \left|R - \beta - h\left(P_{\underline{u}}\right)\right|^{+}\right] \\
&+ \frac{1}{2}\left[D\left(P_{\underline{u}}\|W\right) + \left|R - \beta - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+}\right] \\
&= \frac{1}{2}\left[D\left(P_{\underline{u}}\|W\right) + \left|R - \beta - h\left(P_{\underline{u}}\right)\right|^{+}\right] \\
&+ \frac{1}{2}\left[D\left(P_{\underline{\tilde{u}}}\|W\right) + \left|R - \beta - h\left(P_{\underline{\tilde{u}}}\right)\right|^{+}\right] \\
&\geq E_r\left(R - \beta, W\right).
\end{aligned}$$

Thus it follows that

$$P\left(\mathcal{E}_{\beta}^{\mathrm{ML}}\right) \leq 2(N+1)^{\mathcal{U}^2} 2^{-NE_r(R-\beta,W)}$$

so universally good codes of rate R are also (β, E) ML robust whenever $0 \leq \beta > R - H(U)$ and $0 < E \leq E_r\left(R - \beta, W\right)$.

3.3 The Complexity of Universal Decoding with Linear Codes

Here we consider the computational complexity issues involving performing minimum-entropy decoding for general linear codes. We consider the binary case here - and illustrate that even in this setting complexity issues manifest themselves. Consider a binary linear code $\mathcal{C}$ specified by its parity check matrix $H \in \{0,1\}^M \times \{0,1\}^N$, given

by (3.8). We consider the case where a linear mapping $H : \{0,1\}^N \to \{0,1\}^M$ is used to map $\underline{u} \in \{0,1\}^N$ to $\underline{s} \in \{0,1\}^M$ via (3.5), where $M < N$ and U is memoryless with $P(U_i = 1) = p$. The decoder knows that $\underline{u}$ must be consistent with $\underline{s}$, in other words it must lie in the coset $\mathrm{Co}\,(H, \underline{s})$ given by (3.7). In the case of ML decoding, if the decoder knew that $p < \frac{1}{2}$, then it selects $\hat{\underline{u}}$ as the coset's smallest hamming weight member - termed *coset leader*:

$$\hat{\underline{u}} = \arg \min_{\underline{u} \in \mathrm{Co}(H, \underline{s})} w_h\,(\underline{u})\,. \tag{3.36}$$

In a universal setting, the decoder is unaware of the sign of $p - \frac{1}{2}$, and selects $\hat{\underline{u}}$ as the coset's empirical entropy minimizer, given by (3.10).

It has been shown in [BMvT78] that ML decoding for general linear codes - performing (3.36) for a general matrix H - is NP-complete. Thus it is not overwhelmingly surprising the following theorem holds, but we state it here for the sake of completeness, a solid foundation, and motivation for future approximation techniques in this chapter:

Theorem 3.3.1. *The algorithm* ***MINIMUM-ENTROPY***$[H, \underline{s}]$ *for general binary linear codes is NP-complete.*

Proof. Our approach to proving this will be the usual suspect in complexity theory: a reduction. Our base NP-complete problem will be **COSET-LEADER**$[H, \underline{s}]$, and we will reduce it to **MINIMUM-ENTROPY**$[H, \underline{s}]$.

Suppose we are given an instance of the problem **COSET-LEADER**$[H, \underline{s}]$, which performs (3.36). We would like to reduce this to minimum-entropy decoding by showing that if there exists a polynomial-time algorithm for **MINIMUM-ENTROPY**$[H, \underline{s}]$, which performs (3.10), then it can be used to solve any instance of **COSET-LEADER**$[H, \underline{s}]$. Consider the coset $\mathrm{Co}\left(\tilde{H}, \tilde{\underline{s}}\right)$ where

$$\tilde{H} = \begin{bmatrix} H & 0 \\ 0 & I_N \end{bmatrix}, \quad \tilde{\underline{s}} = \begin{bmatrix} \underline{s} \\ \underline{0} \end{bmatrix},$$

where I_N is the $N \times N$ identity matrix. Consider any $\tilde{\underline{u}} = \begin{bmatrix} \underline{u} \\ \underline{u}' \end{bmatrix} \in \mathrm{Co}\left(\tilde{H}, \tilde{\underline{s}}\right)$ and note that $\tilde{\underline{u}}$ must satisfy

$$\underline{u}' = 0, \quad \underline{u} \in \mathrm{Co}\left(H, \underline{s}\right). \tag{3.37}$$

Furthermore, note that any $\tilde{\underline{u}} \in \mathrm{Co}\left(\tilde{H}, \tilde{\underline{s}}\right)$ satisfies

$$\frac{1}{2N} w_h\left(\tilde{\underline{u}}\right) \leq \frac{1}{2} \quad \Leftrightarrow \quad P_{\tilde{\underline{u}}}(0) \leq \frac{1}{2}. \tag{3.38}$$

Since the binary entropy function is monotonically increasing on $[0, \frac{1}{2})$, we have that the two optimization problems

$$\begin{aligned} \textbf{MINIMUM-ENTROPY}\left[\tilde{H}, \tilde{\underline{s}}\right]: \quad & \min_{\tilde{\underline{u}} \in \mathrm{Co}\left(\tilde{H}, \tilde{\underline{s}}\right)} h\left(P_{\tilde{\underline{u}}}\right), \\ \textbf{COSET-LEADER}\left[\tilde{H}, \tilde{\underline{s}}\right]: \quad & \min_{\tilde{\underline{u}} \in \mathrm{Co}\left(\tilde{H}, \tilde{\underline{s}}\right)} w_h\left(\tilde{\underline{u}}\right) \end{aligned}$$

have the same optimal solution(s). Let $\tilde{\underline{u}}^* = \begin{bmatrix} \underline{u}^* \\ \underline{u}'^* \end{bmatrix}$ be an optimal solution from above. Then from (3.37) and we have that $\underline{u}^*$ is an optimal solution to **COSET-LEADER**$[H, \underline{s}]$. □

3.4 Codes on Graphs

Here we will be interested in graphical realizations of linear systems to describe the coset $\mathrm{Co}\left(H, \underline{s}\right)$. Here we will use Forney's "normal graph realizations approach" [For01, Sec VIII.B]. For our discussion, we will discuss normal realizations whose graphical models have the following properties:

- A graph $G = (V, E \cup \bar{E})$ with $|E| = N$ two-sided edges and $\left|\bar{E}\right|$ one-sided edges. For a vertex $j \in V$ we denote $\Gamma\left(j\right)$ as the set of edges $e \in E$ adjacent to j and $\tilde{\Gamma}\left(j\right)$ as the set of edges $\bar{e} \in \bar{E}$ adjacent to j.

- Each local constraint $\mathcal{C}_j$ is represented by a vertex $j \in V$

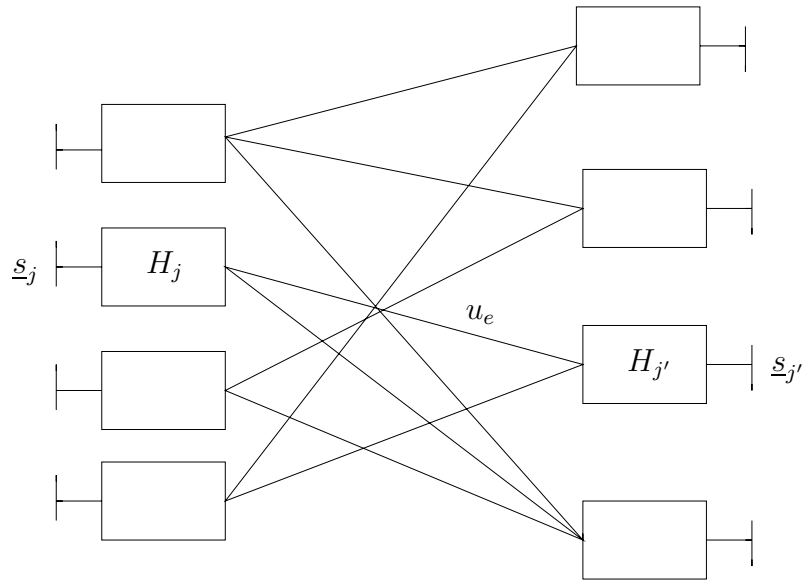


Figure 3-1: Graphical representation of a linear system representing $\mathrm{Co}(H, \underline{s})$

- The state variable $u_e \in \mathbb{F}_{2^t}$ corresponds to a two-sided edge $e \in E$ and is involved in the two local constraints corresponding to the vertices adjacent to e. Thus there are N total state variables and as a vector they are represented as $\underline{u}$. For a vertex $j \in V$ we abbreviate $\{u_e\}_{e \in \Gamma(j)}$ as $\underline{u}_j$.

- The symbol variable $s_{\bar{e}} \in \mathbb{F}_{2^t}$ corresponds to a one-sided 'leaf-edge' (also termed a 'half-edge' or 'dongle') $\bar{e} \in \bar{E}$ and is associated with the one local constraint, corresponding to the vertex adjacent to the half-edge $\bar{e}$. For a vertex $j \in V$, we abbreviate $\{s_{\bar{e}}\}_{\bar{e} \in \tilde{\Gamma}(j)}$ as $\underline{s}_j$.

- Each vertex j and its associated code C_j imposes the constraint that

$$H_j \underline{u}_j + \underline{s}_j = 0 \;\Leftrightarrow\; H_j \underline{u}_j = \underline{s}_j \;\Leftrightarrow\; \underline{u}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right). \tag{3.39}$$

Fundamentally important, the coset $\mathrm{Co}\left(H, \underline{s}\right)$ can be expressed as

$$\mathrm{Co}\left(H, \underline{s}\right) = \left\{ \underline{u} \;\middle|\; \underline{u}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right),\ \forall\, j \in V \right\}. \tag{3.40}$$

We would like to emphasize that there are *more than one* ways to decompose $\mathrm{Co}\,(H, \underline{s})$ into a graphical realization. Indeed, the key to low-complexity decoding algorithms arises in using graphical representations of $\mathrm{Co}\,(H, \underline{s})$ that have nice properties. For a particular graph $G = (V, E)$ we denote $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$ as the way in which we specify $\mathrm{Co}\,(H, \underline{s})$ in terms of (3.40).

3.4.1 Parity-Check Representation

One such code on graph representation, perhaps the simplest, is the *parity-check representation* where we express H as

$$H = \begin{bmatrix} - & H_1' & - \\ - & H_2' & - \\ - & \dots & - \\ - & H_M' & - \end{bmatrix}, \quad \underline{s} = \begin{bmatrix} s_1, \\ s_2, \\ \dots, \\ s_M \end{bmatrix}$$

and representing G with $V = A \cup B$, where $|A| = N$ and $|B| = M$. As illustrated in Figure 3-2, each $j \in A$ corresponds to a repetition code where each edge adjacent to j must be equivalent. Each $j \in B$ corresponds to a row in the H matrix and enforces the constraint that

$$H_j' \underline{u} = s_j.$$

3.4.2 Universal Expander Codes

Now we consider building a code on graph where G has a particularly nice structure that facilitates practical decoding with provably good performance. Here we consider a graphical representation of $\mathrm{Co}\,(H, \underline{s})$ where G is a bipartite *expander graph*. Thus $V = A \cup B$ where $|A| = |B| = \frac{1}{2}|V| = n$ and each edge $e \in E$ has one endpoint in A and one in B. Each node $j \in V$ is adjacent to Δ edges and has $M_j \leq \Delta$ dongles corresponding to $\underline{s}_j$.

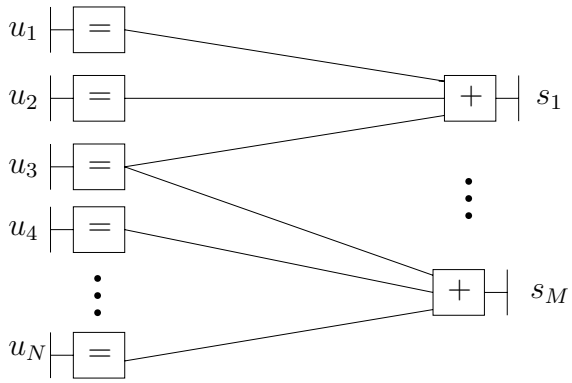


Figure 3-2: Parity-check representation for the coset $\mathrm{Co}(H, \underline{s})$

The idea here is to think of concatenated codes as in Forney [For66] where we let Δ be a *large but fixed* quantity. Thus there are $N = n\Delta$ edges and N corresponds to the block length of our codes. We let every code $\mathcal{C}_j$ for $j \in A$ have M_a constraints where

$$H(U) < R_a = \frac{M_a}{\Delta} < R$$

and let every C_j for $v \in B$ have M_b constraints where

$$R_b = t\frac{M_b}{\Delta} = t\left(\frac{M}{N} - \frac{M_a}{\Delta}\right) = R - R_a.$$

The idea here is that each local code $\mathcal{C}_j$ for $j \in A$ is locally achievable to block compress U to its entropy, i.e. $R_a > H(U)$. The codes $\mathcal{C}_j$ are nearly rateless and their main purpose is to correct a constant fraction of errors in Δ. Moreover, we make each local H_j be universally good.

Properties of Expander Graphs

Expander graphs have the property that every sufficiently small set of vertices has a very large number of neighbors - despite the fact that it is Δ-regular where Δ does not grow with $|V|$.

A common way to characterize the expansion properties of a graph G is to examine the second-largest eigenvalue $\lambda_2(G)$ of its adjacency matrix. The largest eigenvalue

of any Δ-regular graph is Δ, and the larger the eigenvalue separation, the better the expansion properties:

Lemma 3.4.1. *[AC89] Let $G = (V, E)$ be a Δ-regular graph such that $\lambda_2(G) = \lambda$. Let T be a subset of the vertices of G of size $\alpha |V|$. Then, the number of edges contained in the subgraph induced by T in G is at most $\alpha |V| \left(\frac{\Delta\alpha}{2} + \frac{\lambda}{2}(1-\alpha)\right)$.*

The best possible value of $\lambda_2(G)$ is known to be $2\sqrt{\Delta - 1}$ and it surprisingly can be explicitly achieved with polynomial-time constructions [LPS88]. We denote a graph G as $G_{\Delta,n}$ when G is a Δ-regular bipartite graph with $|V| = 2n$ and $\lambda_2(G) = 2\sqrt{\Delta - 1}$. Moreover, we define ECOG $\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ to be the graphical representation of Co $(H, \underline{s})$ where each H_j is universally robust. From here we discuss some properties of expander graphs discussed in [FS05].

Definition [FS05] We will say that a Δ-regular graph G is a (α, ρ)-expander if for every induced graph $G' = (V', E')$ with $|V'| \leq \alpha |V|$, we have $|E'| \leq \rho\Delta |V'|$.

We note that this definition is slightly different than the typical notion of an expander graph [SS96, Sec. III], but will be more convenient for our analysis. We will use Lemma 3.4.1 as in [FS05] to construct a Δ-regular graph that is an (α, ρ)-expander where $\alpha = 2\rho - \frac{\lambda}{\Delta}$.

Definition [FS05] A ρ-orientation of a subgraph $G' = (V', E')$ of a Δ-regular graph G is an assignment of directions to every edge in E' such that each node in V' contains at most $\rho\Delta$ incoming edges from E'.

With this definition we can state the following lemma, which will be useful in our error probability analysis:

Lemma 3.4.2. *[FS05] If a Δ-regular graph G is a (α, ρ)-expander where $\rho\Delta$ is an integer, then all subgraphs $G' = (V', E')$ with $|E'| \leq \alpha\rho\Delta |V|$ contain a ρ-orientation.*

3.4.3 Universal Goodness of Bipartite Graph Codes

Here we consider how a bipartite graph code of the form $\mathrm{ECOG}\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ performs under minimum-entropy decoding, as Δ grows. We first consider the following lemma that will be useful for our analysis:

Lemma 3.4.3. *Consider a set A where $|A| = n$ and suppose that $\{\mathbb{P}^j \in \mathcal{P}_\Delta(\mathcal{U}^2)\}_{j\in A}$, $\mathbb{P} \in \mathcal{P}_{n\Delta}(\mathcal{U}^2)$. Then:*

$$\sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j=\mathbb{P}} \prod_{j\in A} 2^{-\Delta\left(R_a - h\left(\mathbb{P}^j\right)-\delta_\Delta\right)} \leq 2^{-n\Delta\left(R_a-h(\mathbb{P})-\epsilon'_\Delta\right)}$$

where $\epsilon'_\Delta \to 0$ as $\Delta \to \infty$.

Proof.

$$\begin{aligned}
\sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j=\mathbb{P}} \prod_{j\in A} 2^{-\Delta\left(R_a - h\left(\mathbb{P}^j\right)-\delta_\Delta\right)} &= \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j=\mathbb{P}} 2^{-n\Delta\left(R_a - \frac{1}{n}\sum_{j\in A} h\left(\mathbb{P}^j\right)-\delta_\Delta\right)} && (3.41)\\
&\leq \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j=\mathbb{P}} 2^{-n\Delta\left(R_a - h\left(\sum_{j\in A}\frac{1}{n}\mathbb{P}^j\right)-\delta_\Delta\right)} && (3.42)\\
&= \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j=\mathbb{P}} 2^{-n\Delta(R_a - h(\mathbb{P})-\delta_\Delta)}\\
&\leq \left|\mathcal{P}_\Delta\left(\mathcal{U}^2\right)\right|^n 2^{-n\Delta(R_a-h(\mathbb{P})-\delta_\Delta)}\\
&\leq (\Delta+1)^{n|\mathcal{U}|^2} 2^{-n\Delta(R_a-h(\mathbb{P})-\delta_\Delta)} && (3.43)\\
&= 2^{-n\Delta\left(R_a-h(\mathbb{P})-\delta_\Delta-|\mathcal{U}|^2\frac{\log(\Delta+1)}{\Delta}\right)}\\
&= 2^{-n\Delta\left(R_a-h(\mathbb{P})-\epsilon'_\Delta\right)}
\end{aligned}$$

where (3.42) follows from the concavity of entropy, and (3.43) follows from (3.2). □

Now let $N = n\Delta$ and $\mathbb{P} \in \mathcal{P}_N(\mathcal{U}^2)$ correspond to the joint type of any length-N pair $(\underline{u}, \tilde{\underline{u}})$ satisfying $H\underline{u} = H\tilde{\underline{u}}$. Define $\mathbb{P}^j \in \mathcal{P}_\Delta(\mathcal{U}^2)$ to correspond to the joint type

of any *local* length-Δ pair $(\underline{u}', \tilde{\underline{u}}')$ satisfying $H_j \underline{u}' = H_j \tilde{\underline{u}}'$. Then we have:

$$\begin{aligned}
\mathcal{N}_H(\mathbb{P}) &\leq \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j = \mathbb{P}} \prod_{j\in A} \mathcal{N}_{H_j}\left(\mathbb{P}^j\right) \\
&\leq \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j = \mathbb{P}} \prod_{j\in A} 2^{-\Delta\left(R_a - h\left(\mathbb{P}^j\right) - \delta_\Delta\right)} \qquad (3.44) \\
&\leq 2^{-N\left(R_a - h(\mathbb{P}) - \epsilon'_\Delta\right)} \qquad (3.45)
\end{aligned}$$

where $\epsilon'_\Delta \to 0$ as $\Delta \to \infty$, (3.44) follows from (3.12), (3.45) follows from Lemma 3.4.3. Thus it follows that H becomes universally good for large Δ, when thought of having rate $R' = R_a - \epsilon'_\Delta$.

3.4.4 Encoding

Encoding for the compression situation is done quite simply. $\underline{u}$ is mapped to $\underline{s}$ setting the edges on the graph G to $\underline{u}$, and applying $\underline{s}_j = H_j \underline{u}_j$ for all $j \in V$. We note that there are n nodes and each node has degree Δ, and since there are $N = n\Delta$ edges, this is done with linear complexity. For the channel coding scenario, the encoding done is the same as discussed in [SS96, BZ02].

3.5 Linear Programming Decoding Methods with Polynomial Complexity

Here we will be interested in executing *linear programs* to solve a *provably good* approximation to the decoding problem (3.10) with *polynomial complexity* when $\mathrm{Co}(H, \underline{s})$ is specified in a codes on graphs description $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$. This approach is inspired by the work of Feldman [Fel03, FMS+04, FS05], which addressed linear programming relaxations to ML decoding in the binary alphabet case. Here we will consider *universal* decoding in *multiterminal* settings over *non-binary* alphabets.

A linear program (LP) is an optimization problem of the form

$$\begin{aligned} \min \quad & \underline{c}'\underline{x} && (3.46a) \\ s.t. \quad & A\underline{x} = \underline{b} && (3.46b) \\ & \underline{x} \geq 0. && (3.46c) \end{aligned}$$

The constraint set is a polyhedron, which is simply a finite intersection of half-spaces. In the event that the polyhedron is bounded, it is termed a polytope. For any polytope $\mathcal{B}$ we say that a point $v \in \mathcal{B}$ is a *vertex* if it cannot be expressed as a convex combination of two other elements of $\mathcal{B}$. The convex hull of a finite set $\mathcal{S} \subseteq \mathbb{R}^n$ of points is simply the set

$$\left\{ x \; s.t. \; x = \sum_{s_i \in S} \lambda_i s_i, \sum_i \lambda_i = 1, \lambda_i \geq 0 \right\}$$

Some model and definitions we abide by are as follows:

$$\begin{aligned} CH(\mathcal{S}) &\triangleq \text{ the convex hull of } \mathcal{S} \\ \mathcal{V}(\mathcal{B}) &\triangleq \{v : v \text{ is a vertex of the polytope } \mathcal{B}\} \\ \mathcal{H}(\mathcal{B}) &\triangleq \text{ the number of half-spaces representing } \mathcal{B} \end{aligned}$$

One fundamental property of linear programming is the following [BT97, Sec. 2.6]: *if an LP has an optimal solution then there exists a vertex which is optimal*. Linear programs fall within the class of convex optimization problems that exhibit strong duality [BT97, Sec 4.3]. Consequently, for every linear program given by (3.46) - which we call the *primal* LP - with an optimal solution, there exists another corresponding *dual* LP given by

$$\begin{aligned} \max \quad & \underline{p}'\underline{b} && (3.47a) \\ s.t. \quad & \underline{p}'A \leq \underline{c}' && (3.47b) \end{aligned}$$

with an optimal solution of the same cost. Finally, LPs can be performed efficiently in the sense that the LP $\{\min \underline{c}'\underline{x} \ s.t. \ \underline{x} \in \mathcal{B} \subseteq \mathbb{R}^N\}$ has an algorithmic complexity upper bounded by a *polynomial* function of $\mathcal{H}(\mathcal{B})$ and N.

3.5.1 From Discrete to Continuous Optimization

Note that (3.10) is a discrete optimization problem with an exponential number $(2^{N(t-R)})$ of feasible solutions. Our first step is to replace (3.10) by a continuous optimization problem. We first construct indicator variables $I_{a,e} \in \{0,1\}$, for $a \in \mathcal{U}, e \in \{1,\ldots,N\}$, such that $I_{a,e} = 1_{\{u_e=a\}}$. Thus $I_{a,e}$ specifies $\underline{u} \in \mathcal{U}^n$ as

$$\underline{u} \ = \ \mu(I), \text{ where } u_e = \mu_e(I) = \sum_{a\in\mathcal{U}} a I_{a,e}. \tag{3.48}$$

Note that any $\underline{u} \in \mathrm{Co}(H,\underline{s})$ must satisfy the constraints of the linear code. We impose these code constraints on I by defining

$$\mathcal{I}(H,\underline{s}) = \{I \ s.t. \ \mu(I) \in \mathrm{Co}(H,\underline{s})\}. \tag{3.49}$$

For any $I \in \mathcal{I}(H,\underline{s})$ and the corresponding $\underline{u} = \mu(I)$, we can construct $P_{\underline{u}}$ as a linear mapping

$$P = \tau(I), \text{ where } P(a) = \tau_a(I) = \frac{1}{N}\sum_{e=1}^{N} I_{a,e}, \ \ a \in \mathcal{U}.$$

Thus we can define the polytope $\mathcal{B}^{i,p}(H,\underline{s})$ as

$$\mathcal{B}^{i,p}(H,\underline{s}) = \{(I,P) \ s.t. \ I \in CH(\mathcal{I}(H,\underline{s})), \ P = \tau(I)\}.$$

Note that for every $(I,P) \in \mathcal{V}(\mathcal{B}^{i,p}(H,\underline{s}))$:

- I corresponds to a coset member $\underline{u} = \mu(I) \in \mathrm{Co}(H,\underline{s})$.
- The empirical type $P_{\underline{u}}$ associated with $\underline{u} = \mu(I)$ satisfies $P_{\underline{u}} = P = \tau(I)$.

Since the entropy function is *strictly concave*, and since minimizing a strictly concave function over a polytope $\mathcal{B}$ has the property [HT96] that an optimal solution lies in

$\mathcal{V}(\mathcal{B})$, we can perform (3.10) in the continuous domain as

$$\min \quad h(P) \tag{3.50a}$$

$$s.t. \quad (I,P) \in \mathcal{B}^{i,p}(H,\underline{s}) \tag{3.50b}$$

and take the minimum-entropy solution as $\underline{u}^* = \mu(I^*)$ where (I^*, P^*) is an optimal solution to (3.50). At first glance, there are two difficulties that arise in trying to perform (3.50):

1) Since ML-decoding for linear codes is generally NP-complete [BMvT78], the best bound on $\mathcal{H}(\mathcal{B})$ (and thus $\mathcal{H}(\mathcal{B}^{i,p})$) is $O(2^n)$. As a result, it is not obvious how to efficiently represent $\mathcal{B}^{i,p}$.

2) In (3.50), $|\mathcal{V}(\mathcal{B}^{i,p})| = O(2^n)$ and concave minimization over a polytope is NP-hard [HT96] - generally requiring a visit to every $v \in \mathcal{V}(\mathcal{B}^{i,p})$.

However, even though $|\mathrm{Co}(H,\underline{s})| = O(2^n)$, from (3.2) it follows that the number of distinct *types* associated with $\mathrm{Co}(H,\underline{s})$ is *polynomial* in n. This observation suggests somehow restricting our attention to the vertices of the projected polytope $\mathcal{B}^p(H,\underline{s})$, given by

$$\mathcal{B}^p(H,\underline{s}) = \{P \mid (I,P) \in \mathcal{B}^{i,p}(H,\underline{s}) \text{ for some } I\}$$

Note that any $P \in \mathcal{V}(\mathcal{B}^p(H,\underline{s}))$ is the type of some $\underline{u} \in \mathrm{Co}(H,\underline{s})$.

3.5.2 LP Decoding for Arbitrary $\mathbb{F}_{2^t}$

Without loss of generality we define $\mathcal{U} = \{0,1,\ldots,2^t-1\}$. For $a \in \mathcal{U}$, define $\gamma_a \triangleq -\log_2 P(U_i = a)$. We note that the ML decoder selects $\underline{u}^*(\underline{s}) = \mu(I^*(\underline{s}))$ where

$$\begin{aligned} I^*(\underline{s}) &= \arg \min_{I \in \mathcal{I}(H,\underline{s})} \left(\sum_e \sum_{a \in \mathcal{U}} \gamma_a I_{a,e} \right) \\ &= \arg \min_{I \in CH[\mathcal{I}(H,\underline{s})]} \left(\sum_e \sum_{a \in \mathcal{U}} \gamma_a I_{a,e} \right). \end{aligned}$$

Without loss of generality we assume that $\gamma_a < \infty$ so that this is well-defined. As already discussed, this is an NP-complete problem and thus there is no polynomial bound on $\mathcal{H}(\mathcal{B})$. So we consider an LP relaxation where we minimize the same objective function over a different polytope $\tilde{\mathcal{B}}$ that is easily represented in the sense that $\mathcal{H}\left(\tilde{\mathcal{B}}\right)$ is polynomial in n. This approach is inspired from Feldman [Fel03, FMS$^+$04, FS05] but here we consider the arbitrary alphabet case and we will eventually lead to universal decoding algorithms, with this approach as a subcomponent.

Here we consider performing an LP for an arbitrary $\mathrm{Co}(H, \underline{s})$ where a codes on graphs representation, as discussed in Section 3.4 is provided: $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$. For any node $j \in V$ we have a local code u_j associated with a subset of the constraints, H_j and $\underline{s}_j$. Analogous to Feldman's approach, we construct our LP relaxation to be the intersection of all polytopes that are locally consistent with $(H_j, \underline{s}_j)$.

LP-PRIMAL$\left(G, \{\gamma_a\}, \{H_j\}, \{\underline{s}_j\}\right)$

$$\begin{aligned}
\min \quad & \sum_e \sum_{a \in \mathcal{U}} \gamma_a I_{a,e} \quad s.t. \\
\forall\, j: \quad & \sum_{\underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right)} w_{j,\underline{\tilde{u}}_j} = 1 && \text{(3.51a)} \\
\forall\, e = (j, j'),\ a \in \mathcal{U}: \quad & I_{a,e} = \sum_{\substack{\underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right): \\ \tilde{u}_j[e] = a}} w_{j,\underline{\tilde{u}}_j} = \sum_{\substack{\underline{\tilde{u}}_{j'} \in \mathrm{Co}\left(H_{j'}, \underline{s}_{j'}\right): \\ \tilde{u}_{j'}[e] = a}} w_{j',\underline{\tilde{u}}_{j'}} && \text{(3.51b)} \\
& w \geq 0, I \geq 0 && \text{(3.51c)}
\end{aligned}$$

where $w_{j,\underline{\tilde{u}}_j}$ corresponds to a convex hull variable associated with $\underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right)$. We shall denote the primal relaxed polytope $\tilde{\mathcal{B}}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$ as the set of all (I, w) satisfying the above constraints (3.51a)-(3.51c).

We note that there are $Q^{|\Gamma(j)| - \left|\tilde{\Gamma}(j)\right|}$ variables of the type $w_{j,\underline{\tilde{u}}_j}$ associated with each node $j \in V$ and there are $|E|\,|\mathcal{U}|$ variables of the type $I_{a,e}$. So by defining

$$\bar{\Gamma}(G) = \max_{j \in V} |\Gamma(j)| - \left|\tilde{\Gamma}(j)\right| \tag{3.52}$$

$$N' = |V|\, Q^{\bar{\Gamma}(G)} + |E|\,|\mathcal{U}| \tag{3.53}$$

we have that

$$\tilde{\mathcal{B}}\left(G, \{H_j\}, \{\underline{s}_j\}\right) \subseteq \mathbb{R}^{N'}.$$

Since there are $|V|$ constraints associated with (3.51a), $|E|\,|\mathcal{U}|$ constraints associated with (3.51b) and at most N' constraints associated with (3.51c),

$$\mathcal{H}\left(\tilde{\mathcal{B}}\left(G, \{H_j\}, \{\underline{s}_j\}\right)\right) = O\left(N' + |V| + |E|\,|\mathcal{U}|\right).$$

We note that

- For a family of codes $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$ where $\bar{\Gamma}(G)$ does not grow with $|E|$, we note that both N' and $\mathcal{H}\left(\tilde{\mathcal{B}}\left(G, \{H_j\}, \{\underline{s}_j\}\right)\right)$ are $O(N)$, and thus performing **LP-PRIMAL**$\left(G, \{\gamma_a\}, \{H_j\}, \{\underline{s}_j\}\right)$is guaranteed to have running time polynomial in the block length N. We note that low-density parity-check codes $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$ according to Section 3.4.1 fall within this category, as do the universal expander codes $\mathrm{ECOG}\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$.

All valid coset members still correspond to vertices $\tilde{\mathcal{B}}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$, but non-integral vertices, termed pseudocodewords, also arise and thus compete in the LP. By [FKKR01, FKV01, KV03], pseudocodewords also compete with true codewords when the min-sum algorithm is applied to the same graphical realization $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$. Furthermore, [KV03] shows that the region over which the pseudocodewords compete with true codewords is in fact $\tilde{\mathcal{B}}$. Discussions in [KV03, Fel03, VK04] suggest that the two decoders have essentially the same performance. This gives another motivation for considering the LP decoding paradigm - it is more amenable to concrete analysis and is intimately connected to iterative decoding algorithms.

The Dual LP

In a dual LP, we will have the constraints of the form $\underline{p}'A_k \leq c_k$ for each column k of A. Each p_m variable corresponds to one of the constraints (rows) in the matrix A of the primal LP. Note that for a fixed $a \in \mathcal{U}$, any particular edge $e = (j, j')$, will be involved in exactly two constraint equations of the form (3.51b). Thus we can

define those particular p_m variables to be of the form $(\tau_{e,j,a}, \tau_{e,j',a})$. Also note that the column A_k in A, where k corresponds to one of the $I_{a,e}$ variables, will have exactly two positive ones, corresponding to the two equations (3.51b) involving edge e. Thus we get dual constraints of the form

$$\tau_{e,j,a} + \tau_{e,j',a} \leq \gamma_a.$$

Also note that there will be equations of the form (3.51a) for each node $j \in V$ corresponding to a local code. Thus we define those particular p_m variables to be v_j. The column A_k in A, where k corresponds to one of the $w_{j,\underline{u}_j}$ variables, will have a 1 in a row location corresponding to node j in (3.51a) as well as values of -1 in row locations (3.51b) corresponding to each edge $e \in N(j)$. Thus we get dual constraints of the form

$$v_j - \sum_{e \in N(j)} \sum_{a \in \mathcal{U}} \mathbb{I}_{\underline{\tilde{u}}_j, a, e} \tau_{e,j,a} \leq 0,$$

where $\mathbb{I}_{\underline{\tilde{u}}_j, a, e} = 1_{\left\{\underline{\tilde{u}}_j[e]=a\right\}}$. Thus the dual LP becomes
LP-DUAL$\big(G, \{\gamma_a\}, \{H_j\}, \{\underline{s}_j\}\big)$:

$$\begin{aligned}
& & \max \quad & \sum_{j \in V} v_j \ \ s.t. & \\
\forall\, j,\ \forall \underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right): \quad & & & \sum_{e \in N(j)} \sum_{a \in \mathcal{U}} \mathbb{I}_{\underline{\tilde{u}}_j, a, e} \tau_{e,j,a} \geq v_j & \text{(3.54a)} \\
\forall\, e = (j, j'),\ \forall\, a \in \mathcal{U}: \quad & & & \tau_{e,j,a} + \tau_{e,j',a} \leq \gamma_a & \text{(3.54b)}
\end{aligned}$$

3.5.3 Performance Guarantees for LP Decoding on Universal Expander Codes

Consider the decoding problem of trying to infer $\underline{u}$ provided the information $\mathrm{ECOG}\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ and the log-likelihoods $\{\gamma_a\}$. We define the cost $\gamma\left(\underline{\tilde{u}}_j, j\right)$

of a local codeword $\underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right)$ at node j as

$$\gamma\left(\underline{\tilde{u}}_j, j\right) = \sum_a \sum_{e \in \Gamma(j)} \mathbb{I}_{\underline{\tilde{u}}_j, a, e} \gamma_a \tag{3.55}$$

Let us now define

$$T(\underline{u}) \triangleq \left\{ j \in A \mid \exists\, \underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right) \setminus \underline{u}_j, \gamma\left(\underline{\tilde{u}}_j, j\right) - \gamma\left(\underline{u}_j, j\right) \leq \beta\Delta \right\} \tag{3.56}$$

$$\mathcal{N}_{\mathrm{bad}}(\underline{u}) \triangleq |T(\underline{u})|\,. \tag{3.57}$$

Since each H_j is universally robust, with exponentially high probability in Δ a node j in A will not lie in $T(\underline{u})$. Thus any $j \in T(\underline{u})$ we label "bad". This leads to the following theorem.

Theorem 3.5.1. *Suppose $\underline{s} = H\underline{u}$ where $Co(H, \underline{s})$ can be represented in terms of $COG\left(G, \{H_j\}, \{\underline{s}_j\}\right)$. Consider an instance of **LP-PRIMAL**$\left(G, \{\gamma_a\}, \{H_j\}, \{\underline{s}_j\}\right)$ where the following conditions are satisfied:*

1) $0 < \min_{a \in \mathcal{U}} \gamma_a < \max_{a \in \mathcal{U}} \gamma_a < \bar{\gamma} < \infty$

2) For all $j \in A$, H_j is $(\beta, E_r(R_a - \beta, W))$ robust

3) G is an (α, ρ) expander where

$$\begin{aligned}
0 < \delta_A &= \min_{j \in A} \frac{d_{\min}(H_j)}{\Delta} \\
0 < \delta_B &= \min_{j \in B} \frac{d_{\min}(H_j)}{\Delta} \\
\rho' &= \frac{\delta_B}{1 + \delta_B/\delta_A + \bar{\gamma}/\beta} \\
\frac{1}{2}\rho' &\leq \rho \in \mathbb{Z} \leq \rho' \\
0 < \alpha &= 2\rho - \frac{\lambda_2(G)}{\Delta}
\end{aligned}$$

Then if $\mathcal{N}_{bad}(\underline{u}) \leq n\frac{\alpha}{\Delta+1}$ then the LP decoder succeeds with $\underline{u}$ as the unique optimal solution.

Proof. We define $\Gamma(T)$ to be the nodes in the neighborhood of T and note that because G is bipartite, $\Gamma(T) \subseteq B$. We will explicitly provide a dual feasible solution with cost equal to the primal cost of $\underline{u}$. Note that the cost of $\underline{u}$ in the primal LP is

$$\sum_{j \in A} \gamma\left(\underline{u}_j, j\right),$$

which is an upper bound on the cost of any dual solution. We now show we can achieve this cost, which means that $\underline{u}$ is indeed an optimal solution. We define

$$\begin{aligned}
\hat{\tau}_{e,j,a} &= \begin{cases} \gamma_a, & j \in A \\ 0 & j \in B \end{cases} && (3.58)\\
v_j &= \sum_{e \in N(j)} \sum_{a \in \mathcal{U}} \mathbb{I}_{\underline{u}_j,a,e} \hat{\tau}_{e,j,a} && (3.59)\\
&= \begin{cases} \gamma\left(\underline{u}_j, j\right), & j \in A \\ 0 & j \in B \end{cases} && (3.60)\\
\tau_{e,j,a} &= \hat{\tau}_{e,j,a} \mathbb{I}_{\underline{u}_j,a,e} + \tilde{\tau}_{e,j,a}(1 - \mathbb{I}_{\underline{u}_j,a,e}) && (3.61)
\end{aligned}$$

where $\tilde{\tau}_{e,j,a}$ is yet to be specified. Note that

$$\sum_{j \in V} v_j = \sum_{j \in A} \gamma\left(\underline{u}_j, j\right)$$

which is the cost of $\underline{u}$ in the primal. Also note that for $j \in V$, with $\tilde{\underline{u}}_j = \underline{u}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right)$:

$$\begin{aligned}
\sum_{e \in N(j)} \sum_{a \in \mathcal{U}} \mathbb{I}_{\underline{u}_j,a,e} \tau_{e,j,a} &= \sum_{e \in N(j)} \sum_{a \in \mathcal{U}} \mathbb{I}_{\underline{u}_j,a,e} \left(\hat{\tau}_{e,j,a} \mathbb{I}_{\underline{u}_j,a,e} + \tilde{\tau}_{e,j,a}(1 - \mathbb{I}_{\underline{u}_j,a,e})\right) && (3.62)\\
&= \sum_{e \in N(j)} \sum_{a \in \mathcal{U}} \mathbb{I}_{\underline{u}_j,a,e} \hat{\tau}_{e,j,a} && (3.63)\\
&= v_j. && (3.64)
\end{aligned}$$

So the dual constraint (3.54a) is satisfied for $\tilde{\underline{u}}_j = \underline{u}_j \in \mathrm{Co}\left(H_j, s_j\right)$.

We have to be careful with how we set the auxiliary edge weights $(\tilde{\tau}_{e,j,a}, \tilde{\tau}_{e,j',a})$ for

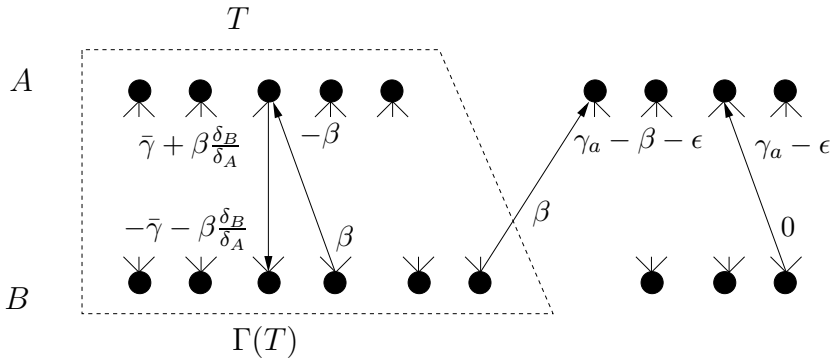


Figure 3-3: Edge weight $\tilde{\tau}_{e,j,a}$ settings for each node j

the dual constraint where $\underline{\tilde{u}}_j \in \text{Co}\left(H_j, \underline{s}_j\right) \setminus \underline{u}_j$. To do this, we define a direction for each edge in the graph. All edges that are not incident to T are directed toward the nodes A . Edges incident to T are directed according to a ρ-orientation of the subgraph induced by $(T \cup \Gamma(T))$. This is possible using Lemma 3.4.2, since $|T \cup \Gamma(T)| \leq \alpha n$ by assumption, and so $|\{\Gamma(j)\}_{j \in T}| \leq \alpha\rho\Delta n$ by expansion. To satisfy the edge constraints (3.54b) of the dual LP, the sum of these two weights should be strictly less than γ_a. We give the assignment in detail below (also see Figure 3.5.3), where $\epsilon > 0$ is a small constant the we specify later:

edge location	$\tilde{\tau}_{e,j,a},\ j \in A$	$\tilde{\tau}_{e,j,a},\ j \in B$
e leaving T	$\bar{\gamma} + \beta\frac{\delta_B}{\delta_A}$	$-\bar{\gamma} - \beta\frac{\delta_B}{\delta_A}$
e entering T	$-\beta$	β
e incident to $\Gamma(T)$ but not T	$\gamma_a - \beta - \epsilon$	β
all other e (not incident to T or $\Gamma(T)$)	$\gamma_a - \epsilon$	0

Note that from summing up up each row in the weighting assignments, all the dual constraints given by (3.54b) are satisfied with slack. Thus from complementary slackness [BT97, Sec 4.3] $\underline{u}^*$ will be the *unique* solution. We now show the above weight assignment also satisfies all the dual constraints given by (3.54a):

(i) For a node $j \in T$, there are at most $\rho\Delta \leq \rho'\Delta$ incoming edges e with weight $-\beta$. The remaining (outgoing) edges have weight $\bar{\gamma} + \beta\frac{\delta_B}{\delta_A}$. Furthermore, for any $\underline{\tilde{u}}_j \in \text{Co}\left(H_j, \underline{s}_j\right) \setminus \underline{u}_j$ we have that $w_h\left(\underline{u}_j + \underline{\tilde{u}}_j\right) \geq \delta_A\Delta$ so in the worst case, we have that

1. $\rho'\Delta$ edges, in locations where $\underline{\tilde{u}}_j[e] \neq \underline{u}_j[e]$, have weight $\tilde{\tau}_{e,j,a} = -\beta$

2. $(\delta_A - \rho')\Delta$ edges, in locations where $\underline{\tilde{u}}_j[e] \neq \underline{u}_j[e]$, have weight $\tilde{\tau}_{e,j,a} = \bar{\gamma} + \beta\frac{\delta_B}{\delta_A}$

3. the remaining $(1-\delta_A)\Delta$ edges - denoted as $\tilde{E}(j)$ - in locations where $\underline{\tilde{u}}_j[e] = \underline{u}_j[e]$, have weight $\hat{\tau}_{e,j,a} = \gamma_a$.

Thus

$$\begin{aligned}
\sum_{a\in\mathcal{U}}\sum_{e\in\Gamma(j)} \mathbb{I}_{\underline{\tilde{u}}_j,a,e}\tau_{e,j,a} &\geq -\rho'\Delta\beta + (\delta_A-\rho')\Delta\left(\bar{\gamma}+\beta\frac{\delta_B}{\delta_A}\right) + \sum_{a\in\mathcal{U}}\sum_{e\in\tilde{E}(j)} \mathbb{I}_{\underline{u}_j,a,e}\gamma_a \\
&= -\rho'\Delta\beta + (\delta_A-\rho')\Delta\left(\bar{\gamma}+\beta\frac{\delta_B}{\delta_A}\right) \\
&+ v_j - \sum_{a\in\mathcal{U}}\sum_{e\in(\Gamma(j)\setminus\tilde{E}(j))} \mathbb{I}_{\underline{u}_j,a,e}\gamma_a \\
&\geq -\rho'\Delta\beta + (\delta_A-\rho')\Delta\left(\bar{\gamma}+\beta\frac{\delta_B}{\delta_A}\right) + v_j - \delta_A\Delta\bar{\gamma} \\
&= v_j + \Delta\left[\beta\delta_B - \rho'\left(\bar{\gamma}+\beta+\beta\frac{\delta_B}{\delta_A}\right)\right] \\
&= v_j.
\end{aligned}$$

(ii) For a node $j \in \Gamma(T)$, there are at most $\rho\Delta \leq \rho'\Delta$ incoming edges e with weight $\tilde{\tau}_{e,j,a} = -\bar{\gamma} - \beta\frac{\delta_B}{\delta_A}$ and the remaining (outgoing) edges have weight $\tilde{\tau}_{e,j,a} = \beta$. Furthermore, for any $\underline{\tilde{u}}_j \in \mathrm{Co}\left(H_j, \underline{s}_j\right) \setminus \underline{u}_j$ we have that $w_h\left(\underline{u}_j + \underline{\tilde{u}}_j\right) \geq \delta_B\Delta$ so in the worst case, we have that

1. $\rho'\Delta$ edges, in locations where $\underline{\tilde{u}}_j[e] \neq \underline{u}_j[e]$, have weight $\tilde{\tau}_{e,j,a} = -\bar{\gamma} - \beta\frac{\delta_B}{\delta_A}$

2. $(\delta_B - \rho')\Delta$ edges, in locations where $\underline{\tilde{u}}_j[e] \neq \underline{u}_j[e]$, have weight $\tilde{\tau}_{e,j,a} = \beta$

3. the remaining $(1-\delta_B)\Delta$ edges, in locations where $\underline{\tilde{u}}_j[e] = \underline{u}_j[e]$, have weight $\hat{\tau}_{e,j,a} = 0$.

Thus

$$\begin{aligned}\sum_{a\in\mathcal{U}}\sum_{e\in\Gamma(j)}\mathbb{I}_{\tilde{\underline{u}}_j,a,e}\tau_{e,j,a} &\geq -\rho'\Delta\left(\bar{\gamma}+\beta\frac{\delta_B}{\delta_A}\right)+(\delta_B-\rho')\Delta\beta\\ &= \Delta\left[\delta_B\beta-\rho'\left(\bar{\gamma}+\beta+\beta\frac{\delta_B}{\delta_A}\right)\right]\\ &= 0\\ &= v_j\end{aligned}$$

(iii) For a node $j\in(A\setminus T)$, we have that every incident edge e is incoming and has weight $\tilde{\tau}_{e,j,a}$ equal to either $\gamma_a-\epsilon$ or $\gamma_a-\beta-\epsilon$. In the worst case, they all have weight $\tilde{\tau}_{e,j,a}=\gamma_a-\beta-\epsilon$. So for any $\underline{u}\neq\tilde{\underline{u}}\in\mathrm{Co}\left(H_j,\underline{s}_j\right)$ we have

$$\begin{aligned}\sum_{a\in\mathcal{U}}\sum_{e\in\Gamma(j)}\mathbb{I}_{\tilde{\underline{u}}_j,a,e}\tau_{e,j,a} &\geq \sum_{a\in\mathcal{U}}\sum_{e\in\Gamma(j)}\mathbb{I}_{\tilde{\underline{u}}_j,a,e}\gamma_a-\Delta(\beta+\epsilon)\\ &= \gamma\left(\tilde{\underline{u}}_j,j\right)-\Delta(\beta+\epsilon)\\ &> \Delta(\beta+\epsilon)+\gamma\left(\underline{u}_j,j\right)-\Delta(\beta+\epsilon)\\ &= \gamma\left(\underline{u}_j,j\right)\\ &= v_j\end{aligned}$$

where the second inequality holds for small enough ϵ because $j\in(A\setminus T)$ and H_j is robust.

(iv) For a node $j\in(B\setminus\Gamma(T))$, we have every edge is leaving j and thus we have that $\tilde{\tau}_{e,j,a}=0$ for all a and all $e\in\Gamma(j)$. Thus

$$\sum_{a\in\mathcal{U}}\sum_{e\in\Gamma(j)}\mathbb{I}_{\tilde{\underline{u}}_j,a,e}\tau_{e,j,a}=v_j. \qquad \square$$

The Universal LP Decoding Algorithm

Here we assume that $\underline{u}\in\mathcal{U}^N$ has been mapped to $\underline{s}$ via H, and the decoder has knowledge of the COG $\left(G,\{H_j\},\{\underline{s}_j\}\right)$ representation:

UNIV-DEC-LP$\big(G, \{H_j\}, \{\underline{s}_j\}\big)$

0. Set $\underline{u}^* \doteq \tilde{\underline{u}}$ for any $\tilde{\underline{u}}$ satisfying $P_{\tilde{\underline{u}}} = \mathbb{U}$.
1. **For** each $\mathbb{P} \in \mathcal{P}_\Delta(\mathcal{U})$ **do**
2. Execute **LP-PRIMAL**$\big(G, \{-\log \mathbb{P}(a)\}, \{H_j\}, \{\underline{s}_j\}\big)$.
3. **If** the optimal solution I^* to **LP-PRIMAL**$\big(G, \{-\log \mathbb{P}(a)\}, \{H_j\}, \{\underline{s}_j\}\big)$ is integral and $h\left(P_{\mu(I^*)}\right) \leq h\left(P_{\underline{u}^*}\right)$ **then** set $\underline{u}^* \doteq \mu(I^*)$.
4. **end for**
5. **return** $\underline{u}^*$

where $\mathbb{U}$ is the uniform probability distribution. Note that this algorithm has complexity proportional to the complexity of executing a single instance of **LP-PRIMAL**$\big(G, \{-\log \mathbb{P}(a)\}, \{H_j\}, \{\underline{s}_j\}\big)$.

Error Probability Analysis

Let us define $\mathbb{P}^*_{\Delta,W} \in \mathcal{P}_\Delta(\mathcal{U})$ as the following:

$$\mathbb{P}^*_{\Delta,W} \in \arg \min_{\mathbb{P} \in \mathcal{P}_\Delta(\mathcal{U})} |\mathbb{P} - W|_1 .$$

Note that $\lim_{\Delta\to\infty} \mathbb{P}^*_{\Delta,W} \to W$. Thus in step 2 of **UNIV-DEC-LP**$\big(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\big)$, when $\mathbb{P} = \mathbb{P}^*_{\Delta,W}$, with very high probability $\underline{u}$ will be the optimal solution to **LP-PRIMAL**$\big(G, \{-\log \mathbb{P}(a)\}, \{H_j\}, \{\underline{s}_j\}\big)$.

Theorem 3.5.2. *For sufficiently large but fixed Δ, the family of code constructions ECOG$\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ exhibits exponential error probability decay in n under* ***UNIV-DEC-LP***$\big(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\big)$ *decoding.*

Proof. Let us define the events

$$\mathcal{E}_1 \triangleq \left\{\mathcal{N}_{\text{bad}}(\underline{U}) \geq n\frac{\alpha}{\Delta+1}\right\} \tag{3.65}$$

$$\mathcal{E}_2 \triangleq \{h(P_{\tilde{\underline{u}}}) \leq h(P_{\underline{U}}) \text{ for some } \tilde{\underline{u}} \neq \underline{U} \text{ where } H\tilde{\underline{u}} = H\underline{U}\}. \tag{3.66}$$

Note that the error event E for **UNIV-DEC-LP**$\big(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\big)$ can be expressed

as $E \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$. Thus we have that

$$\begin{aligned} P(E) &\leq P(\mathcal{E}_1 \cup \mathcal{E}_2) \\ &\leq P(\mathcal{E}_1) + P(\mathcal{E}_2)\,. \end{aligned}$$

Let us define the event

$$\begin{aligned} \mathcal{E}(\underline{U}_j, \beta, j) = \Big\{ & \exists \underline{\tilde{u}} \in \mathrm{Co}\left(H_j, \underline{s}_j\right) \setminus \underline{u}_j \ s.t.\ H_j \underline{\tilde{u}} = H_j \underline{u}_j, \\ & D\left(P_{\underline{\tilde{u}}} \| \mathbb{P}^*_{\Delta,W}\right) + h\left(P_{\underline{\tilde{u}}}\right) \leq D\left(P_{\underline{u}_j} \| \mathbb{P}^*_{\Delta,W}\right) + h\left(P_{\underline{u}_j}\right) + \beta \Big\} \end{aligned}$$

and note that the indicator variable $1_{\left\{\mathcal{E}(\underline{U}_j, \beta, j)\right\}}$ has the property that

$$\begin{aligned} p^1_\Delta &\triangleq E\left\{1_{\left\{\mathcal{E}(\underline{U}_j, \beta, j)\right\}}\right\} \\ &= P\left(\mathcal{E}(\underline{U}_j, \beta, j)\right) \\ &\leq 2^{-\Delta[E_r(R_a - \beta, W) - \nu_\Delta]}. \end{aligned} \tag{3.67}$$

where $\nu_\Delta \to 0$ in (3.67) follows because $\mathbb{P}^*_{\Delta,W} \to W$. So we have

$$\begin{aligned} \mathcal{N}_{\mathrm{bad}}\left(\underline{U}\right) &= \sum_{j \in A} 1_{\left\{\mathcal{E}(\underline{U}_j, \beta, j)\right\}}, \\ E[\mathcal{N}_{\mathrm{bad}}\left(\underline{U}\right)] &= n p^1_\Delta \end{aligned}$$

Note that for any $0 < \beta < R - R_a$ we define p_Δ^2 and note that it satisfies

$$\begin{aligned} p_\Delta^2 &= \frac{\alpha}{\Delta+1} \\ &= \frac{1}{\Delta+1}\left(2\rho - \frac{\lambda_2(G)}{\Delta}\right) \\ &= \frac{1}{\Delta+1}\left(2\rho - 2\frac{\sqrt{\Delta-1}}{\Delta}\right) \\ &\geq \frac{1}{\Delta+1}\left(\rho' - \frac{2}{\sqrt{\Delta}}\right) \\ &= \frac{1}{\Delta+1}\left(\frac{\delta_B}{1+\delta_B/\delta_A+\bar{\gamma}/\beta} - \frac{2}{\sqrt{\Delta}}\right) \\ &= \frac{1}{\Delta+1}\left(\frac{\delta_B}{1+\delta_B/\delta_A+\log(\Delta)/\beta} - \frac{2}{\sqrt{\Delta}}\right) \\ &= \Theta\left(\frac{1}{\Delta\log\Delta}\right) \end{aligned}$$

because $\lambda_2(G) = 2\sqrt{\Delta-1}$. Note that our condition for LP decoding success is

$$\begin{aligned} \mathcal{N}_{\text{bad}}\left(\underline{u}\right) &\leq n\frac{\alpha}{\Delta+1} \\ &= np_\Delta^2 \end{aligned}$$

So for sufficiently large Δ, we have $p_\Delta^2 > p_\Delta^1$. Combining this with how the random variables $\left\{1_{\left\{\mathcal{E}(\underline{U}_j,\beta,j)\right\}}\right\}_{j\in A}$ are i.i.d. because G is bipartite, we have

$$\begin{aligned} P\left(\mathcal{E}_1\right) = P\left(\mathcal{N}_{\text{bad}}\left(\underline{U}\right) > np_\Delta^2\right) &= 2^{n[o(n)]}P\left(\mathcal{N}_{\text{bad}}\left(\underline{U}\right) = np_\Delta^2\right) \\ &\leq 2^{-n\left[D\left(p_\Delta^2\|p_\Delta^1\right)-o(n)\right]}. \end{aligned}$$

Note that even if $\mathcal{E}_1^c$ occurs, an error can still occur of there is another $\tilde{\underline{u}} \neq \underline{u}$ with smaller empirical entroy. This corresponds to event $\mathcal{E}_2$. Characterizing $\mathcal{E}_2$ is straightforward because by Section 3.4.3 our constituent codes are good:

$$P\left(\mathcal{E}_2\right) \leq 2^{-NE_r\left(R_a-\nu_\Delta^2,W\right)}$$

where $\nu_\Delta^2 \to 0$ as $\Delta \to \infty$. □

3.6 Iterative Decoding Decoding Methods with Linear Complexity

Here we will be interested in using *iterative decoding* to solve a *provably good* approximation to the decoding problem (3.10) with *linear complexity* when $\mathrm{Co}\left(H, \underline{s}\right)$ is specified in a codes on graphs description $\mathrm{COG}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$. This approach is inspired by the work of Barg and Zémor [BZ02], which addressed approximations to ML decoding over the binary symmetric channel. Here we will consider *universal* decoding in *multiterminal* settings over *non-binary* alphabets.

Let $\underline{u} \in \{0,1\}^N$ be the true sequence that has been mapped to $\underline{s} \in \{0,1\}^M$ according to (3.5). Before we describe our decoding algorithm, we present two subcomponents. Let us define

$$\begin{aligned} d_{ME}\left(H, \underline{s}\right) &= \arg \min_{\tilde{\underline{u}} \in \mathrm{Co}(H, \underline{s})} h\left(P_{\tilde{\underline{u}}}\right) && (3.68) \\ d_{MD}\left(H, \underline{s}, \underline{u}'\right) &= \arg \min_{\tilde{\underline{u}} \in \mathrm{Co}(H, \underline{s})} w_h\left(\tilde{\underline{u}} + \underline{u}'\right) && (3.69) \\ \phi = \phi(\{H_j\}) &= \min_{j \in V} \frac{1}{2} d_{\min}(H_j) && (3.70) \\ \lambda &= \lambda_2(G) && (3.71) \end{aligned}$$

Technically speaking there might be more than one solution to each of the top two above optimization problems so we say that the output of the functions in such a case is any optimal solution. Our algorithm contains an iteration counter variable $i \in \mathbb{Z}$, a fixed point detection variable $FP \in \{0,1\}$, a set variable $V' \subset V$ which is either A or B, a state variable $\hat{\underline{u}} \in \mathcal{U}^{\mathcal{N}}$ which is the current iteration's estimate of $\underline{u}$, and a state variable $\hat{\underline{u}}' \in \mathcal{U}^{\mathcal{N}}$ which is the previous iteration's estimate of $\underline{u}$. For any $0 < \alpha < 1$ our decoder proceeds as follows:

UNIV-DEC-ITER$\big(G, \{H_j\}, \{\underline{s}_j\}\big)$

0. Initialize: $i \doteq 0, FP \doteq 0$, and $V' \doteq A$.
1. Set $\underline{\hat{u}}_j \doteq d_{ME}\left(H_j, \underline{s}_j\right)$ for each $j \in V'$.
2. **while** $i \leq \frac{\log\left(\alpha n\left(\frac{\phi-\lambda}{\Delta}\right)\right)}{\log(2-\alpha)}$ **and** $FP = 0$ **do**
3. Set $V' \doteq V \setminus V'$ and $\underline{\hat{u}}' \doteq \underline{\hat{u}}$.
4. Set $\underline{\hat{u}}_j \doteq d_{MD}\left(H_j, \underline{s}_j, \underline{\hat{u}}'_j\right)$ for each $j \in V'$.
5. Set $i \doteq i + 1$.
6. Set $FP \doteq 1_{\{\underline{\hat{u}}' = \underline{\hat{u}}\}}$.
7. **end while**
8. **return** $\underline{\hat{u}}$

Let us consider the ECOG $\big(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\big)$ setting and performing **UNIV-DEC-ITER**$\big(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\big)$. Because Δ is fixed and does not vary with n, the complexity of performing (3.68) and (3.69) is a fixed constant. Furthermore, note that because the graph is bipartite and V' is either A or B, each instance of $d_{ME}\left(H_j, \underline{s}_j\right)$ for $j \in V'$ can be done in parallel and thus the overall complexity of performing step 1 is $O(N)$. Analogously, the same holds true for $d_{MD}\left(H_j, \underline{s}_j, \underline{\hat{u}}'_j\right)$ for $j \in V'$ and so 4 also has $O(N)$ complexity.

3.6.1 Error Probability

Now we consider the error probability associated with this decoder. Before doing so the following lemma will come in handy.

Lemma 3.6.1. *[Z01, Lemma 5] Suppose $\phi \geq \frac{3}{2}\lambda_2(G)$. Let $A' \subseteq A$ be such that*

$$|A'| \leq \alpha n \left(\frac{\phi - \lambda}{\Delta}\right) \tag{3.72}$$

where $\alpha < 1$. Suppose $B' \subseteq B$ and $Y \subset E$ satisfy

1) every $e \in Y$ has an endpoint in A'.

2) every $j \in B'$ satisfies $|\Gamma(j) \cap Y| \geq \phi$.

Then $|B'| \leq \frac{1}{2-\alpha}|A'|$.

This lemma states provided Δ is large enough so that the condition $\phi \geq \frac{3}{2}\lambda_2(G)$ is met, then for any point in the while loop of the decoding algorithm we define the 'survivor' nodes in V' as

$$T_{V'}(\underline{u}, \hat{\underline{u}}) = \left\{ j \in V' \; s.t. \; w_h\left(\underline{u}_j + \hat{\underline{u}}_j\right) \geq \phi \right\} \tag{3.73}$$

then the algorithm exhibits a contraction property and will converge to $\underline{u}$ in a logarithmic number of steps (which is upper-bounded by the quantity on the right-hand side of the inequality in step 2 of **UNIV-DEC-ITER**$\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$). Thus we have

Corollary 3.6.2. *If the original number of survivor nodes*

$$T_A(\underline{u}) \triangleq \left\{ j \in A \; s.t. \; d_{ME}\left(H_j, \underline{s}_j\right) \neq \underline{u}_j \right\}$$

satisfies $|T_A(\underline{u})| \leq n\left(\frac{\phi-\lambda}{\Delta}\right)$ *then* ***UNIV-DEC-ITER***$\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ *successfully decodes* $\underline{u}$.

That the overall decoding complexity is $O(N)$ follows from using a circuit of size $O(N \log N)$ and depth $O(\log N)$, as discussed in [SS96, BZ02]. We are now in position to claim exponential error probability decay:

Theorem 3.6.3. *For sufficiently large but fixed* Δ*, the family of code constructions ECOG* $\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ *exhibits exponential error probability decay in* n *under* ***UNIV-DEC-ITER***$\left(G_{\Delta,n}, \{H_j\}, \{\underline{s}_j\}\right)$ *decoding.*

Proof. For large enough Δ the condition $\phi \geq \frac{3}{2}\lambda_2(G)$ in Lemma 3.6.1 is surely satisfied - because when the $\{h_j\}_{j \in V}$ are universally good codes, ϕ grows linearly in Δ and $\lambda_2(G) = O(\sqrt{\Delta})$. Let us define the event

$$\begin{aligned} \mathcal{E}(\underline{U}_j, j) \;\; = \;\; & \Big\{ \exists \tilde{\underline{u}} \in \mathrm{Co}\left(H_j, \underline{s}_j\right) \setminus \underline{U}_j \; s.t. \; H_j \tilde{\underline{u}} = H_j \underline{U}_j, \\ & h\left(P_{\tilde{\underline{u}}}\right) \leq h\left(P_{\underline{U}_j}\right) \Big\} \end{aligned}$$

and note that the indicator variable $1_{\{\mathcal{E}(\underline{U}_j,\beta,j)\}}$ has the property that

$$p_\Delta^1 \triangleq E\left\{1_{\{\mathcal{E}(\underline{U}_j,j)\}}\right\} = P\left(\mathcal{E}(\underline{U}_j,j)\right) \leq 2^{-\Delta E_r(R_a-\epsilon_\Delta,W)}.$$

Furthermore, the random variables $\left\{1_{\{\mathcal{E}(\underline{U}_j,j)\}}\right\}_{j\in A}$ are i.i.d. because G is bipartite and U is memoryless. So we have

$$\begin{aligned} \mathcal{N}_{\text{bad}}\left(\underline{U}\right) &\triangleq |T_A(\underline{u})| = \sum_{j\in A} 1_{\{\mathcal{E}(\underline{U}_j,\beta,j)\}}, \\ E[\mathcal{N}_{\text{bad}}\left(\underline{U}\right)] &= np_\Delta^1 \end{aligned}$$

Now by defining p_Δ^2 as

$$\begin{aligned} p_\Delta^2 &= \alpha\left(\frac{\phi}{\Delta}-\frac{\lambda}{\Delta}\right) \\ &= \alpha\left(\frac{\phi}{\Delta}-\frac{2\sqrt{\Delta-1}}{\Delta}\right) \\ &> \alpha\left(\frac{\phi}{\Delta}-\frac{2\sqrt{\Delta}}{\Delta}\right) \\ &= \alpha\left(\frac{\phi}{\Delta}-\frac{2}{\sqrt{\Delta}}\right) \end{aligned}$$

we have that since ϕ is linear in Δ, for sufficiently large Δ, $p_\Delta^2 > p_\Delta^1$ and thus

$$\begin{aligned} P_e &\leq P\left(\mathcal{N}_{\text{bad}}\left(\underline{U}\right) > np_\Delta^2\right) \\ &= 2^{n[o(n)]}P\left(\mathcal{N}_{\text{bad}}\left(\underline{U}\right) = np_\Delta^2\right) \\ &\leq 2^{-n[D\left(p_\Delta^2\|p_\Delta^1\right)-o(n)]}. \quad \square \end{aligned}$$

3.7 Universal Decoding in Multiterminal Settings

In this section we consider universal decoding for a pair of discrete i.i.d. sources $(U^1,U^2)\in\mathcal{U}=\mathcal{U}_1\times\mathcal{U}_2$ drawn according to a joint probability distribution $W\in\mathcal{P}\left(\mathcal{U}\right)$. For $k\in\{1,2\}$ we define $Q^k=|\mathcal{U}_k|=2^{t_k}$ and without loss of generality assume $\mathcal{U}^k=\{0,1,\ldots,Q^k-1\}$. For $k\in\{1,2\}$ we consider the case where a linear mapping

H^k:

$$H^k = \begin{bmatrix} -H_1^{k\prime}- \\ -H_2^{k\prime}- \\ \vdots \\ -H_{M_k}^{k\ \prime}- \end{bmatrix} : {\mathcal{U}_k}^N \to {\mathcal{U}_k}^{M_k}$$

is used over $\mathbb{F}_{2^{t_k}}$ to map $\underline{u} \in \mathcal{U}^N$ to $\underline{s} \in \mathcal{U}^{M_k}$ via

$$\underline{s}^k = H^k \underline{u} \tag{3.74}$$

where $M_k < N$. We will denote the rates as

$$R_1 = t_1 \frac{M_1}{N} \tag{3.75}$$

$$R_2 = t_2 \frac{M_2}{N} \tag{3.76}$$

The decoder knows that

$$\underline{u} \triangleq \left((u_1^1, u_1^2), (u_2^1, u_2^2) \ldots (u_N^1, u_N^2)\right) \in \mathcal{U}^N$$

must be consistent with

$$\underline{s} \triangleq (\underline{s}^1, \underline{s}^2),$$

in other words it must lie in the coset

$$\mathrm{Co}\left(H, \underline{s}\right) \triangleq \mathrm{Co}\left(H^1, H^2, \underline{s}^1, \underline{s}^2\right) \triangleq \left\{\underline{u} \;\middle|\; H^1\underline{u}^1 = \underline{s}^1,\ H^2\underline{u}^2 = \underline{s}^2\right\}, \tag{3.77}$$

and selects $\hat{\underline{u}}$ as the 'best' coset member (in a universal sense). This encompasses two settings:

a) Fixed-to-fixed length near-lossless Slepian-Wolf data compression, where $\underline{u}$ is identified as the sourceword and $\underline{s}$ is the syndrome, the output of the compression operation.

b) A multiple access channel where $\underline{x}^1 \in \mathcal{U}_1$ and $\underline{x}^2 \in \mathcal{U}_2$ are mapped to

$$\underline{y} = \left((y_1^1, y_1^2), (y_2^1, y_2^2) \ldots (y_N^1, y_N^2)\right) \in \mathcal{U}^N$$

according to

$$\left(\underline{y}^k = \underline{x}^k \oplus \underline{u}^k\right)_{k=1,2}$$

By using linear codes $\mathcal{C}^k$ for $\underline{x}^k$, and identifying the parity check matrix H^k with $\mathcal{C}^k$ as

$$\mathcal{C}^k = \left\{\underline{x} : H^k \underline{x} = \underline{0}\right\}, \tag{3.78}$$

then we have that a sufficient statistic for decoding is the pair

$$\left(H^k \underline{y}^k = H^k \underline{u}^k = \underline{s}^k\right)_{k=1,2}$$

Successfully decoding for $\underline{u}$ is equivalent to successfully decoding for the transmitted codeword $\underline{x}^k$:

$$\hat{\underline{x}^k} = \hat{\underline{u}}^k \oplus \underline{y}^k.$$

We assume that the rate pair (R_1, R_2) is achievable [SW73]:

$$\begin{aligned} R_1 &\geq H(U^1|U^2) && \text{(3.79a)} \\ R_2 &\geq H(U^2|U^1) && \text{(3.79b)} \\ R_1 + R_2 &\geq H(U^1, U^2) && \text{(3.79c)} \end{aligned}$$

As discussed in [Csi82], *linear code pairs* still suffice to attain all achievable rates and can universally attain the same error exponent $E_r(R_1, R_2, W)$ given by

$$\begin{aligned}
R_3 &\triangleq R_1 + R_2 && (3.80)\\
E_r^1(R, W) &\triangleq \min_{\mathbb{P}\in\mathcal{P}_N(\mathcal{U})} D(\mathbb{P}\|W) + \left|R - h\left(\mathbb{P}_{U^1|U^2}|\mathbb{P}_{U^2}\right)\right|^+ && (3.81)\\
E_r^2(R, W) &\triangleq \min_{\mathbb{P}\in\mathcal{P}_N(\mathcal{U})} D(\mathbb{P}\|W) + \left|R - h\left(\mathbb{P}_{U^2|U^1}|\mathbb{P}_{U^1}\right)\right|^+ && (3.82)\\
E_r^3(R, W) &\triangleq \min_{\mathbb{P}\in\mathcal{P}_N(\mathcal{U})} D(\mathbb{P}\|W) + \left|R - h(\mathbb{P})\right|^+ && (3.83)\\
E_r(R_1, R_2, W) &\triangleq \min_{i\in\{1,2,3\}} E_r^i(R_i, W) && (3.84)\\
\liminf_{N\to\infty} -\frac{1}{N}\log P_e^{\mathrm{ML}}(N) &\geq E_r(R_1, R_2, W), &&
\end{aligned}$$

under the minimum-entropy decoding rule

$$\hat{\underline{u}} \in \arg\min_{\underline{u}\in\mathrm{Co}(H,\underline{s})} h(P_{\underline{u}}). \tag{3.85}$$

Note that $E_r(R_1, R_2, W) > 0$ for all achievable rate pairs (R_1, R_2).

3.7.1 Universally Good Code Pairs

Csiszár's lemma specifying good encoders [Csi82, Sec. III] illustrates the existence of pairs of linear encoders $\left(H^k : \mathcal{U}_k^N \to \mathcal{U}^{M_k}\right)_{k=1,2}$ such that for any joint type $\mathbb{P} \in \mathcal{P}_N(\mathcal{U}^2)$ with the definitions

$$\mathcal{N}_{H^1,H^2}(\mathbb{P}) \triangleq \left|\left\{(\underline{u}\in\mathcal{U} \;\middle|\; \begin{array}{l} H^1\underline{u}^1 = H^1\tilde{\underline{u}}^1 \\ H^2\underline{u}^2 = H^2\tilde{\underline{u}}^2 \quad \text{for some } (\tilde{\underline{u}}^1, \tilde{\underline{u}}^1) \neq (\underline{u}^2, \underline{u}^2) \\ P_{\underline{u}^1,\tilde{\underline{u}}^1,\underline{u}^2,\tilde{\underline{u}}^2} = \mathbb{P} \end{array}\right\}\right|, \tag{3.86}$$

$$\mathcal{P}_N^i(\mathcal{U}^2) = \left\{\mathbb{P} = P_{\underline{u}^1,\tilde{\underline{u}}^1,\underline{u}^2,\tilde{\underline{u}}^2} \in \mathcal{P}_N(\mathcal{U}^2) \;\middle|\; \left\{\begin{array}{ll} \tilde{\underline{u}}_1 \neq \underline{u}_1, \tilde{\underline{u}}_2 = \underline{u}_2 & i = 1 \\ \tilde{\underline{u}}_1 = \underline{u}_1, \tilde{\underline{u}}_2 \neq \underline{u}_2 & i = 2 \\ \tilde{\underline{u}}_1 \neq \underline{u}_1, \tilde{\underline{u}}_2 \neq \underline{u}_2 & i = 3 \end{array}\right.\right\} \tag{3.87}$$

for $i \in \{1,2,3\}$ with $R_3 \triangleq R_1 + R_2$, *every* joint type $\mathbb{P} = P_{\underline{u}^1,\underline{\tilde{u}}^1,\underline{u}^2,\underline{\tilde{u}}^2} \in \mathcal{P}_N^i(\mathcal{U}^2)$ satisfies:

$$a) \qquad \mathcal{N}_H(\mathbb{P}) \leq 2^{-N(R_i - h(\mathbb{P}) - \delta_N)} \tag{3.88}$$

$$b) \quad \text{if } h\left(\mathbb{P}_{U^1-\tilde{U}^1,U^2-\tilde{U}^2}\right) \leq R_i - \delta_N \text{ then } \mathcal{N}_H(\mathbb{P}) = 0 \tag{3.89}$$

where $\delta_N \to 0$ as $N \to \infty$. We will denote such code pairs as **universally good**. Note that the bound (3.88) can be strengthened to:

$$\begin{aligned}
\mathcal{N}_{H^1,H^2}(\mathbb{P}) &\leq 2^{-N(R_i - h(\mathbb{P}) - \delta_N)} \\
&= 2^{N\left(h\left(\mathbb{P}_{U^1,U^2}\right) - \left(R_i - h\left(\mathbb{P}_{\tilde{U}^1,\tilde{U}^2|U^1,U^2}|\mathbb{P}_{U^1,U^2}\right) - \delta_N\right)\right)} \\
\Rightarrow \mathcal{N}_{H^1,H^2}(\mathbb{P}) &\leq 2^{N\left(h\left(\mathbb{P}_{U^1,U^2}\right) - \left|R_i - h\left(\mathbb{P}_{\tilde{U}^1,\tilde{U}^2|U^1,U^2}|\mathbb{P}_{U^1,U^2}\right) - \delta_N\right|^+\right)} \qquad (3.90) \\
&\leq \begin{cases} 2^{N\left[h\left(\mathbb{P}_{U^1,U^2}\right) - \left|R_1 - h\left(\mathbb{P}_{\tilde{U}^1|U^2}|\mathbb{P}_{\tilde{U}^2}\right) - \delta_N\right|^+\right]} & i = 1 \\ 2^{N\left[h\left(\mathbb{P}_{U^1,U^2}\right) - \left|R_2 - h\left(\mathbb{P}_{\tilde{U}^2|U^1}|\mathbb{P}_{\tilde{U}^1}\right) - \delta_N\right|^+\right]} & i = 2 \qquad (3.91) \\ 2^{N\left[h\left(\mathbb{P}_{U^1,U^2}\right) - \left|R_1+R_2 - h\left(\mathbb{P}_{\tilde{U}^1,\tilde{U}^2|U^1,U^2}|\mathbb{P}_{U^1,U^2}\right) - \delta_N\right|^+\right]} & i = 3 \end{cases}
\end{aligned}$$

where (3.90) follows because by the definition of $\mathcal{N}_{H^1,H^2}(\mathbb{P})$, $\mathcal{N}_{H^1,H^2}(\mathbb{P}) \leq |T(\mathbb{P}_{U^1,U^2})| \leq 2^{Nh\left(\mathbb{P}_{U^1,U^2}\right)}$ and (3.91) follows from (3.87).

Distance Properties of Universally Good Code Pairs

By defining the ordered pair $(0,0)$ to correspond to 0 in the usual hamming weight definition, we have that for $\underline{u} \in \mathcal{U}^N = \{\mathcal{U}^1 \times \mathcal{U}^2\}^N$,

$$w_h(\underline{u}) = \sum_{i=1}^{N} 1_{\left\{(u_i^1,u_i^2) \neq (0,0)\right\}}.$$

We define the minimium distance here as

$$d_{\min}(H^1,H^2) \triangleq \min_{\underline{u} \in \mathrm{Co}((H^1,H^2),\underline{0}) \setminus \underline{0}} w_h(\underline{u}).$$

Without much effort it follows that

$$d_{\min}(H^1, H^2) = \min_{k\in\{1,2\}} d_{\min}(H^K).$$

Note that from (3.89) that universally good code pairs will satisfy that condition for each $k \in \{1,2\}$, $d_{\min}(H^K)$ will lie on the Gilbert-Varshamov bound and thus $d_{\min}(H^1, H^2)$ will also grow linearly in N.

(β, E) Robust Code Pairs

As in the point-to-point case, we generalize the notion of error exponents for universally good code pairs. This will be useful in the error probability analysis for multitierminal LP decoding to be discussed later. Consider the events

$$\begin{aligned}
\mathcal{E}_\beta^{\text{univ}} &= \Big\{\exists\, (\underline{\tilde{u}}^1, \underline{\tilde{u}}^2) \neq (\underline{u}^1, \underline{u}^2) \;\Big|\; \\
&\qquad (\underline{\tilde{u}}^1, \underline{\tilde{u}}^2) \in \text{Co}\left((H^, H^2), (\underline{s}^1, \underline{s}^2)\right), \\
&\qquad h\left(P_{\underline{\tilde{u}}^1, \underline{\tilde{u}}^2}\right) \leq h\left(P_{\underline{u}^1, \underline{u}^2}\right) + \beta \Big\} \\
\mathcal{E}_\beta^{\text{ML}} &= \Big\{\exists\, (\underline{\tilde{u}}^1, \underline{\tilde{u}}^2) \neq (\underline{u}^1, \underline{u}^2) \;\Big|\; \\
&\qquad (\underline{\tilde{u}}^1, \underline{\tilde{u}}^2) \in \text{Co}\left((H^, H^2), (\underline{s}^1, \underline{s}^2)\right), \\
&\qquad D\left(P_{\underline{\tilde{u}}^1, \underline{\tilde{u}}^2} \| W\right) + h\left(P_{\underline{\tilde{u}}^1, \underline{\tilde{u}}^2}\right) \leq D\left(P_{\underline{u}^1, \underline{u}^2} \| W\right) + h\left(P_{\underline{u}^1, \underline{u}^2}\right) + \beta \Big\}
\end{aligned}$$

where $\beta \geq 0$. We say that the pair (H^1, H^2) is (β, E) *universally robust* if

$$-\frac{1}{N} \log P\left(\mathcal{E}_\beta^{\text{univ}}\right) \geq E.$$

and (β, E) *ML-robust* if

$$-\frac{1}{N} \log P\left(\mathcal{E}_\beta^{\text{ML}}\right) \geq E.$$

For $i \in \{1, 2, 3\}$, by defining the sets

$$\begin{aligned}
\mathcal{T}_{\beta,i}^{\text{univ}} &\triangleq \left\{\mathbb{P} \in \mathcal{P}_N^i\left(\mathcal{U}^2\right) \mid h\left(\mathbb{P}_{\tilde{U}^1, \tilde{U}^2}\right) \leq h\left(\mathbb{P}_{U^1, U^2}\right) + \beta\right\} \\
\mathcal{T}_{\beta,i}^{\text{ML}} &\triangleq \left\{\mathbb{P} \in \mathcal{P}_N^i\left(\mathcal{U}^2\right) \mid D\left(\mathbb{P}_{\tilde{U}^1, \tilde{U}^2} \| W\right) + h\left(\mathbb{P}_{\tilde{U}^1, \tilde{U}^2}\right) \leq D\left(\mathbb{P}_{U^1, U^2} \| W\right) + h\left(\mathbb{P}_{U^1, U^2}\right) + \beta\right\}
\end{aligned}$$

we have from direct application of the analysis in Sections 3.2.3 and 3.2.4 that universally good code pairs are both (β, E) *universally robust* and (β, E) *ML-robust* for any E satisfying

$$0 < E < E_r\left(R_1, R_2, \beta, W\right) \triangleq \min_{i=1,2,3} E_r^i\left(R_i - \beta, W\right).$$

Note that whenever $\{R_i - \beta\}_{i=1,2,3}$ all lie within the Slepian-Wolf region then $E_r\left(R_1, R_2, \beta, W\right) > 0$.

3.7.2 Code Pairs On Graphs

Here we consider the codes on graphs approach applied in the previous section to multiterminal settings. Here we would like to consider code constructions that employ *one graph* $G = (V, E)$ to specify the whole multiterminal system in the following way:

- Each $j \in V$ is associated with a code pair (C_j^1, C_j^2) and a syndrome pair $(\underline{s}^1, \underline{s}^2)$.
- Each edge $e \in E$ is associated with a pair of states $u_e = (u_e^1, u_e^2)$.
- Each local code pair (C_j^1, C_j^2) enforces the constraint that

$$\left(\underline{u}_j^k \in \mathrm{Co}\left(H_j^k, \underline{s}_j^k\right)\right)_{k=1,2} \Leftrightarrow \underline{u}_j \in \mathrm{Co}\left((H_j^1, H_j^2), (\underline{s}_j^1, \underline{s}_j^2)\right). \tag{3.92}$$

The coset pair $\mathrm{Co}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right)$ can be expressed as

$$\mathrm{Co}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right) = \left\{\underline{u} \;\middle|\; \underline{u}_j \in \mathrm{Co}\left((H_j^1, H_j^2), (\underline{s}_j^1, \underline{s}_j^2)\right),\ \forall\, j \in V\right\}. \tag{3.93}$$

For a particular graph $G = (V, E)$ we denote $\mathrm{COG}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ as the way in which we specify $\mathrm{Co}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right)$ in terms of (3.93).

Parity-Check Representation

In some situations, universal decoding might need to be performed when the structure of the codes is out of the control of the designer of the decoding algorithm - and

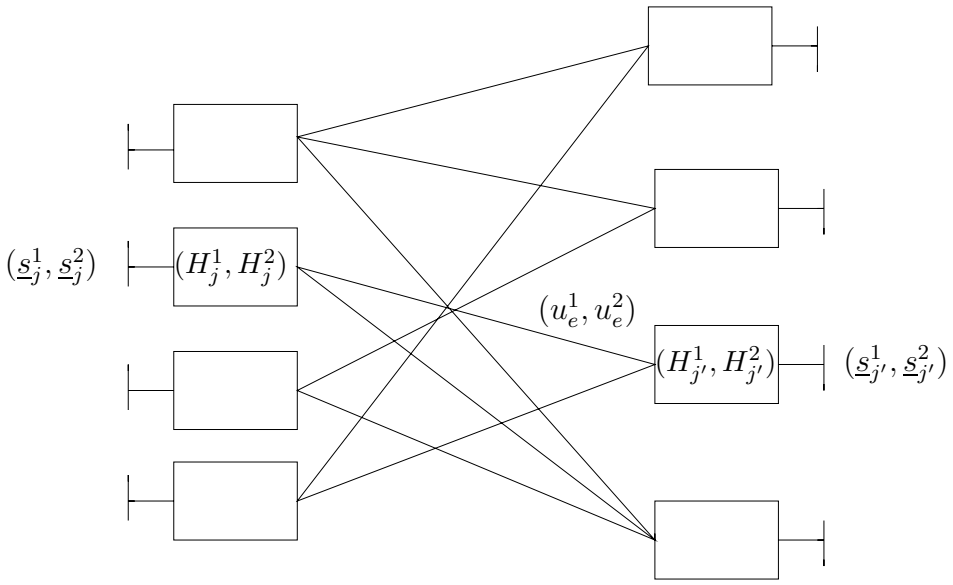


Figure 3-4: Graphical representation of $\mathrm{Co}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right)$

thus the expander graph construction with universally good constituent code pairs does not apply. One example of such a scenario is network coding for correlated sources [HMEK04], where nodes throughout a network locally perform random linear operations and the coefficients are handed to the decoder. In such a setting, we can consider without loss of generality the parity-check representation (as discussed in Section 3.4.1) for each individual source is provided, and thus we have the information $G^1, G^2, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}$. In this case we define G to be $\{G^1, G^2\}$ and still use the terminology $\mathrm{COG}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$.

Expander Code on Graph Pairs

From here on, unless specified otherwise, we assume that we are working with structured encoders with a single graph G that is an expander, i.e. $G_{\Delta,N}$. For $j \in A$, we let each code pair $(H^k)_{k=1,2}$ with rates $(R_{a,k})_{k=1,2}$ be universally robust and we assume they are achievable, i.e. they satisfy (3.79). For $j \in B$ we let each code pair

$(H^k)_{k=1,2}$ with rates $(R_{a,k})_{k=1,2}$ also be universally robust, where

$$(R_{b,k} = R_k - R_{a,k})_{k=1,2}\,.$$

We adhere to denoting such code pairs on graphs as ECOG $\big(G_{\Delta,n}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\big)$.

3.7.3 Universal Goodness of Bipartite Graph Code Pairs

Here we consider how a bipartite graph code of the form
ECOG $\big(G_{\Delta,n}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\big)$ performs under minimum-entropy decoding, as Δ grows. Let $N = n\Delta$. For $i \in \{1,2,3\}$, let $\mathbb{P} \in \mathcal{P}_N^i(\mathcal{U}^2)$ correspond to the joint type of any length-N $4-tuple$ $(\underline{u}^1, \tilde{\underline{u}}^1, \underline{u}^2, \tilde{\underline{u}}^2)$ satisfying $\{H^k\underline{u}^k = H^k\tilde{\underline{u}}^k\}_{k=1,2}$. Define $\mathbb{P}^j \in \mathcal{P}_\Delta^i(\mathcal{U}^2)$ to correspond to the joint type of any *local* length-Δ 4-tuple $(\underline{u}_j^1, \tilde{\underline{u}}_j^1, \underline{u}_j^2, \tilde{\underline{u}}_j^2)$ satisfying $\{H_j^k\underline{u}^k = H_j^k\tilde{\underline{u}}^k\}_{k=1,2}$. Then by defininig $R_{a,3} = R_{a,1} + R_{a,2}$ we have:

$$\begin{aligned}
\mathcal{N}_{H^1,H^2}(\mathbb{P}) &\leq \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j = \mathbb{P}} \prod_{j\in A} \mathcal{N}_{H_j^1,H_j^2}\left(\mathbb{P}^j\right) \\
&\leq \sum_{\frac{1}{n}\sum_{j\in A}\mathbb{P}^j = \mathbb{P}} \prod_{j\in A} 2^{-\Delta\left(R_{a,i} - h\left(\mathbb{P}^j\right) - \delta_\Delta\right)} \qquad (3.94) \\
&\leq 2^{-N\left(R_{a,i} - h(\mathbb{P}) - \epsilon'_\Delta\right)} \qquad (3.95)
\end{aligned}$$

where $\epsilon'_\Delta \to 0$ as $\Delta \to \infty$, (3.94) follows from (3.88), and (3.95) follows from Lemma 3.4.3. Thus it follows that the pair (H^1, H^2) becomes universally good for large Δ, when thought of having rate $R'_i = R_{a,i} - \epsilon'_\Delta$.

Encoding

Encoding for the compression situation is done by applying applying $\{\underline{s}_j^k = H_j^k\underline{u}_j^k\}_{k=1,2}$ for all $j \in V$. For the channel coding scenario, the encoding done is the same as discussed in [SS96, BZ02].

3.7.4 Decoding Methods

LP Decoding Methods

The LP decoding algorithm for the multiterminal setting is almost completely a direct application of Section 3.5. We consider a codes on graphs representation We first construct indicator variables $I_{a,e} \in \{0,1\}$, for $a \in \mathcal{U} = (\mathcal{U}_1 \times \mathcal{U}_2), e \in V$, such that $I_{a,e} = 1_{\{u_e = a\}}$. Thus $I_{a,e}$ specifies $\underline{u} \in \mathcal{U}^n$ as

$$\underline{u}^1 = \mu^1(I) = \sum_{(a_1,a_2)\in\mathcal{U}} a_1 I_{(a_1,a_2),e}$$
$$\underline{u}^2 = \mu^2(I) = \sum_{(a_1,a_2)\in\mathcal{U}} a_2 I_{(a_1,a_2),e}.$$

When $\mathrm{COG}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ corresponds to G which is a single graph, the coset pair $\mathrm{Co}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right)$ is well-defined from (3.77) and polytope $\tilde{\mathcal{B}}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ is well-defined from direct application of the definitions in Section 3.5.2.

When $\mathrm{COG}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ corresponds to $G = \{G^1, G^2\}$, then the polytope $\tilde{\mathcal{B}}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ is easily defined as

$$\tilde{\mathcal{B}}\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right) = \Big\{ I \quad | \quad \mu^1(I) \in \tilde{\mathcal{B}}\left(G, \{H_j^1\}, \{\underline{s}_j^1\}\right), \\ \mu^2(I) \in \tilde{\mathcal{B}}\left(G, \{H_j^2\}, \{\underline{s}_j^2\}\right) \Big\}$$

where $\tilde{\mathcal{B}}\left(G, \{H_j\}, \{\underline{s}_j\}\right)$ is discussed in Section 3.5.2 and given by (3.51a)- (3.51c).

The formulation of the LP relaxation $\mathbf{LP\text{-}PRIMAL}\big(G, \{\gamma_a\}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\big)$ for this setting follows directly from the definition of $\mathrm{Co}\left((H_j^1, H_j^2), (\underline{s}_j^1, \underline{s}_j^2)\right)$. We can thus consider the following universal decoding algorithm

UNIV-DEC-LP$\left(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$

0. Set $\left(\underline{u}^{1^*}, \underline{u}^{2^*}\right) \doteq \left(\tilde{\underline{u}}^1, \tilde{\underline{u}}^2\right)$ for any $\left(\tilde{\underline{u}}^1, \tilde{\underline{u}}^2\right)$ satisfying $P_{\tilde{\underline{u}}^1, \tilde{\underline{u}}^2} = \mathbb{U}$.

1. **For** each $\mathbb{P} \in \mathcal{P}_\Delta(\mathcal{U})$ **do**

2. Execute **LP-PRIMAL**$\left(G, \{-\log \mathbb{P}(a)\}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$.

3. **If** the optimal solution I^* to **LP-PRIMAL**$\left(G, \{-\log \mathbb{P}(a)\}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ is integral and $h\left(P_{\mu^1(I^*), \mu^1(I^*)}\right) \leq h\left(P_{\underline{u}^{1^*}, \underline{u}^{2^*}}\right)$ **then** set $\left(\underline{u}^{1^*}, \underline{u}^{2^*}\right) \doteq (\mu^1(I^*), \mu^1(I^*))$.

4. **end for**

5. **return** $\left(\underline{u}^{1^*}, \underline{u}^{2^*}\right)$

By defining

$$\begin{aligned}
0 < \delta_A &= \min_{j \in A} \frac{d_{\min}(H_j^1, H_j^2)}{\Delta} \\
0 < \delta_B &= \min_{j \in B} \frac{d_{\min}(H_j^1, H_j^2)}{\Delta} \\
p_\Delta^1 &= 2^{-\Delta E_r(R_{a,1}, R_{a,2}, \beta, W)} \\
p_\Delta^2 &= \frac{1}{\Delta + 1}\left(\frac{\delta_B}{1 + \delta_B/\delta_A + \log(\Delta)/\beta} - \frac{2}{\sqrt{\Delta}}\right)
\end{aligned}$$

where $\beta > 0$ is such that $E_r(R_{a,1}, R_{a,2}, \beta, W) > 0$, we have from Theorem 3.5.1 that for sufficiently large but fixed Δ, the error probability P_e of **UNIV-DEC-LP**$\left(G_{\Delta,n}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\right)$ satisfies

$$P_e \leq 2^{-n[D\left(p_\Delta^1 \| p_\Delta^2\right) - o(n)]}.$$

Iterative Decoding Methods

The iterative expander decoding algorithm for the multiterminal setting is almost completely a direct application of Section 3.6. Define

$$\begin{aligned}
d_{ME}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right) &= \arg \min_{\tilde{\underline{u}} \in \mathrm{Co}((H^1, H^2), (\underline{s}^1, \underline{s}^2))} h\left(P_{\tilde{\underline{u}}}\right) \\
d_{MD}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2), \underline{u}'\right) &= \arg \min_{\tilde{\underline{u}} \in \mathrm{Co}((H^1, H^2), (\underline{s}^1, \underline{s}^2))} w_h\left(\tilde{\underline{u}} + \underline{u}'\right) \\
\phi = \phi(\{H_j\}) &= \min_{j \in V} \frac{1}{2} d_{\min}(H_j^1, H_j^2) \\
\lambda &= \lambda_2(G)
\end{aligned}$$

and consider for any $0 < \alpha < 1$ the algorithm:

UNIV-DEC-ITER$\big(G, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\big)$

0. Initialize: $i \doteq 0, FP \doteq 0$, and $V' \doteq A$.
1. Set $\underline{\hat{u}}_j \doteq d_{ME}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2)\right)$ for each $j \in V'$.
2. **while** $i \leq \frac{\log\left(\alpha n\left(\frac{\phi-\lambda}{\Delta}\right)\right)}{\log(2-\alpha)}$ **and** $FP = 0$ **do**
3. Set $V' \doteq V \setminus V'$ and $\underline{\hat{u}}' \doteq \underline{\hat{u}}$.
4. Set $\underline{\hat{u}}_j \doteq d_{MD}\left((H^1, H^2), (\underline{s}^1, \underline{s}^2), \underline{\hat{u}}_j'\right)$ for each $j \in V'$.
5. Set $i \doteq i + 1$.
6. Set $FP \doteq 1_{\{\underline{\hat{u}}' = \underline{\hat{u}}\}}$.
7. **end while**
8. **return** $\underline{\hat{u}}$

By defining

$$\begin{aligned} p_\Delta^1 &= 2^{-\Delta E_r(R_{a,1}, R_{a,2}, W)} \\ p_\Delta^2 &= \alpha\left(\frac{\phi}{\Delta} - \frac{2}{\sqrt{\Delta}}\right) \end{aligned}$$

we have from Corollary 3.6.2 that for sufficiently large Δ the error probability P_e of **UNIV-DEC-ITER**$\big(G_{\Delta,n}, \{(H_j^1, H_j^2)\}, \{(\underline{s}_j^1, \underline{s}_j^2)\}\big)$ satisfies

$$P_e \leq 2^{-n[D\left(p_\Delta^1 \| p_\Delta^2\right) - o(n)]}.$$

Chapter 4

Reasons and Practical Methods for Coding on the Deterministic Broadcast Channel

Background: Several notions of duality between the Slepian-Wolf, multiple access, and deterministic broadcast channel have been previously established. Rate-splitting techniques to pipeline the decoding process for attaining any set of achievable rates have been previously established for Slepian-Wolf and multiple access.

Our contribution: Here we motivate the consideration of deterministic broadcast channels with wireless interference management examples. We next extend this 'triangle of duality' by illustrating a rate-splitting approach for pipelining the encoding process to attaining any set of achievable rates for deterministic broadcast. We also illustrate that a practical 'enumerative source coding' approach can apply to first-stage vertex coding for any deterministic broadcast problem. We also show that for degraded deterministic broadcast channels, this enumerative approach can be extended for encoding with side information and decoding to provide a complete solution for all achievable rates. For two-receiver deterministic broadcast settings where encoding for one receiver imposes constraints on the alphabet of other receivers, we characterize the optimal input probability distribution and illustrate that for vertex rates on the boundary of the capacity region, the second stage of pipelined encoding has an 'erasure channel' interpretation. From this interpretation we dualize the low-complexity erasure correction 'LT codes' and develop capacity-achieving codes for our problem. Such settings include our motivating wireless examples as well as the Blackwell channel - the simplest non-trivial deterministic broadcast channel.

The structural similarity between the capacity regions and random coding achievable rate strategies for the Slepian-Wolf problem [SW73] and the deterministic broadcast channel problem [Mar77, Pin78] has been illustrated in [Cov98, sec. III] (see also Figure 4-1). Duality connections between the capacity regions of the discrete multiple access channel [Lia72, Ahl71] and the deterministic broadcast have been recently discussed in [JVG03]. Also, the error probability analysis for jointly typical decoding in the discrete multiple access channel is dual to that of the Slepian-Wolf problem [CT91, p. 416-418]. Indeed, a more precise notion of duality between these two problems is mentioned in [Csi82]. Different aspects of these dualities have also been explored in [DFK04, SCX04]. This suggests the existence of a strong relationship amongst these three canonical information theory problems.

Practically speaking, the decoding process is one of the biggest challenges in achieving rates near the boundary of the achievable region in the Slepian-Wolf problem and the discrete multiple access channel, as a single decoder must jointly decode messages from multiple senders. Analogously, the encoding process represents a formidable challenge for the deterministic broadcast channel, as one encoder must jointly incorporate multiple messages into a single channel codeword (see Figure 4-2).

Recently, splitting techniques have been discussed as a way to significantly reduce the complexity of decoding in the discrete multiple access [GRUW01] and Slepian-Wolf problems ([RU97], Chapter 2), for *any* achievable rate - not just a 'vertex' or 'corner point'. Paralleling the source coding example discussed at length in Section 2.2, this technique decomposes the problem of jointly decoding M users into a set of $2M-1$ pipelined single-user channel decoding with side information.

Motivated by the strong relationship between the Slepian-Wolf, deterministic broadcast, and discrete multiple access channel problems along with the splitting techniques for Slepian-Wolf and multiple access, we consider a rate-splitting technique for deterministic broadcast. The goal here is to take an arbitrary point in the M-receiver deterministic broadcast channel achievable rate region and transform it, by rate-splitting each source at most once, to a *vertex* of another $2M-1$-receiver deterministic broadcast channel achievable region.

Deterministic broadcast is of interest in part because it gives a very simple model of interference in wireless scenarios. We provide a couple of multiple antenna wireless broadcast examples to illustrate this. In this chapter we discuss practical strategies to attain all achievable rates. We first consider techniques that are applicable to any instance of an M-receiver deterministic broadcast problem. These include rate-splitting, and an 'Enumerative Source Coding' [Cov73] approach for first-stage vertex encoding. Then we discuss specific examples of two-receiver problems (including the wireless examples as well as the Blackwell channel) that relate to coding for the erasure channel. We show how in these settings, dualizing the 'Luby Transform' [Lub02] code construction and encoding/decoding algorithms applies to second-stage vertex encoding for these problems. Together the general-purpose first-stage vertex encoding approach and the second-stage 'Luby Transform' dualization approach give low-complexity, capacity-achieving deterministic broadcast codes for these specific examples. These are to our knowledge the first such codes for any deterministic broadcast channel in the literature.

4.1 Background on the Deterministic Broadcast Channel

The deterministic broadcast channel, illustrated in Figure 4-2, has one sender and M receivers. Let $m_j \in \left\{1, \ldots, 2^{nR_j}\right\}$ denote the message for receiver $j \in \{1, \ldots, M\}$. The sender combines the M independent messages $\{m_j\}_{j=1}^{M}$ into a single length-n channel input $\underline{x} = (x_1, \ldots, x_n)' \in \mathcal{X}^n$. At receiver j each symbol $y_i^j \in \mathcal{Y}_j$ is a deterministic function of x_i, i.e. $y_i^j = f_j(x_i)$. The jth decoder attempts to reconstruct m_j, i.e. $\hat{m}_j = d_j(\underline{y}^j)$. A memoryless probability distribution $P(x)$ on X, combined with $f_1, \ldots, f_M$, induces a memoryless joint distribution $P(y^1, \ldots, y^M)$ on $\{Y^1, \ldots, Y^M\}$. For a fixed memoryless $P(x)$, the set of all achievable rates is given by [Pin78, Mar77]

$$\mathcal{R}\left[P(x); f_1, \ldots, f_M\right] = \left\{ \underline{R} \in \mathbb{R}_+^M \;\middle|\; \sum_{i \in S} R_i < H\left(Y(S)\right) \;\forall\, S \subseteq \{1 \ldots M\} \right\} \quad (4.1)$$

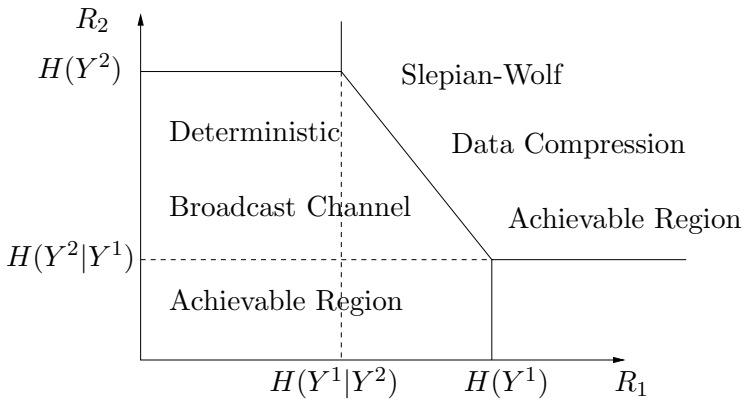


Figure 4-1: The dual relationship between the deterministic broadcast and Slepian-Wolf problems

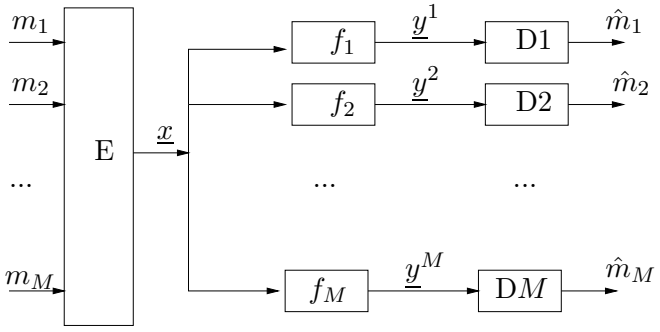


Figure 4-2: Basic Model for the Deterministic Broadcast Channel.

where $Y(S) = \{Y^j, j \in S\}$. The similarity between $\mathcal{R}[P(x); f_1, \ldots, f_M]$ and the Slepian-Wolf achievable rate region is illustrated in Figure 4-1. The full capacity region of the deterministic broadcast channel is given by

$$\mathcal{R}[f_1, \ldots, f_M] = \text{cl}\left[\text{CH}\left(\bigcup_{P(x)\in\mathcal{P}(\mathcal{X})} \mathcal{R}[P(x); f_1, \ldots, f_M]\right)\right]$$

where *cl* denotes closure and CH denotes convex hull.

4.1.1 Binning as an Achievable Strategy

As discussed in [Cov98, sec. III], the achievable rate strategies for the Slepian-Wolf and deterministic broadcast problems allow for one problem's encoder to mimic the

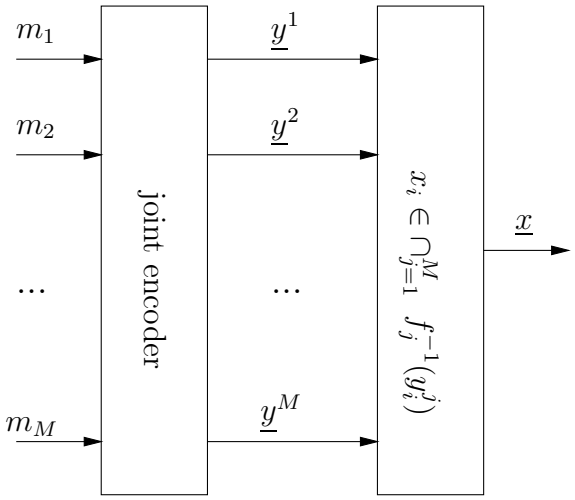


Figure 4-3: Joint encoding with binning for the deterministic broadcast channel.

other's decoder, and vice versa. We here describe the code design used in the achievability proof for the deterministic broadcast capacity region. Given a fixed distribution $P(x)$ on the channel input X, let $P(y^1, \ldots, y^M)$ be the distribution that results from setting $\{Y^j = f_j(X)\}_{j=1}^M$. We place each $\underline{Y}^j \in \mathcal{Y}_j^n$ uniformly and randomly into one of 2^{nR_j} bins and use $d_j(\underline{Y}^j)$ to denote the resulting bin index. The encoder first maps indices $(m_1, \ldots, m_M)$ to any tuple $(\underline{Y}^1, \ldots, \underline{Y}^M)$ such that $(\underline{Y}^1, \ldots, \underline{Y}^M)$ are jointly typical and $d_j(\underline{Y}^j) = m_j$ for each j. The encoder then maps $(\underline{Y}^1, \ldots, \underline{Y}^M)$ to a corresponding channel input by chosing $x_i \in \cap_{j=1}^M f_j^{-1}(y_i^j)$ for each $i \in \{1, \ldots, n\}$. The jth decoder receives $\underline{Y}^j$ and decodes to message $d_j(\underline{Y}^j)$. Since there is no noise in the system, communication fails if and only if the encoder fails. Figure 4-3 illustrates this encoding strategy.

4.1.2 Vertices: Successive Encoding

If we consider an encoding strategy with a memoryless probability distribution $P(x)$, then the set of achievable rates $\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]$ has 'vertices' or 'corner points' associated with expanding $H(Y^1, \ldots, Y^M)$ into M terms by successive applications of the chain rule for entropy and assigning to each rate the unique corresponding term in the expansion. Transmitting at such rates allows for the joint search over all users'

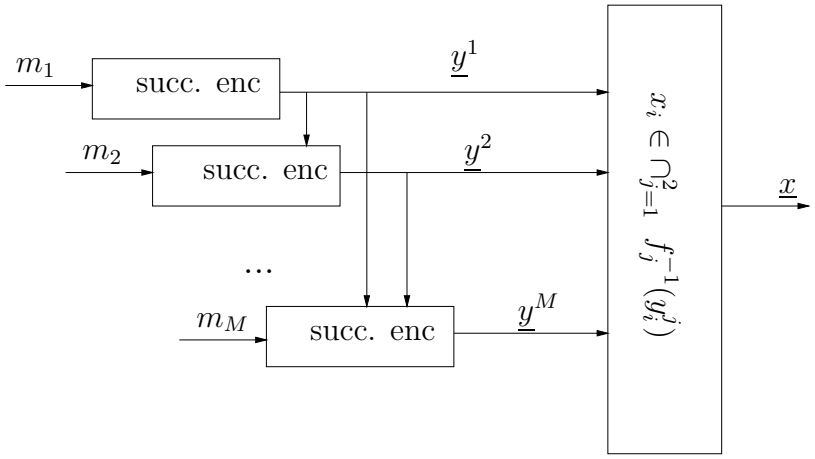


Figure 4-4: Pipelined encoder for communicating at a vertex rate for the deterministic broadcast channel.

bins to be done successively. For example, consider communicating at the vertex rate $(R_1, R_2, \ldots, R_M) = \big(H(Y^1), H(Y^2|Y^1), \ldots, H(Y^M|Y^1 \ldots Y^{M-1})\big)$:

- Encoding message m_1 at rate $R_1 = H(Y^1)$ can be done by searching in the bin of message m_1 for a typical $\underline{y}^1$ sequence. There are 2^{nR_1} such bins, one for each possible value of m_1, and there are asymptotically $2^{nH(Y^1)}$ typical $\underline{y}^1$ sequences.

- Consider any $j \in \{2, \ldots, M\}$. After successful choice of channel outputs $\underline{y}^1, \ldots, \underline{y}^{j-1}$ to describe $m_1, \ldots, m_{j-1}$, encoding message m_j at rate $R_j = H(Y^j|Y^1 \ldots Y^{j-1})$ can be done by searching in bin m_j for a sequence $\underline{y}^j$ that allows for $(\underline{y}^1, \ldots, \underline{y}^j)$ to be jointly typical. There are 2^{nR_j} such bins, and there are asymptotically $2^{nH(Y^j|Y^1 \ldots Y^{j-1})}$ sequences $\underline{y}^j$ that allow for $(\underline{y}^1, \ldots, \underline{y}^j)$ to be jointly typical.

Figure 4-4 illustrates the successive encoding mechanism.

4.1.3 Practical Challenges for the Deterministic Broadcast Channel

Deterministic broadcast channel code design presents a variety of challenges that differ from other practically solvable multi-terminal binning problems (such as Slepian-Wolf

[SW73]).

Shaping

First, the optimal input to the channel $P(x)$ need not be uniform. Thus, a shaping code is needed to map uniform message bits to channel input symbols with often non-uniform probabilities. This operation is in some sense the dual of lossless compression, which takes non-equiprobable source symbols and maps them to uniform compressed bits. Gallager [Gal68, pp. 208-209] discusses one encoding approach using linear codes, but he also notes that the decoding process is prohibitively complex.

Binning

As discussed in Section 4.1.1, the binning strategy required for code design maps the messages $\{m_j\}_{j=1}^{M}$ to channel outputs $\left\{\underline{y}^j\right\}_{j=1}^{M}$ for which each $\underline{y}^j$ falls in bin m_j and $\left\{\underline{y}^j\right\}_{j=1}^{M}$ is jointly typical. Unfortunately, for all achievable rates, with high probability there will be *exponentially many* such jointly typical codewords. The possibility of finding multiple jointly typical solutions represents an important challenge in designing low complexity binning codes for deterministic broadcast channels. In particular, traditional iterative coding algorithms may never converge if the solution is not unique.

4.2 Wireless Motivations for Interest in Practical Channel Codes for Deterministic Broadcast

One major issue in wireless networks is interference. At a high level, we understand how to deal with noise relatively well, but our understanding of the interaction between multiple transmitters and receivers is still somewhat limited. As a result, most network designs attempt to suppress these interactions by making the different users orthogonal or performing successive cancellation. When properly managed, these interactions can actually result in a net benefit to the system.

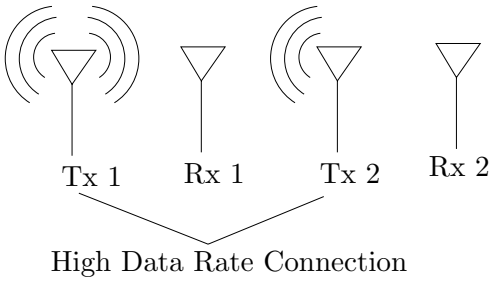


Figure 4-5: A wireless downlink communication scenario. One user, Rx 1, receives signals from both transmitters Tx 1 and Tx 2. The the second user, Rx 2, receives only the signal from the second transmitter. If the two transmitters are connected by a high data rate link such as a land-line for terrestrial transmitters or a laser communication link for satellites, transmitter cooperation can increase the communication rate.

Consider the wireless communication scenario shown in Figure 4-5 where two users both receive signals from a pair of cooperating transmitters. Intuitively, the interference to Rx 2 from Tx 1 is negligible since the signal from Tx 1 experiences much higher path loss than Tx 2. In contrast, Rx 1 receives potentially interfering signals from both Tx 1 and Tx 2. Similar situations can arise in wireless networks employing multi-hop transmission as illustrated in Figure 4-6. How can we model such scenarios to understand the effect of such interference and develop efficient communication schemes? One approach is to consider a Gaussian broadcast channel model where the transmitter has perfect channel side information (CSI) describing the propagation parameters. For such models, the capacity region [WSS04] is obtained using Costa's idea of writing on dirty paper [Cos83].

Unfortunately, this approach requires perfect knowledge of the channel. Thus it is not clear to what extent the so-called dirty paper coding ideas apply to non-coherent communication or to relay networks like the one in Figure 4-6, which employ distributed transmitter cooperation. Furthermore, even when perfect channel state information is available, no practical coding scheme is known that achieves capacity. Specifically, to our knowledge, the best known dirty paper coding systems are at least a decibel away from capacity [EtB05, SLSX05]. At low signal-to-noise ratios, which are common in certain types of wireless networks, a decibel may correspond to a

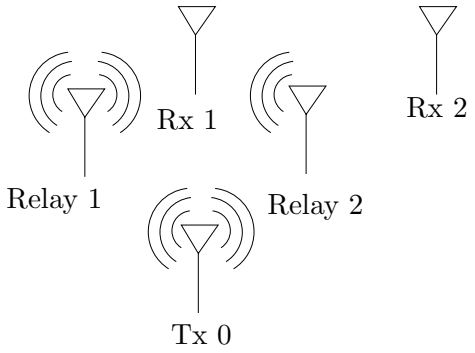


Figure 4-6: A wireless relay communication scenario. A single transmitter, Tx 0, sends a message intended for two receivers Rx 1 and Rx 2 via multi-hop transmission. The two intermediate relay nodes each decode the message, re-encode, and transmit to the ultimate destination. Rx 1 receives potentially interfering signals from both relays, while Rx 2 sees only the signal from Relay 2.

large fraction of the transmitted power or rate. For multi-antenna channels, the gap to capacity may be larger.

The existence of this gap to capacity may seem surprising in light of the spectacular success of turbo codes [BGT93], low density parity check codes (LDPCs) [Gal62], and other codes on graphs in approaching capacity for single user channels. Intuitively, the gap to capacity for dirty paper coding is caused by the lack of efficient codes for the shaping/binning required in the random coding arguments for broadcast channels.

We approach the wireless network communication problem in Figures 4-5 and 4-6 from a different perspective. Since dealing with noise via coding is fairly well understood, we focus purely on interference issues by considering a deterministic broadcast channel. This allows us to develop clearer insights into the code design in systems with interference.

We now consider a deterministic broadcast channel model based on Figures 4-5 and 4-6 with a pair of binary input symbols $X = [X^1, X^2]'$ and two outputs Y^1, Y^2. Intuitively, X^i corresponds to the channel input for Tx i or Relay i in Figures 4-5 and 4-6. If the two channel inputs are the same, then they are perfectly received at both receivers. If they differ, then receiver 2 receives X^2 correctly while receiver 1 suffers destructive interference and can't determine X^1. We model this scenario via

the deterministic broadcast channel in (4.2):

$$\text{if } X^1 = X^2 \quad \text{then} \quad Y^1 = X^1 \text{ and } Y^2 = X^2 \tag{4.2a}$$

$$\text{if } X^1 \neq X^2 \quad \text{then} \quad Y^1 = * \text{ and } Y^2 = X^2 \tag{4.2b}$$

where $*$ means 'erasure'. Thus we have the following input-output relationship:

(X^1, X^2)	Y^1	Y^2
$(-1,-1)$	-1	-1
$(-1,1)$	$*$	1
$(1,-1)$	$*$	-1
$(1,1)$	1	1

The channel in (4.2) can model a variety of physical scenarios. Perhaps the simplest is binary phase shift keying (BPSK) with additive combining. For this model, the channel inputs are $X^i = \pm 1$ with $Y^2 = X^2$ for the receiver without interference and additive interference corresponding to $Y^1 = X^1 + X^2$ for the other receiver. Thus the $*$ output in (4.2b) represents the case where $X^1 = -X^2$ resulting in a received signal of $Y^1 = 0$. Equation (4.2) can also represent non-coherent modulation such as frequency shift keying (FSK). In an FSK model, each transmitter sends either on frequency f_0 or f_1 corresponding to $X^i = -1$ or 1. If the two transmitted signals both equal t, then both receivers see a signal on frequency f_t and decode correctly. If the two transmitted signals are opposite, the first receiver sees no interference and decodes correctly while the second receiver observes signals on both frequencies, corresponding to an erasure.

4.3 Complexity-Reduction Techniques for Arbitrary Deterministic Broadcast Channels

Here we discuss complexity reduction techniques that apply to an arbitrary instance of a deterministic broadcast problem.

4.3.1 Rate-Splitting

We now illustrate that for a fixed memoryless distribution $P(x)$, a rate-splitting approach can be applied so that *any* rate in $\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]$ can be transformed to a vertex in another $\mathcal{R}\left[P(x); \tilde{f}_{1a}, \tilde{f}_{1b}, \ldots, \tilde{f}_{(M-1)a}, \tilde{f}_{(M-1)b}, f_M\right]$, for some appropriately defined functions $\left\{\tilde{f}_{1a}, \tilde{f}_{1b}, \ldots, \tilde{f}_{(M-1)a}, \tilde{f}_{(M-1)b}\right\}$. The *dominant face* is given by

$$\mathcal{D}\left[\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]\right] = \left\{ R \in \mathcal{R}\left[P(x); f_1, \ldots, f_M\right] \,\middle|\, \sum_{i=1}^{M} R_i = H\left(Y^1, \ldots Y^M\right) \right\}.$$

Since any point in $\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]$ is dominated (with respect to the standard partial order on $\mathbb{R}_+^M$) by a point in $\mathcal{D}\left[\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]\right]$, we restrict our attention to rates lying on $\mathcal{D}\left[\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]\right]$.

We now discuss the two-receiver problem where $m_1 \in \{1, \ldots, 2^{nR_1}\}$, $m_2 \in \{1, \ldots, 2^{nR_2}\}$ and $(R_1, R_2) \in \mathcal{D}\left[\mathcal{R}\left[P(x); f_1, f_2\right]\right]$ but (R_1, R_2) is not a vertex. Consider the probability distribution $P(y^1, y^2)$ induced by $P(x)$ and f_1, f_2. The splitting operation constructs $Y_i^{1a} = f_{1a}(Y_i^1), Y_i^{1b} = f_{1b}(Y_i^1)$ such that (Y_i^{1a}, Y_i^{1b}) and Y_i^1 form a bijection. We again use the splitting algorithm from Section 2.2. Assuming, without loss of generality, that $\mathcal{Y}_1 = \{0, 1, \ldots, Q-1\}$, the splitting mechanism is constructed as follows:

$$f_{1a}(y^1) = \min(\pi(y^1), T) \tag{4.3a}$$

$$f_{1b}(y^1) = \max(\pi(y^1), T) - T \tag{4.3b}$$

$$f_1(y^{1a}, y^{1b}) = \pi^{-1}\left(y^{1a} + y^{1b}\right) \tag{4.3c}$$

where $T \in \mathcal{Y}_1$ and π is a permutation of $\mathcal{Y}_1$. Such a splitting mechanism induces a memoryless distribution $P(y^{1a}, y^{1b}, y^2)$ where $H(Y^{1a}, Y^{1b}, Y^2) = H(Y^1, Y^2)$. As discussed in Section 2.2.2, splitting according to (4.3) allows for *any* rate $R \in$

$\mathcal{D}\left[\mathcal{R}\left[P(x); f_1, f_2\right]\right]$ to satisfy

$$\begin{aligned} R_{1a} &= H\left(Y^{1a}\right) && \text{(4.4a)} \\ R_2 &= H\left(Y^2|Y^{1a}\right) && \text{(4.4b)} \\ R_{1b} &= H\left(Y^{1b}|Y^{1a}, Y^2\right) && \text{(4.4c)} \\ R_1 &= R_{1a} + R_{1b}. && \text{(4.4d)} \end{aligned}$$

Now let us assume the splitting operation has been performed so that (4.4) holds. The encoder takes the message $m_1 \in \{1, \ldots, 2^{nR_1}\}$ and represents it as a pair of messages

$$m_1 \in \{1, \ldots, 2^{nR_1}\} \tag{4.5a}$$

$$\Leftrightarrow$$

$$(m_{1a}, m_{2a}) \in \left(\{1, \ldots, 2^{nR_{1a}}\}, \{1, \ldots, 2^{nR_{1b}}\}\right). \tag{4.5b}$$

It partitions all possible $\underline{y}^{1a}$ sequences into $2^{nR_{1a}}$ bins, all possible $\underline{y}^{1b}$ sequences into $2^{nR_{1b}}$ bins, and all $\underline{y}^2$ sequences into 2^{nR_2} bins. Encoding is done as if to construct a jointly typical $(\underline{y}^{1a}, \underline{y}^{1b}, \underline{y}^2)$. Note that although (R_1, R_2) is not a vertex in $\mathcal{R}\left[P(x); f_1, f_2\right]$, (R_{1a}, R_{1b}, R_2) is a vertex in $\mathcal{R}\left[P(x); f_{1a} \circ f_1, f_{1b} \circ f_1, f_2\right]$; thus the encoding strategy described in Section 4.1.2 suffices. See Figure 4-7. The decoder for receiver 2 observes $\underline{y}^2$ and specifies its bin number m_2. Receiver 1's decoder observes $\underline{y}^1$, performs the splitting operation to construct $(\underline{y}^{1a}, \underline{y}^{1b})$, and specifies the bin numbers (m_{1a}, m_{1b}). Finally, the message pair (m_{1a}, m_{2a}) is combined to form the message m_1 by simply reversing the operation (4.5). See Figure 4-8. This approach generalizes to M users.

Theorem 4.3.1. *Any achievable rate for an arbitrary M-receiver deterministic broadcast channel can be attained via rate-splitting to give an equivalent $(2M-1)$-receiver deterministic broadcast channel, where each user is split at most once and successive encoding suffices.*

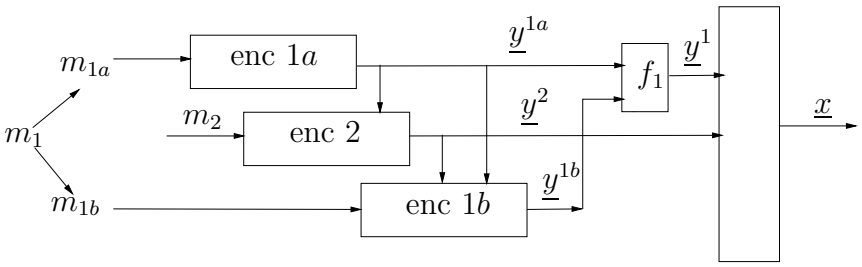


Figure 4-7: Rate-splitting based encoding.

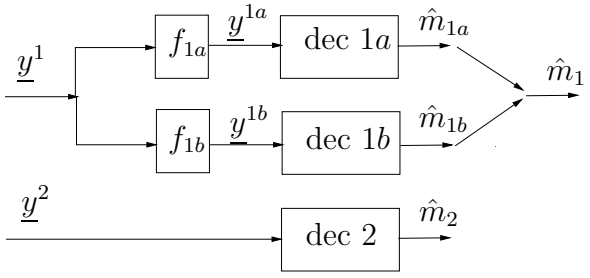


Figure 4-8: Rate-splitting based decoding.

Since $\mathcal{D}\left[\mathcal{R}\left[P(x); f_1, \ldots, f_M\right]\right]$ has precisely the same characterization as the dominant face of the M-source Slepian-Wolf achievable rate region, the proof of theorem 4.3.1 follows from the analogous source-splitting results for the Slepian-Wolf problem in Chapter 2.

4.3.2 Practical First-Stage Vertex Pipelined Encoding for the General DBC

We now consider communicating at a vertex rate and attempt to construct the first sequence $\underline{y}^j$ in the pipeline, which has a rate given by $R_j = H\left(Y^j\right)$. We use Cover's enumerative source coding [Cov73], which relies on the method of types [Csi98]. De-

fine:

$$\begin{aligned}
\mathcal{M} &= \{0, 1, \ldots 2^{nR} - 1\} \\
P_{\underline{y}} &= \left(\left\{ \frac{1}{n} \sum_{i=1}^{n} 1_{y_i = a} \right\}_{a \in \mathcal{Y}} \right) \text{ for } \underline{y} \in \mathcal{Y}^n \\
\mathcal{P}_n(\mathcal{Y}) &= \left\{ \mathbb{P} \in \mathcal{P}(\mathcal{Y}) : \mathbb{P} = P_{\underline{y}} \text{ for some } \underline{y} \in \mathcal{Y}^n \right\} \\
T(\mathbb{P}) &= \left\{ \underline{y} \in \mathcal{Y}^n : P_{\underline{y}} = \mathbb{P} \right\}
\end{aligned}$$

For a target distribution $P(y^j)$, we select type $\mathbb{P}^{*n}_{Y^j}$ such that

$$\mathbb{P}^{*n}_{Y^j} \in \arg \min_{\mathbb{P} \in \mathcal{P}_n(\mathcal{Y}_|)} \left| \mathbb{P} - P(y^j) \right|_1 . \tag{4.6}$$

Because $\mathbb{Q}$ is dense in $\mathbb{R}$, $\mathbb{P}^{*n}_{Y^j} \to P(y^j)$. Moreover, since

$$\begin{aligned}
\left| T\left(\mathbb{P}^{*n}_{Y^j}\right) \right| &= \binom{n}{n\mathbb{P}^{*n}_{Y^j}(0) \ \ldots \ n\mathbb{P}^{*n}_{Y^j}(|\mathcal{Y}_1| - 1)} \\
&= 2^{n\left[h\left(\mathbb{P}^{*n}_{Y^j}\right) - o(n)\right]}
\end{aligned} \tag{4.7}$$

where $o(n) \to 0$, we have that

$$\frac{1}{n} \log \left| T\left(\mathbb{P}^{*n}_{Y^j}\right) \right| \to H\left(Y^j\right).$$

We construct a lexicographic order on $\underline{y^j} \in T\left(\mathbb{P}^{*n}_{Y^j}\right)$. Any two sequences $\underline{y^j}, \underline{\hat{y}^j} \in T\left(\mathbb{P}^{*n}_{Y^j}\right)$ can be compared order-wise by observing the first leftmost symbol i such that y^j_i and $\hat{y}^j_i$ differ. The order relation gives higher precedence to the sequence with larger value in the ith symbol. For example, for the type $\mathbb{P} = (\frac{1}{2}, \frac{1}{2})$ with $n = 6$:

$$111000 > 110100 > 110010 > \ldots > 000111.$$

The encoder $E_j : \mathcal{M}_j \to T\left(\mathbb{P}^{*n}_{Y^j}\right)$ maps m_j to the m_jth lexicographically ordered sequence $\underline{y^j} \in T\left(\mathbb{P}^{*n}_{Y^j}\right)$. Likewise, the decoder$D_j = E_j^{-1} : T\left(\mathbb{P}^{*n}_{Y^j}\right) \to \mathcal{M}_j$ operates in the reverse direction. The mapping is accomplished with low complexity by perform-

ing combinatorial calculations and exploiting (4.7). If we consider multiplication as a fixed cost operation, then encoding and decoding can be done in $O(n)$ time by saving in memory, throughout the process, previously counted operations and dividing or multiplying by at most $|\mathcal{Y}|$ numbers. The encoding and decoding process follows from [Cov73].

4.4 Practical Algorithms for Specific Classes of Deterministic Broadcast Channels

Here in this section we consider specific types of deterministic broadcast channels that admit a low-complexity solution for coding at vertices. We will discuss an extension of the enumerative source coding approach for vertex rates of degraded deterministic broadcast channels that admits zero-error coding, as well as an iterative encoding approach for binning when an erasure-style situation is present.

4.4.1 Enumerative Source Coding at Vertex Rates for Degraded Deterministic Broadcast Channels

Here we consider *degraded* deterministic broadcast channels where y^1 and y^2 are degraded in the sense that $X \to Y^2 \to Y^1$ forms a Markov chain. Since y^1 and y^2 are functions of x, this is equivalent to saying that $y^1 = g(y^2)$ for some deterministic function g, and thus $H(Y^1|Y^2) = 0$. Now consider the vertex rate $(R_1, R_2) = (H(Y^1), H(Y^2|Y^1))$. In this setting, as in the case of more general degraded broadcast channels [Cov98], we can code for $\underline{y}^1$ to specify a cloud center, and for y^2 to specify satellite codewords. Since $\underline{y}^1$ can be recovered by both decoders, we can consider extending the enumerative source coding approach discussed in Section 4.3.2 to all stages of pipelined vertex coding. Specifically, select $\mathbb{P}^{*n}_{Y^1,Y^2}$ according to (4.6). We can enumeratively source code $\underline{y}^1$ from message m_1 with a distribution

$\mathbb{P}^{*n}_{Y^1}$ induced from $\mathbb{P}^{*n}_{Y^1,Y^2}$. Note that for any $\underline{y}^1 \in T\left(\mathbb{P}^{*n}_{Y^1}\right)$,

$$\left|\left\{\underline{y}^2 : \left(\underline{y}^1, \underline{y}^2\right) \in T\left(\mathbb{P}\right)\right\}\right| = \frac{\left|T\left(\mathbb{P}^{*n}_{Y^1,Y^2}\right)\right|}{\left|T\left(\mathbb{P}^{*n}_{Y^1}\right)\right|} = 2^{n\left[H\left(Y^2|Y^1\right)-o(n)\right]}.$$

To begin encoding m_2 into $\underline{y}^2$ given $\underline{y}^1$, the encoder next partitions the set of indices $\{1 \dots n\}$ into the sets $\left\{i_a(\underline{y}^1)\right\}_{a\in\mathcal{Y}_1}$ where

$$i_a(\underline{y}^1) = \left\{i : y_i^1 = a\right\}. \tag{4.8}$$

Note that for each $\underline{y}^1 \in T\left(\mathbb{P}^{*n}_{Y^1}\right)$, $\left\{\left|i_a(\underline{y}^1)\right|\right\}_{a\in\mathcal{Y}_1} = \left\{n\mathbb{P}^{*n}_{Y^1}(a)\right\}_{a\in\mathcal{Y}_1}$. Consider the induced conditional types

$$\left\{\mathbb{P}^{*n}_{Y^2|Y^1}(\cdot|a)\right\}_{a\in\mathcal{Y}_1} = \left\{\frac{\mathbb{P}^{*n}_{Y^1,Y^2}(\cdot|a)}{\mathbb{P}^{*n}_{Y^1}(a)}\right\}_{a\in\mathcal{Y}_1}$$

and note that there are $2^{n\left[P_{Y^1}(a)H\left(Y^2|Y^1=a\right)-o(n)\right]}$ distinct $\underline{y}^2_{|i_a(\underline{y}^1)}$ sequences such that $\left(\underline{y}^1, \underline{y}^2\right) \in T\left(\mathbb{P}\right)$. Thus we can represent $m_2 \in \left\{1 \dots 2^{nR_2}\right\}$ as a set of messages

$$m_2 \in \{1, \dots, 2^{nR_2}\} \tag{4.9a}$$

$$\Leftrightarrow$$

$$(m_{2a}, \dots m_{2a'}) \in \left(\{1, \dots, 2^{nR_{2a}}\}, \dots, \{1, \dots, 2^{nR_{2a'}}\}\right). \tag{4.9b}$$

Then for each $a \in \mathcal{Y}_1$ separately, let $R_{2a} = P_{Y^1}(a)H\left(Y^2|Y^1 = a\right)$ and enumeratively source code m_{2a} to $\underline{y}^2_{|i_a(\underline{y}^1)}$ according to Section 4.3.2. Decoding of Y^2 can be done with zero error because we are in a degraded setting where for some g, $Y^1 = g(Y^2)$. So the decoder for Y^2 can first recover $\underline{y}^1 = g(\underline{y}^2)$, and from this it can recover the indices $i_a(\underline{y}^1)$. Next it can perform the enumerative decoding algorithm described in Section 4.3.2 to map, for each $a \in \mathcal{Y}_1$, $\underline{y}^2_{|i_a(\underline{y}^1)}$ to m_{2a}. Finally, using (4.9), $\{m_{2a}\}_{a\in\mathcal{Y}_1}$ can be mapped to m_2. Since

$$R_2 = \sum_{a\in\mathcal{Y}_1} R_{2a} = \sum_{a\in\mathcal{Y}_1} P_{Y^1}(a)H\left(Y^2|Y^1 = a\right) = H\left(Y^2|Y^1\right),$$

this approach attains the vertex corner point $(R_1, R_2) = (H(Y^1), H(Y^2|Y^1))$.

We note that this approach does not apply to general deterministic broadcast channels because the side information $\underline{y}^1$ cannot always be constructed at the decoder for $\underline{y}^2$.

4.4.2 Low-Complexity Capacity-Achieving Codes for Erasure Encoding with Side Information

We now consider coding at vertex rates for a class of deterministic broadcast channels that have an erasure correcting style situation at the encoder:

Lemma 4.4.1. *Consider a deterministic broadcast channel specified with two receivers Y^1 and Y^2. Suppose that there exists an $a^* \in \mathcal{Y}_1$ such that for all $a \in (\mathcal{Y}_1 \setminus a^*)$,*

$$Y^1 = a \Rightarrow Y^2 = b \text{ for some } b \in \mathcal{Y}_2.$$

Define

$$B^* = \{b \in \mathcal{Y}_2 \ : \ \exists \ x \ s.t. f_1(x) = a^*, f_2(x) = b\}.$$

Then the dominant rate points $(R_1, R_2) \in \mathcal{R}[f_1, f_2]$ that maximize

$$\mu R_1 + (1-\mu) R_2 \tag{4.10}$$

for $\mu \in (\frac{1}{2}, 1]$ can be achieved by corner points $(R_1, R_2) = (H(Y^1), H(Y^2|Y^1))$ of the region $\mathcal{R}[P'(x); f_1, f_2]$ where $P'(x)$ has the property that the induced $\left\{P'_{Y^1,Y^2}(a^*, b)\right\}_{b \in B^*}$ are equal.

Proof. Note that since $\mu \in (\frac{1}{2}, 1]$, for any distribution $P(x)$, a vertex rate of $\mathcal{R}[P; f_1, f_2]$ corresponding to $(R_1, R_2) = (H(Y^1|Y^2), H(Y^2))$ will not suffice: interchanging R_1 and R_2 gives a strictly larger objective function. Now consider achieving the point $(R_1 = H(Y^1), R_2 = H(Y^2|Y^1)) \in \mathcal{R}[P; f_1, f_2]$ for some distribution P. Define P' as

$$P' = \sum_{k=1}^{|B^*|!} \frac{1}{|B^*|!} \pi_k(P) \tag{4.11}$$

where π_k operates on B^* as the kth element of the symmetric group and π_k is the identity operator on $(\mathcal{Y}_2 \backslash B^*)$. Let us denote $H'()$ as the entropy corresponding to P'. Note that since P' is a convex combination of permutations of P, by the Shur-concavity of the entropy function, $H'(Y^1) \geq H(Y^1)$. Also note that $\left\{ P'_{Y^1,Y^2}(a^*, b) \right\}_{b \in B^*}$ are equal. Thus

$$\begin{aligned} H'(Y^2|Y^1) &= P_{Y^1}(a^*)\, H'(Y^2|Y^1 = a^*) \\ &= P_{Y^1}(a^*) \log_2(|B^*|) \qquad (4.12) \\ &\geq P_{Y^1}(a^*)\, H(Y^2|Y^1 = *) \\ &= H(Y^2|Y^1) \end{aligned}$$

where (4.12) follows from (4.11). □

Note from this Lemma and (4.12) in particular that coding at rate $R_2 = H'(Y^2|Y^1)$ for Y^2 given Y^1 as side information seems somewhat related to coding over an erasure channel. We now pursue this observation in detail.

Low-Complexity Code Constructions: Duals of LT Codes

Let us now consider coding at rates $(R_1, R_2) = (H(Y^1), H(Y^2|Y^1))$ with joint distributions P' as discussed in the proof of Lemma 4.4.1. Note that to encode Y^1 at rate $R_1 = H(Y^1)$, we can use the enumerative source coding approach discussed in Section 4.3.2.

To encode $\underline{y}^2$ given $\underline{y}^1$, as discussed in Section 4.1.1, binning suffices as an achievable strategy. One general approach for using linear codes for binning purposes is as follows. A code and its parity check matrix H are fixed before transmission begins. A sequence $\underline{y}^1$ is selected for Rx 1 based on the shaping code discussed in Section 4.3.2. To send a message m_2 which we represent as a vector $\underline{s}^2$, to Rx 2, two conditions must be satisfied. The first is that $H \cdot \underline{y}^2 = \underline{s}^2$, which ensures that Rx 2 can decode the message $\underline{s}^2$ by looking at the bin index of $\underline{y}^2$. The second condition is that $\underline{y}^1$ and $\underline{y}^2$ are consistent, which is represented by the equation $T \cdot \underline{y}^1 = T \cdot \underline{y}^2$ where T is the

identity matrix with entries corresponding to the a^* symbols in $\underline{y}^1$ set to 0. We can combine these equations into a single linear system using block matrices to get

$$[H \quad T] \cdot \underline{y}^2 = [\underline{s} \quad T \cdot \underline{y}^1].$$

Thus a general linear code used for this problem needs a matrix inversion to determine $\underline{y}^2$ requiring $O(n^3)$ complexity. Ideally, we would like to use a low density parity check code or some other sparse graph code to reduce this complexity.

Throughout the remainder of this section we assume for convenience that $\mathcal{Y}_1 = \{0, 1, *\}$ and $\mathcal{Y}_2 = \{0, 1, \}$ but these results can directly be extended to $\mathbb{F}_{2^t}$. We now exploit the structure of this problem and its similarity to binary erasure quantization using codes on graphs [MY03]. In that setting, a sequence of symbols $\underline{y}^1 \in \{0, 1, *\}^n$ is given to an encoder which decodes to $\underline{x} \in C$ for some binary linear code $\mathcal{C}$ such that $\underline{x}$ agrees with $\underline{y}^1$ in non-erased positions. There are an exponential number of such $\underline{x}$'s for any typical $\underline{y}^1$, just as in our case. The authors exploit the fact that a k-dimensional binary linear code $\mathcal{C}$ of length n can be expressed in two ways:

$$\mathcal{C} = \{\underline{x} \mid H\underline{x} = \underline{0}\} = \{\underline{u}G\}_{\underline{u} \in \{0,1\}^k} \tag{4.13}$$

where H is the parity-check matrix and G is the generator matrix. They discuss how the dual code $C^\perp$ can be expressed as

$$\mathcal{C}^\perp = \{\underline{x} \mid G\underline{x} = \underline{0}\} = \{\underline{u}H\}_{\underline{u} \in \{0,1\}^{n-k}}. \tag{4.14}$$

They combine this with Forney's 'normal graph' representation [For01] for codes on graphs to dualize a code by using the same graph and simply replacing each local code associated with a vertex in the graph by its dual (for instance, see Figure 4-9 ignoring the bit values on the dongles). By dualizing capacity-achieving parity-check graphical representations of linear codes for the binary erasure channel (BEC), the authors construct a rate-distortion optimal generator-form graphical representation of the linear code for binary erasure quantization. The dual quantization algorithm

discussed in [MY03] fails if and only if the analogous BEC decoding algorithm fails. One slight difference in our setting is that there is an extra constraint that must be satisfied: if H is the parity-check matrix for C, then we must have $H\underline{y}^2 = \underline{s}^2$ where $\underline{s}^2$ is the message bin index. Thus mapping from a parity-check representation to a generator representation will not apply here, because the generator matrix for any code produces codewords $\underline{x}$ that lie in C, which means that $H\underline{x} = \underline{0}$. Moreover, attempting to dualize a generator representation that has a graphical representation like that of an LDPC will provably fail: any representation with a constant fraction of nodes with bounded degree will have a non-negligible probability of encoding failure [MY03, Theorem 1].

Luby has constructed LT codes [Lub02] that have degrees $O(\log n)$, are decoded in generator-representation form, and are provably capacity-achieving on the BEC under the following low-complexity algorithm:

ERASURE-DECODE-BEC$(G, \underline{y})$

1. **While** $\underline{u}$ has at least one unrecovered sample **do**
2. **if** $\exists$ one unerased (check) i connected to exactly one neighbor u_j **then**
3. Recover u_j immediately and propagate it to any adjacent unerased checks i' via $y_{i'} \doteq y_{i'} \oplus u_j$.
4. **else return** FAIL
5. **end if**
6. **end while**
7. Set $\underline{u}$ to the values from the checks obtained from $\underline{y}$
8. **return** $\underline{u}$

Reversely analogous to [MY03], dualizing an LT code in generator form yields another code in parity check matrix form. Once in parity matrix form, we can transform this to a syndrome-former representation by adding dongles on checks to represent the coset constraints for the message index [For01, sec. VIII.B]. The dual algorithm is as follows:

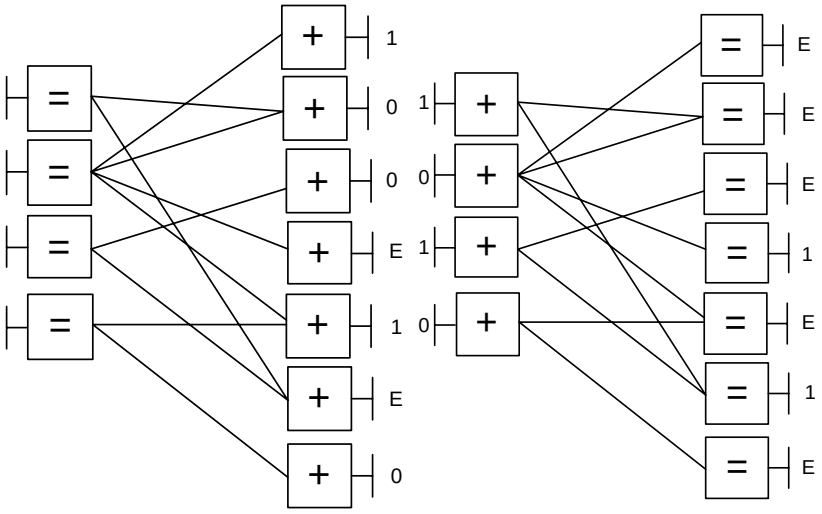


Figure 4-9: (L) Generator form LT code for decoding on a binary erasure channel; (R) syndrome former dual LT code for encoding on a deterministic broadcast channel

ERASURE-ENCODE-DBC$(H, \underline{s}, \underline{z})$

1. **While** $\underline{z}$ has at least one erased sample **do**
2. **if** $\exists$ one z_i connected to exactly one neighbor check j **then**
3. Reserve z_i to later satisfy check j with syndrome s_j and erase check j
4. **else return** FAIL
5. **end if**
6. **end while**
7. Arbitrarily set unreserved erased z_i values.
8. Set reserved variables to satisfy the corresponding checks starting from the last reserved variable and working backward to the first reserved variable
9. **return** $\underline{z}$

We are now in position to state the following proposition:

Proposition 4.4.2. *Consider a linear code with generator matrix G and its dual code with $G^{\perp} = H$. The algorithm* ***ERASURE-DECODE-BEC***$(G, \underline{y})$ *fails in step* 4 *if and only if the algorithm* ***ERASURE-ENCODE-DBC***$(H, \underline{s}, \underline{z})$ *fails in step* 4 *where $\underline{y}$ has erasures specified by $\underline{e}$ and $\underline{z}$ has erasures specified by $\underline{e}^{\perp} = 1 - \underline{e}$.*

The proof follows directly from [MY03, Proof of Theorem 4], where it is shown

that the Luby BEC decoding algorithm on a code in parity-check form H fails if and only if the dual BEQ encoding algorithm fails on the dual code in generator form with $G = H^{\perp}$.

The algorithm for decoding LT codes **ERASURE-DECODE-BEC**$(G, \underline{y})$ has $O(n \log n)$ and thus so does **ERASURE-ENCODE-DBC**$(H, \underline{s}, \underline{z})$. Figure 4-9 (L) gives an example of decoding with a generator form LT code. The partially erased received sequence $\underline{y}$ lies on the right and the decoder must recover $\underline{u}$ corresponding to the non-existent symbols on the left. **ERASURE-DECODE-BEC**$(G, \underline{y})$ performs successfully here and the unique solution is given by $\underline{u} = (1, 1, 0, 0)$ and thus $\underline{u}G = (1, 0, 0, 1, 1, 1, 0)'$. (R) of Figure 4-9 gives the syndrome-former dual of the LT code in (L). Here, the syndrome is given on the left part of the graph by $\underline{s} = (1, 0, 1, 0)'$. The partially erased sequence $\underline{z}$ lies on the right and the encoder must recover $\underline{z}$. **ERASURE-ENCODE-DBC**$(H, \underline{s}, \underline{z})$ performs successfully here and one possible solution is given by $\underline{z} = (0, 0, 0, 1, 1, 1, 1)'$

We now first discuss the wireless interference management problem provided in Section 4.2 and develop dual LT codes for a class of rates on the boundary of the capacity region.

Wireless Interference Management

Let us re-examine the example problem from Section 4.2 that served as a motivation for considering deterministic broadcast channels. We will evaluate $\mathcal{R}[f_1, f_2]$ and identify rates on the boundary that can be attained with our dual LT code construction framework.

Let us consider a point (R_1, R_2) on the boundary of $\mathcal{R}[f_1, f_2]$ that maximizes

$$\mu R_1 + (1 - \mu) R_2 \tag{4.15}$$

where $\mu \in (\frac{1}{2}, 1]$. By identifying $y^1 = *$ as a^*, note that Y^1 and Y^2 satisfy the conditions of Lemma 4.4.1. Thus encoding at the vertices $(R_1, R_2) = (H(Y^1), H(Y^2|Y^1))$ using $P'(x)$ given in Table 4.1 will attain all rates on the boundary of $\mathcal{R}[f_1, f_2]$ for

(X^1, X^2)	Y^1	Y^2	$P'(\cdot)$
$(-1,-1)$	-1	-1	$\frac{1}{2}(1-2p)$
$(-1,1)$	$*$	1	p
$(1,-1)$	$*$	-1	p
$(1,1)$	1	1	$\frac{1}{2}(1-2p)$

Table 4.1: Optimal input distributions for a portion of the boundary of $\mathcal{R}[f_1, f_2]$ for the wireless interference management example.

X	Y^1	Y^2	$P'(\cdot)$
1	1	1	$1-p$
2	0	1	$\frac{1}{2}p$
3	0	0	$\frac{1}{2}p$

Table 4.2: Optimal input distributions for a portion of the boundary of $\mathcal{R}[f_1, f_2]$ for the Blackwell channel.

$\mu \in (\frac{1}{2}, 1]$. Furthermore, using the code construction and encoding algorithms consisting of enumerative source coding for Y^1 and dual LT encoding for Y^2 given Y^1 suffices. Maximizing $R_1 + R_2$ (i.e. $\mu = \frac{1}{2}$) corresponds to making X uniform and this can be achieved by P' in Table 4.1 with $p = \frac{1}{4}$. Note that the distribution that maximizes sum rate has as the other corner point $(R_1, R_2) = (H(Y^1|Y^2), H(Y^2)) = (1, 1)$, which also maximizes R_2. Thus it suffices to only consider using distributions of the form P'. Figure 4-10 shows the capacity region, as well as the boundary points that can be attained with our enumerative followed by dual LT encoding approach (in green).

The Blackwell Channel

The Blackwell channel is considered the simplest non-trivial deterministc broadcast channel, and it also satisfies the conditions of Lemma 4.4.1. In this two-receiver channel, the input is $X \in \{1, 2, 3\}$ with binary outputs. Its input-output relationship is given in Table 4.2. We now discuss the capacity region. Consider a point (R_1, R_2)

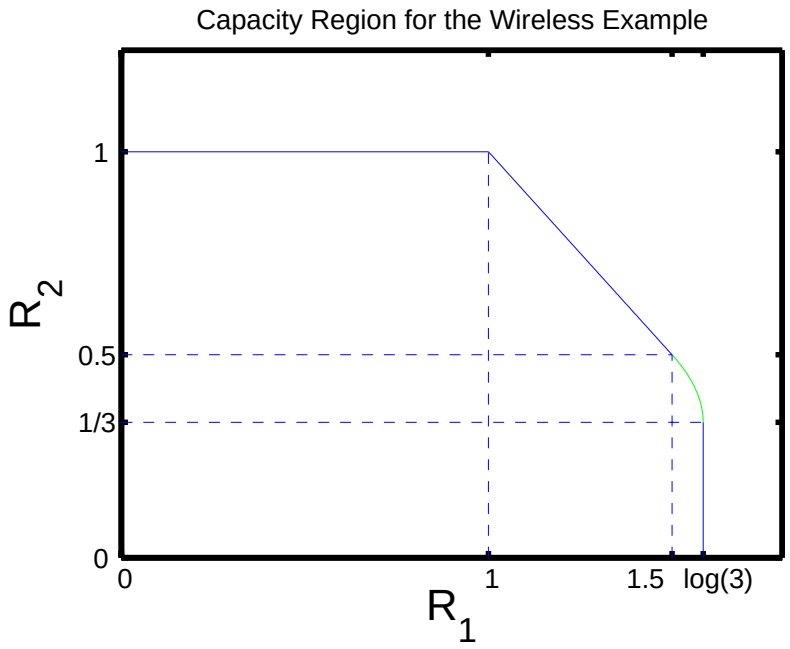


Figure 4-10: The capacity region of the wireless example. The boundary points in green can be attained with our proposed approach.

on the boundary of $\mathcal{R}[f_1, f_2]$ that maximizes

$$\mu R_1 + (1-\mu)R_2 \tag{4.16}$$

where $\mu \in (\frac{1}{2}, 1]$. By identifying $y^1 = 0$ as a^*, note that Y^1 and Y^2 satisfy the conditions of Lemma 4.4.1. Thus encoding at the vertices $(R_1, R_2) = (H(Y^1), H(Y^2|Y^1))$ using $P'(x)$ given in Table 4.2 will attain all rates on the boundary of $\mathcal{R}[f_1, f_2]$ for $\mu \in (\frac{1}{2}, 1]$. Furthermore, using the code construction and encoding algorithms consisting of enumerative source coding for Y^1 and dual LT encoding for Y^2 given Y^1 suffices. Maximizing $R_1 + R_2$ (i.e. $\mu = \frac{1}{2}$) corresponds to making X uniform and this can be achieved by P' in Table 4.2 with $p = \frac{2}{3}$. To maximize $\mu R_1 + (1-\mu)R_2$ where $\mu \in [0, \frac{1}{2})$, we simply reverse the roles of Y^1 and Y^2 and from the symmetry of the problem the reasoning above follows. Thus enumerative source coding for Y^2 at rate $R_2 = H(Y^2)$ followed by dual LT encoding Y^1 given Y^2 at rate $R_1 = H(Y^1|Y^2)$ suffices. The boundary points that can be attained with our enumerative followed by

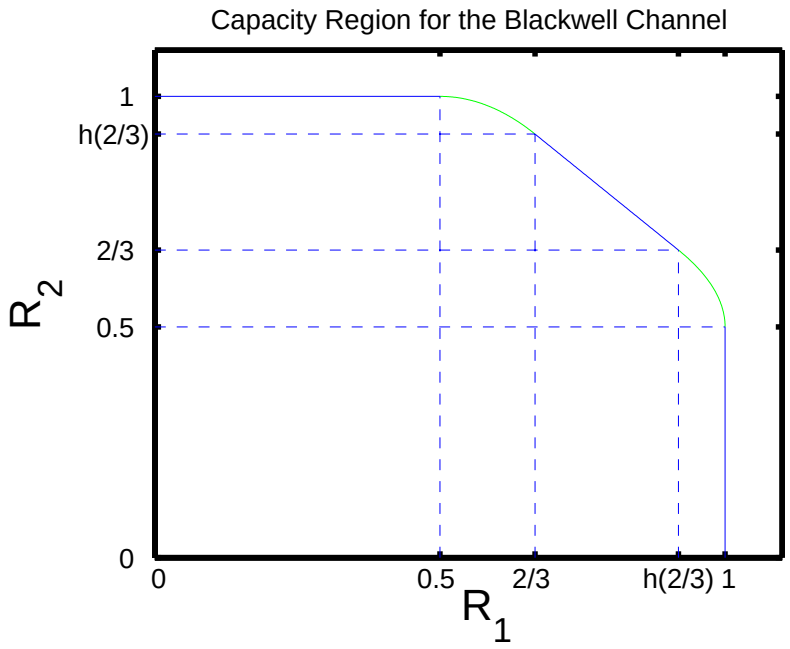


Figure 4-11: The capacity region of the Blackwell channel. The boundary points in green can be attained with our proposed approach. The rest of the points can be attained by time-sharing.

dual LT encoding approach are shown in green in Figure 4-11. Note that all other boundary points can be achieved with time-sharing.

Chapter 5

Summary and Future Work

5.1 Summary

In this thesis we presented a family of information-theoretic multiterminal information dissemination problems that have recently sparked interest in the research community. We have used a number of recent results in the applied math and electrical engineering literature to attack these problems and construct solutions with low complexity and good performance - quantified both theoretically and empirically.

To attack the Slepian-Wolf problem in Chapter 2, we applied the notion of 'rate-splitting' that was introduced in other communities and applied it here. This allowed us to significantly reduce the complexity of decoding by using a pipelined, single-user approach with side information. Furthermore, we used iterative decoding algorithms highly successful in the channel coding literature and showed that they are equally as successful in attaining points on the theoretical boundary. We justified this with simulations using simulated data as well as empirical data from the International Space Station.

In Chapter 3, we addressed polynomial complexity algorithms for universal coding when linear codes suffice. To do so, we used a 'divide-and-conquer' approach to construct large good codes from smaller good ones and connect them with the edges of a graph with good expansion properties. The expansion properties of the graph allowed us to formulate linear programming and iterative decoding algorithms that

provably work with exponentially high probability. Hopefully these provably good algorithms and code constructions will lead to further developments in constructions and algorithms for universal coding with better performance vs. complexity tradeoffs.

Finally, we addressed the deterministic broadcast channel in Chapter 4. First we illustrated how this problem bears similarities with the Slepian-Wolf near-lossless distributed data compression problem - both in terms of its achievable rate region as well as sufficient encoding/decoding strategies (binning). We also illustrated how a rate-splitting technique applies in this setting to reduce the complexity of encoding for an arbitrary number of receivers to encoding for one receiver with side information. However, we also showed that there are significant differences in the two problems in terms of searching through the bins. This observation also tells us that we cannot directly apply iterative techniques that have been successfully applied to channel coding and Slepian-Wolf problems.

However, we were able to have some success with low-complexity encoding. Before illustrating this, we first discussed why the *deterministic* broadcast channel is even interesting to consider. At first glance, one might not see this, since the outputs are deterministic functions of the inputs. We introduced some multiterminal wireless scenarios that, with some basic modulation techniques, can be cast into such deterministic channel models. Furthermore, we did have some success in constructing low-complexity encoding techniques. When encoding at vertices with pipelined approaches, we illustrated that the first stage of the encoding process can be done with 0 probability of error and linear complexity using Cover's 'enumerative source coding' technique. Furthermore, for our wireless scenario (as well as the Blackwell channel), we were able to construct a complete solution for certain rate points lying on the boundary of the achievable rate region by dualizing some codes and decoding algorithms for the binary erasure channel.

5.2 Future Work

Universal Coding

We note that the error exponents for the linear programming based universal decoder and the iterative 'expander codes' based decoder both have an error exponent characterized in terms of a Kullback-Leibler distance between an exponentially decaying quantity in the inner block length Δ and another quantity that decays much slower in Δ. The analysis derived here gives the latter approach a larger error exponent. However, empirical comparisons between linear programming decoding and expander code decoding on the same graphical representation of a code show that linear programming performs much better [FS05]. Thus we think a better error probability analysis for linear programming based decoding is possible to illustrate this observation theoretically.

Although the theoretical results of Chapter 3 discuss polynomial (and even linear) complexity decoding algorithms with guaranteed exponential error probability decay, the coefficients in the polynomials can be large enough to prevent implementation in silicon. Most likely, an approach with good empirical performance but no theoretical guarantees (such as applying iterative algorithms on bipartite graph codes with small fixed degrees and cycles) will have higher likelihood to make its way into real systems. Thus, one possible step in that direction is to consider designing iterative low-complexity algorithms that mimic the **UNIV-DEC-LP**$\big(G, \{H_j\}, \{\underline{s}_j\}\big)$and **UNIV-DEC-ITER**$\big(G, \{H_j\}, \{\underline{s}_j\}\big)$algorithms of Sections 3.5.2 and 3.6 with low complexity and good empirical performance.

Also, we find it worthwhile to mention that although linear codes suffice for all instances of point-to-point near-lossless data compression problems as well as their multiterminal counterparts (the Slepian-Wolf problem), this is in general not the case for channel coding. Indeed, a uniform probability distribution need not be capacity-achieving for all discrete memoryless channels (the same statement holds for multiple access). So the best we could hope for if we use universal coding under linear codes is to get close to (or attain) the random coding exponent corresponding to a uniform

input distribution. Characterizing the error exponent loss in such settings could potentially be the direction of future work.

Deterministic Broadcast Channels

In Chapter 4, we first illustrated through rate-splitting that for any achievable rate (which can be attained with a binning approach), we can isolate the encoding process to vertex coding - where a series of encoding with side information problems applies. Furthermore, we constructed a general-purpose linear complexity encoder to perform the first stage of the vertex encoding process. However, performing a general-purpose binning scheme with side information to code at below or at the conditional entropy seems quite daunting. In the erasure scenario, we were able to exploit the strong notion of joint typicality which can be expressed algebraically. This allowed us to dualize decoding schemes for the binary erasure channel and directly apply them to this problem. The same approach was shown to work for the Blackwell channel - again for the same reason: the one-to-one relationship between typicality and an algebraic constraint. Since matrix inversion is essentially all that is involved to find such a typical sequence, a worst-case $O(n^3)$ complexity is an upper bound on what would be required. We were able to construct codes and aglorithms with $O(n \log n)$ complexity that with probability approaching 1 can attain all achievable rates. Recently, Shokrollahi constructed 'Raptor Codes' [Sho03] for decoding on the BEC in generator-form that have higher probability of success and $O(n)$. It would not be surprising if one could dualize these codes to construct a linear complexity encoder for the same setting that we discussed above.

In the most general setting, however, the special cases that allow an algebraic constraint to characterize joint typicality does not exist. Thus a matrix inversion does not suffice, and in general the worst-case scenario of just searching through all bins has exponential complexity. Equally as daunting, it is not yet evident how to generalize our algorithms. One step in that direction might be to take the 'Survey Propagation' [BMZ05, BMWZ02, MZ02] framework, which attempts to solve iteratively an NP-hard 3-SAT problem (which also has an exponential number of equally good candidate

solutions consistent with what is observed) and does so with high probability provided certain conditions hold.

Bibliography

[AC89] N. Alon and F. R. K. Chung. Explicit construction of linear sized tolerant networks. Discrete Math, 72:15–19, 1989.

[AG02] A. Aaron and B. Girod. Compression with side information using turbo codes. In IEEE Data Compression Conference, pages 252–261, April 2002.

[AH89] S.-I. Amari and T.S. Han. Statistical inference under multiterminal rate restrictions: a differential geometric approach. IEEE Transactions on Information Theory, 35(2):217–227, 1989.

[Ahl71] R. Ahlswede. Multi-way communication channels. ISIT, pages 23–52, 1971.

[AN00] S Amari and H. Nagoka. Methods of Information Geometry, volume 191. Oxford University Press, January 2000.

[AU] A. Amarou and R. Urbanke. Ldpcopt. http://lthcwww.epfl.ch/research/ldpcopt/.

[BGT93] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting codes and decoding: Turbo codes. Proc. IEEE International Communications Conference, 1993.

[BM01] J. Bajcsy and P. Mitran. Coding for the Slepian-Wolf problem with turbo codes. In IEEE GLOBECOM, pages 1400–1404, November 2001.

[BMvT78] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg. On the intractability of certain coding problems. IEEE Transactions on Information Theory, 24(3):384–386, 1978.

[BMWZ02] A. Braunstein, M. Mézard, M. Weigt, and R. Zecchina. Constraint satisfaction by survey propagation. ArXiv Condensed Matter e-prints, dec 2002. http://lanl.arXiv.org/cond-mat/0212451.

[BMZ05] A. Braunstein, M. Mézard, and R. Zecchina. Survey propagation: an algorithm for satisfiability. In Random Structures and Algorithms, volume 27, pages 201–226, 2005.

[BT97] D. Bertsimas and J. N. Tsitsiklis. Introduction to Linear Optimization. Athena Scientific, Belmont, MA, 1997.

[BZ02] A. Barg and G. Zémor. Error exponents of expander codes. IEEE Transactions on Information Theory, 48(6):1725–1729, 2002.

[CGFRU01] S. Chung, Jr. G.D. Forney, T.J. Richardson, and R. Urbanke. On the design of low-density parity-check codes within 0.0045 db of the Shannon limit. IEEE Communications Letters, 5(2):58–60, February 2001.

[CK82] I. Csiszár and J. Korner. Information Theory: Coding Theorems for Discrete Memoryless Systems. Elsevier Science and Technology, 1982.

[Cos83] M. Costa. Writing on dirty paper. IEEE Transactions on Information Theory, 29(3):439–441, May 1983.

[Cov73] T. Cover. Enumerative source coding. IEEE Transactions on Information Theory, 19(1):73–77, Jan 1973.

[Cov75] T. M. Cover. A proof of the data compression theorem of Slepian and Wolf for ergodic sources. IEEE Transactions on Information Theory, 21:226–228, 1975.

[Cov98] T. M. Cover. Comments on broadcast channels. IEEE Transactions on Information Theory, 44:2524–2530, 1998.

[Csi82] I. Csiszár. Linear codes for sources and source networks: Error exponents, universal coding. IEEE Transactions on Information Theory, 28(4):585–592, 1982.

[Csi98] I. Csiszár. The method of types. IEEE Transactions on Information Theory, 44(6):2205–2523, 1998.

[CT91] T. M. Cover and J. Thomas. Elements of Information Theory. John Wiley & Sons, New York, NY, 1991.

[DFK04] S. C. Draper, B. Frey, and F. Kschischang. On interacting encoders and decoders in multiuser settings. In IEEE International Symposium on Information Theory, Chicaco , Ill, June 27–July 2 2004.

[DJ02] A. G. Dabak and D. H. Johnson. Relations between Kullback-Leibler distance and Fisher information. unpublished manuscript, 2002. http://cmc.rice.edu/docs/docs/Dab2002Sep1Relationsb.pdf.

[EtB05] U. Erez and S. ten Brink. A close-to-capacity dirty paper coding scheme. IEEE Transactions on Information Theory, 51(10):3417–3432, October 2005.

[EVKV02] M. Effros, K. Visweswariah, S. R. Kulkarni, and S. Verdu. Universal lossless source coding with the Burrows Wheeler transform. IEEE Transactions on Information Theory, 48(5):1061–1081, May 2002.

[Fel03] J. Feldman. Decoding Error-Correcting Codes via Linear Programming. PhD dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, September 2003.

[FGH02] W. Foslien, V. Guralnik, and K. Haigh. Report on data mining activity for the international space station. In Honeywell Labs, Minneapolis, MN, December 20 2002.

[FKKR01] G. Forney, R. Koetter, J. Kschischang, and A. Reznik. On the effective weights of pseudocodewords for codes defined on graphs with cycles. Codes, Systems and graphical models, pages 101–112, 2001.

[FKV01] B. J. Frey, R. Koetter, and A. Vardy. Signal space characterization of iterative decoding. IEEE Transactions on Information Theory, 47(2):766–781, 2001.

[FL98] M. Feder and A. Lapidoth. Universal decoding for channels with memory. IEEE Transactions on Information Theory, 44(5):1726–1745, 1998.

[FMS+04] J. Feldman, T. Malkin, C. Stein, R. A. Servedio, and M. J. Wainwright. LP decoding corrects a constant fraction of errors. In IEEE International Symposium on Information Theory, Chicago, Ill, June 27 – July 2 2004.

[For65] G. D. Forney. Concatendated Codes. PhD thesis, MIT, Cambridge, MA, June 1965.

[For66] G. D. Forney. Generalized minimum distance decoding. IEEE Transactions on Information Theory, 12(2):125–131, 1966.

[For01] G. D. Forney. Codes on graphs: Normal realizations. IEEE Transactions on Information Theory, pages 101–112, 2001.

[FS05] J. Feldman and C. Stein. LP decoding achieves capacity. In ACM-SIAM Symposium on Discrete Algorithms (SODA), January 2005.

[FWK03] J. Feldman, M. Wainwright, and D. R. Karger. Using linear programming to decode linear codes. Proceedings of Conference on Information Sciences and Systems, The John Hopkins University, March 2003.

[Gal62] R. Gallager. Low-density parity-check codes. IRE Transactions on Information Theory, 8:21–28, January 1962.

[Gal68] R. Gallager. Information Theory and Reliable Communication. John Wiley & Sons, New York, NY, 1968.

[Gal76] R. G. Gallager. Source coding with side information and universal coding. MIT LIDS Technical Report (LIDS-P-937), 1976. http://web.mit.edu/gallager/www/papers/paper5.pdf.

[GFZ01] J. Garcia-Frias and Y. Zhao. Compression of correlated binary sources using turbo codes. IEEE Communications Letters, 5:417–419, October 2001.

[GFZ03] J. Garcia-Frias and W. Zhong. LDPC codes for compression of multi-terminal sources with hidden Markov correlation. IEEE Communications Letters, 7(3):115–117, March 2003.

[GLT00] A. Ganti, A. Lapidoth, and I.E. Telatar. Mismatched decoding revisited: general alphabets, channels with memory, and the wide-band limit. IEEE Transactions on Information Theory, 46(7):2315–2328, 2000.

[Gop75] V. D. Goppa. Universal decoding for symmetric channels. Probl. Peredachi Inform., 11(1):15–22, 1975. (In Russian).

[GRUW01] A. Grant, B. Rimoldi, R. Urbanke, and P. A. Whiting. Rate-splitting multiple access for discrete memoryless channels. IEEE Transactions on Information Theory, 47(3):873–890, 2001.

[HA95] Te Sun Han and S. Amari. Parameter estimation with multiterminal data compression. IEEE Transactions on Information Theory, 41:1802–1833, 1995.

[HA98] Te Sun Han and S. Amari. Statistical inference under multiterminal data compression. IEEE Transactions on Information Theory, 44(6):2300–2324, 1998.

[HMEK04] T. Ho, M. Médard, M. Effros, and R. Koetter. Network coding for correlated sources. In Proceedings of CISS, 2004.

[HT96] R. Horst and H. Tuy. Global Optimization: Deterministic Approaches. Springer Verlag, Berlin, Germany, third revised and enlarged edition edition, 1996.

[Jor01] R. Jornsten. Data compression and Its Statistical Implications, with an Application to the Analysis of Microarray Images. PhD thesis, University of California, Berkeley, Berkeley, CA, December 2001.

[JVG03] N. Jindal, S. Vishwanath, and A. Goldsmith. On the duality between general multiple-access/broadcast channels. In IEEE International Symposium on Information Theory, Yokohama, Japan, June 29–July 4 2003.

[JY02] R. Jornsten and Bin Yu. Multiterminal estimation - extensions and geometric interpretation. In Information Theory, 2002. Proceedings. 2002 IEEE International Symposium on, 2002.

[KFL01] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. IEEE Transactions on Information Theory, 47(2):498 – 519, 2001.

[KV03] R. Koetter and P. O. Vontobel. Graph-covers and iterative decoding of finite length codes. Proceedings of Turbo Codes Conference, Brest, 2003.

[Lap96] A. Lapidoth. Mismatched decoding and the multiple-access channel. IEEE Transactions on Information Theory, 42(5):1439–1452, 1996.

[Lia72] H. Liao. Multiple Access Channels. PhD dissertation, University of Hawaii, Department of Electrical Engineering and Computer Science, June 1972.

[LLN+03] A. Liveris, C. Lan, K. Narayanan, Z. Xiong, and C. Georghiades. Slepian-Wolf coding of three binary sources using LDPC codes. In Proc.

Intl. Symp. Turbo Codes and Related Topics, Brest, France, September 2003.

[LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. Combinatorica, 8(3):261–277, 1988.

[Lub02] M. Luby. LT codes. Foundations of Computer Science, pages 271–280, November 2002.

[LXG03a] A. Liveris, Z. Xiong, and C. Georghiades. Distributed compression of binary sources using conventional parallel and serial concatenated convolutional codes. In Proc. IEEE DCC, pages 193–202, Brest, France, March 2003.

[LXG03b] A. D. Liveris, Z. Xiong, and C. Georghiades. Compression of binary sources with side information at the decoder using LDPC codes. IEEE Communications Letters, 6:440–442, October 2003.

[LZ77] A. Lempel and J. Ziv. A universal algorithm for sequential data compression. IEEE Transactions on Information Theory, pages 337–343, 1977.

[LZ78] A. Lempel and J. Ziv. Compression of individual sequences via variable-rate coding. IEEE Transactions on Information Theory, pages 530–536, 1978.

[LZ97] A. Lapidoth and J. Ziv. Universal decoding for noisy channels: an algorithmic approach. In Information Theory. 1997. Proceedings., 1997 IEEE International Symposium on, Ulm, 1997.

[Mar77] K. Marton. The capacity region of deterministic broadcast channels. In IEEE International Symposium on Information Theory, Paris-Cachan, France, 1977.

[MKLSS94] N. Merhav, G. Kaplan, A. Lapidoth, and S. Shamai Shitz. On information rates for mismatched decoders. IEEE Transactions on Information Theory, 40(6):1953–1967, 1994.

[MY03] E. Martinian and J. S. Yedida. Iterative quantization using codes on graphs. In Allerton Conference on Communication, Control, and Computing, 2003.

[MZ02] M. Mézard and R. Zecchina. Random k-satisfiability: from an analytic solution to an efficient algorithm. Physical Review E, 66, 2002.

[Pin78] M. S. Pinsker. Capacity of noiseless broadcast channels. Probl. Inform. Transm., pages 97–102, 1978.

[RU97] B. Rimoldi and R. Urbanke. Asynchronous Slepian-Wolf coding via source-splitting. In IEEE International Symposium on Information Theory, page 271, Ulm, Germany, June 29–July 4 1997.

[SCX04] V. Stankovic, S. Cheng, and Z. Xiong. On dualities in multiterminal coding problems. In Proc. 42nd Annual Allerton Conference on Communications, Control and Computing, Monticello, IL, October 2004.

[Sho03] A. Shokrollahi. Raptor codes. Allerton, Oct 2003.

[SLSX05] Y. Sun, A. D. Liveris, V. Stankovic, and Z. Xiong. Near-capacity dirty-paper code design: A source-channel coding approach. In Conference on Information Sciences and Systems, John Hopkins University, March 2005.

[SPR02] D. Schonberg, S. S. Pradhan, and K. Ramchandran. LDPC codes can approach the Slepian-Wolf bound for general binary sources. In Proceedings of the 40th Allerton Conference on Communication, Control and Computing, October 2002.

[SS96] M. Sipser and D. Spielman. Expander codes. IEEE Transactions on Information Theory, 42(6):1710–1722, 1996.

[SW73] D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. IEEE Transactions on Information Theory, 19(4):471–480, 1973.

[TGFZ03] T. Tian, J. Garcia-Frias, and W. Zhong. Compression of correlated sources using LDPC codes. In IEEE Data Compression Conference, 2003.

[VK04] P. O. Vontobel and R. Koetter. On the relationship between linear programming decoding and min-sum algorithm decoding. In International Symposium on Information Theory and its Applications, Parma, Italy, October 2004.

[Wil88] F. M. J. Willems. Totally asynchronous Slepian-Wolf data compression. IEEE Transactions on Information Theory, 34(1):35–44, 1988.

[WK03] S. Wicker and S. Kim. Fundamentals of Codes, Graphs, and Iterative Decoding. Kluwer Academic Publishers, Norwell, MA, 2003.

[WSS04] H. Weingarten, Y. Steinberg, and S. Shamai. The capacity region of the Gaussian MIMO broadcast channel. In IEEE International Symposium on Information Theory, Chicago, IL, June 2004.

[Zém01] G. Zémor. On expander codes. IEEE Transactions on Information Theory, 47(2):835–837, 2001.

Appendix A

Proofs of Chapter 2 Lemmas

A.1 Definitions

The following definitions and lemma are useful for proving Lemmas 2.2.1, 2.2.3 and 2.2.4. Define

$$e_{\min}(Q,\mathcal{U}) = \inf_{P\in\mathcal{P}(\mathcal{U})} \{H(P) + D(P\|Q)\} \tag{A.1}$$

$$e_{\max}(Q,\mathcal{U}) = \sup_{P\in\mathcal{P}(\mathcal{U})} \{H(P) + D(P\|Q)\}. \tag{A.2}$$

Lemma A.1.1. *Consider any $Q \in \mathcal{P}(\mathcal{U})$ such that $|\mathcal{U}| > 1$ and $Q(a) > 0$ for each $a \in \mathcal{U}$. Then $e_{\min}(Q,\mathcal{U})$ as defined in (A.1) satisfies $e_{\min}(Q,\mathcal{U}) > 0$, and $e_{\max}(Q,\mathcal{U})$ as defined in (A.2) satisfies $e_{\max}(Q,\mathcal{U}) < \infty$.*

Proof. For any $P \in \mathcal{P}(\mathcal{U})$,

$$H(P) + D(P\|Q) = \sum_{a\in\mathcal{U}} -P(a)\log_2(Q(a)). \tag{A.3}$$

Since $Q(a) > 0$ for each $a \in \mathcal{U}$, $Q(a) < 1$ for each $a \in \mathcal{U}$. Thus $-\log_2(Q(a)) > 0$ for each $a \in \mathcal{U}$. Since Q is fixed in the optimization (A.1), and since the log function is monotonic, there exists a P^* that minimizes (A.3) and satisfies $P^*(a_{\max}) = 1$ where

$a_{\max} \in \arg\max_{a \in \mathcal{U}} Q(a)$. Thus

$$e_{\min}(Q, \mathcal{U}) = -\log_2(Q(a_{\max})) > 0.$$

Since $Q(a) > 0$ for each $a \in \mathcal{U}$, $-\log_2(Q(a)) < \infty$ for all $a \in \mathcal{U}$. Again, since Q is fixed in the optimization (A.2), and since the log function is monotonic, there exists a P^* that maximizes (A.3) that satisfies $P^*(a_{\min}) = 1$ where $a_{\min} \in \arg\min_{a \in \mathcal{U}} Q(a)$. Thus

$$e_{\max}(Q, \mathcal{U}) = -\log_2(Q(a_{\min})) < \infty.$$

□

To aid in proving Lemmas 2.2.1, 2.2.2, and 2.2.4, map each $\underline{u}_{[n]} \in \mathcal{U}^n$ to $\tau_1\left(\underline{u}_{[n]}\right) \in \{0, 1, \ldots, |\mathcal{U}|^n - 1\}$ using the type class integral representation given in (2.14) with $\epsilon = 1$:

$$\tau_1\left(\underline{u}_{[n]}\right) = \left(\sum_{i=0}^{j(\underline{u}_{[n]})-1} |T(P^{i,n})| \right) + k(\underline{u}_{[n]}). \tag{A.4}$$

Define $\xi = |\mathcal{U}|^n$ and the random variable $\tilde{\underline{U}}^a_{[n]}(\epsilon)$ with alphabet $\{0, 1, \ldots, |\mathcal{U}|^n - 1\} \cup \{\xi\}$ in terms of $\underline{U}_{[n]}$ as

$$\tilde{\underline{U}}^a_{[n]}(\epsilon) = \begin{cases} \tau_1\left(\underline{U}_{[n]}\right) & \text{if } k(\underline{U}_{[n]}) < A\left(j(\underline{U}_{[n]}), \epsilon, n\right) \\ \xi & \text{if } k(\underline{U}_{[n]}) \geq A\left(j(\underline{U}_{[n]}), \epsilon, n\right). \end{cases} \tag{A.5}$$

For every $\epsilon \in [0, 1]$,

$$\underline{U}^a_{[n]}(\epsilon) = \tau_\epsilon\left(\underline{u}^{j,k}_{[n]}\right) \quad \text{iff} \quad \underline{U}_{[n]} = \underline{u}^{j,k}_{[n]} \text{ and } k < A(j, \epsilon, n) \tag{A.6}$$

$$\tilde{\underline{U}}^a_{[n]}(\epsilon) = \tau_1\left(\underline{u}^{j,k}_{[n]}\right) \quad \text{iff} \quad \underline{U}_{[n]} = \underline{u}^{j,k}_{[n]} \text{ and } k < A(j, \epsilon, n) \tag{A.7}$$

$$\underline{U}^a_{[n]}(\epsilon) = T_{\epsilon,n} \quad \text{iff} \quad k(\underline{U}_{[n]}) \geq A\left(j(\underline{U}_{[n]}), \epsilon, n\right) \tag{A.8}$$

$$\tilde{\underline{U}}^a_{[n]}(\epsilon) = \xi \quad \text{iff} \quad k(\underline{U}_{[n]}) \geq A\left(j(\underline{U}_{[n]}), \epsilon, n\right). \tag{A.9}$$

Thus $\underline{\tilde{U}}^a_{[n]}(\epsilon)$ and $\underline{U}^a_{[n]}(\epsilon)$ form a bijection. Note the following properties of $P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\cdot)$:

$$P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\xi) = P_{\underline{U}^a_{[n]}(\epsilon)}(T_{\epsilon,n}) \tag{A.10}$$

$$P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}\left(\tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right) = 1_{\{k<A(j,\epsilon,n)\}} P_{\underline{U}_{[n]}}\left(\underline{u}^{j,k}_{[n]}\right) \ \forall\, j \in \mathcal{J}(n), k \in \mathcal{K}(j,n). \tag{A.11}$$

Since $\underline{U}^a_{[n]}(\epsilon)$ is a function of $\underline{U}_{[n]}$, it follows that $\underline{U}^{\mathcal{S}}_{[n]} \to \underline{U}_{[n]} \to \underline{U}^a_{[n]}(\epsilon)$ forms a Markov chain. Since $\underline{U}^a_{[n]}(\epsilon) \leftrightarrow \underline{\tilde{U}}^a_{[n]}(\epsilon)$, $\underline{U}^{\mathcal{S}}_{[n]} \to \underline{U}_{[n]} \to \underline{\tilde{U}}^a_{[n]}(\epsilon)$ also forms a Markov chain. Thus for any $j \in \mathcal{J}(n)$, $k \in \mathcal{K}(j,n)$,

$$\begin{aligned}
P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^a_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]}, \tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right) &= \sum_{\underline{u}_{[n]}} P_{\underline{\tilde{U}}^a_{[n]}(\epsilon),\underline{U}_{[n]}}\left(\tau_1\left(\underline{u}^{j,k}_{[n]}\right), \underline{u}_{[n]}\right) P_{\underline{U}^{\mathcal{S}}_{[n]}|\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}|\underline{u}_{[n]}\right) \\
&= P_{\underline{\tilde{U}}^a_{[n]}(\epsilon),\underline{U}_{[n]}}\left(\tau_1\left(\underline{u}^{j,k}_{[n]}\right), \underline{u}^{j,k}_{[n]}\right) P_{\underline{U}^{\mathcal{S}}_{[n]}|\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}|\underline{u}^{j,k}_{[n]}\right) \\
&= P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)|\underline{U}_{[n]}}\left(\tau_1\left(\underline{u}^{j,k}_{[n]}\right)|\underline{u}^{j,k}_{[n]}\right) P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}, \underline{u}^{j,k}_{[n]}\right) \\
&= 1_{\{k<A(j,\epsilon,n)\}} P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}, \underline{u}^{j,k}_{[n]}\right) \qquad \text{(A.12)} \\
P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^a_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]}, \xi\right) &= P_{\underline{U}^{\mathcal{S}}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}\right) - \sum_{j\in\mathcal{J}(n)} \sum_{k\in\mathcal{K}(j,n)} P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^a_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]}, \tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right).
\end{aligned}$$

Define

$$DP^{n,\epsilon,\epsilon'}(\cdot,\cdot) \triangleq P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\cdot,\cdot) - P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^a_{[n]}(\epsilon')}(\cdot,\cdot) \tag{A.13}$$

and note that for any $1 \geq \epsilon > \epsilon' \geq 0$, $A(j,\epsilon,n) \geq A(j,\epsilon',n)$ implies

$$DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]}, \tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right) = \left(1_{\{k<A(j,\epsilon,n)\}} - 1_{\{k<A(j,\epsilon',n)\}}\right) P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}, \underline{u}^{j,k}_{[n]}\right) \geq 0 \tag{A.14}$$

$$DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]}, \xi\right) = -\sum_{j\in\mathcal{J}(n)} \sum_{k\in\mathcal{K}(j,n)} DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]}, \tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right) \leq 0. \tag{A.15}$$

A.2 Proof of Lemma 2.2.1

Assume without loss of generality that $0 \leq \epsilon' \leq \epsilon \leq 1$. Since $\epsilon \geq \epsilon'$ implies $A(j,\epsilon,n) \geq A(j,\epsilon',n)$,

$$\begin{aligned}
\underline{\tilde{U}}^a_{[n]}(\epsilon') &= 1_{\left\{\underline{\tilde{U}}^a_{[n]}(\epsilon) < T_{\epsilon',n}\right\}} \underline{\tilde{U}}^a_{[n]}(\epsilon) + 1_{\left\{\underline{\tilde{U}}^a_{[n]}(\epsilon) \geq T_{\epsilon',n}\right\}} \xi & \text{(A.16)}\\
P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\xi) &\leq P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(\xi) & \text{(A.17)}\\
P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(r) &\geq P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(r) \ \forall\, r \in \{0,\ldots,|\mathcal{U}|^n - 1\}. & \text{(A.18)}
\end{aligned}$$

As a result,

$$\begin{aligned}
\left| P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\cdot) - P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(\cdot) \right|_1 &= P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(\xi) - P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(\xi) + \sum_{r=0}^{|\mathcal{U}|^n-1} \left[P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(r) - P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(r) \right] \\
&= \left(1 - \sum_{r=0}^{|\mathcal{U}|^n-1} P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(r) \right) - \left(1 - \sum_{r=0}^{|\mathcal{U}|^n-1} P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(r) \right) \\
&\quad + \sum_{r=0}^{|\mathcal{U}|^n-1} \left[P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(r) - P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(r) \right] \\
&= 2 \sum_{r=0}^{|\mathcal{U}|^n-1} \left[P_{\underline{\tilde{U}}^a_{[n]}(\epsilon)}(r) - P_{\underline{\tilde{U}}^a_{[n]}(\epsilon')}(r) \right] \\
&= 2 \sum_{j \in \mathcal{J}(n)} \sum_{k \in \mathcal{K}(j,n)} \left(1_{\{k < A(j,\epsilon,n)\}} - 1_{\{k < A(j,\epsilon',n)\}} \right) P_{\underline{U}_{[n]}}\left(\underline{u}^{j,k}\right) \qquad \text{(A.19)} \\
&= 2 \sum_{j \in \mathcal{J}(n)} \left(A(j,\epsilon,n) - A(j,\epsilon',n) \right) 2^{-n\left[H\left(P^{j,n}\right) + D\left(P^{j,n} \| Q\right)\right]} \\
&= 2 \sum_{j \in \mathcal{J}(n)} \left(\lceil \epsilon |T\left(P^{j,n}\right)| \rceil - \lceil \epsilon' |T\left(P^{j,n}\right)| \rceil \right) 2^{-n\left[H\left(P^{j,n}\right) + D\left(P^{j,n} \| Q\right)\right]}.
\end{aligned}$$

Note that

$$\begin{aligned}
\left| \lceil \epsilon |T\left(P^{j,n}\right)| \rceil - \lceil \epsilon' |T\left(P^{j,n}\right)| \rceil - (\epsilon - \epsilon') |T\left(P^{j,n}\right)| \right| &\leq 1, \\
\sum_{j \in \mathcal{J}(n)} (\epsilon - \epsilon') |T\left(P^{j,n}\right)| 2^{-n\left[H\left(P^{j,n}\right) + D\left(P^{j,n} \| Q\right)\right]} &= \epsilon - \epsilon', \qquad \text{(A.20)}
\end{aligned}$$

where (A.20) follows from

$$1 = \sum_{\underline{u}_{[n]} \in \mathcal{U}^n} P_{\underline{U}_{[n]}}\left(\underline{u}_{[n]}\right) = \sum_{j \in \mathcal{J}(n)} \left|T\left(P^{j,n}\right)\right| 2^{-n\left[H\left(P^{j,n}\right)+D\left(P^{j,n}\|Q\right)\right]}.$$

Therefore

$$\begin{aligned}
\left|\left|P_{\tilde{\underline{U}}^a_{[n]}(\epsilon)}(\cdot) - P_{\tilde{\underline{U}}^a_{[n]}(\epsilon')}(\cdot)\right|_1 - 2(\epsilon-\epsilon')\right| &\leq 2\sum_{j\in\mathcal{J}(n)} 2^{-n\left[H\left(P^{j,n}\right)+D\left(P^{j,n}\|Q\right)\right]} \\
&\leq 2\left|\mathcal{P}_n(\mathcal{U})\right| 2^{-ne_{\min}(Q,\mathcal{U})} \qquad \text{(A.21)} \\
&\leq 2(n+1)^{|\mathcal{U}|} 2^{-ne_{\min}(Q,\mathcal{U})}, \qquad \text{(A.22)}
\end{aligned}$$

where (A.21) is due to (A.1) and (2.11) and (A.22) is due to (2.9). Thus

$$\lim_{n\to\infty}\left|\left|P_{\tilde{\underline{U}}^a_{[n]}(\epsilon)}(\cdot) - P_{\tilde{\underline{U}}^a_{[n]}(\epsilon')}(\cdot)\right|_1 - 2(\epsilon-\epsilon')\right| = 0,$$

by Lemma A.1.1. □

A.3 Proof of Lemma 2.2.2

Proof. Assume without loss of generality that $0 \leq \epsilon' < \epsilon \leq 1$. Thus

$$\begin{aligned}
\left|DP^{n,\epsilon,\epsilon'}(\cdot,\cdot)\right|_1 &= \sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left\{\left|DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right| + \sum_{j\in\mathcal{J}(n)}\sum_{k\in\mathcal{K}(j,n)}\left|DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]},\tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right)\right|\right\} \\
&= 2\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\sum_{j\in\mathcal{J}(n)}\sum_{k\in\mathcal{K}(j,n)} DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]},\tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right) \qquad \text{(A.23)} \\
&= 2\sum_{j\in\mathcal{J}(n)}\sum_{k\in\mathcal{K}(j,n)}\left(1_{\{k<A(j,\epsilon,n)\}} - 1_{\{k<A(j,\epsilon',n)\}}\right)\sum_{\underline{u}^{\mathcal{S}}_{[n]}} P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]},\underline{u}^{j,k}_{[n]}\right) \qquad \text{(A.24)} \\
&= 2\sum_{j\in\mathcal{J}(n)}\sum_{k\in\mathcal{K}(j,n)}\left(1_{\{k<A(j,\epsilon,n)\}} - 1_{\{k<A(j,\epsilon',n)\}}\right) P_{\underline{U}_{[n]}}\left(\underline{u}^{j,k}_{[n]}\right) \\
&= \left|P_{\tilde{\underline{U}}^a_{[n]}(\epsilon)}(\cdot) - P_{\tilde{\underline{U}}^a_{[n]}(\epsilon')}(\cdot)\right|_1 \qquad \text{(A.25)}
\end{aligned}$$

where (A.23) is due to (A.15), (A.24) is due to (A.12), and (A.25) is due to (A.19). From here we finish the proof by applying Lemma 2.2.1. □

A.4 Proof of Lemma 2.2.3

Proof. For an arbitrary $\epsilon \in [0,1]$,

$$\begin{aligned}
\frac{1}{n}H\left(\underline{U}^a_{[n]}(\epsilon)\right) &= \frac{1}{n}\sum_{k=0}^{T_{\epsilon,n}} -\Pr\left(\underline{U}^a_{[n]}(\epsilon)=k\right)\log_2\left(\Pr\left(\underline{U}^a_{[n]}(\epsilon)=k\right)\right) \\
&= -\frac{1}{n}\sum_{j\in\mathcal{J}(n)} A\left(j,\epsilon,n\right)2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]}\log_2\left(2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]}\right) \\
&\quad -\frac{1}{n}\Pr\left(\underline{U}^a_{[n]}(\epsilon)=T_{\epsilon,n}\right)\log_2\left(\Pr\left(\underline{U}^a_{[n]}(\epsilon)=T_{\epsilon,n}\right)\right) \\
&= \sum_{j\in\mathcal{J}(n)}\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]A\left(j,\epsilon,n\right)2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]} \\
&\quad -\frac{1}{n}\Pr\left(\underline{U}^a_{[n]}(\epsilon)=T_{\epsilon,n}\right)\log_2\left(\Pr\left(\underline{U}^a_{[n]}(\epsilon)=T_{\epsilon,n}\right)\right). \qquad \text{(A.26)}
\end{aligned}$$

Therefore

$$\begin{aligned}
\mathcal{H}\left(U^a(\epsilon)\right) &= \lim_{n\to\infty}\frac{1}{n}H\left(\underline{U}^a_{[n]}(\epsilon)\right) \\
&= \lim_{n\to\infty}\sum_{j\in\mathcal{J}(n)}\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]A\left(j,\epsilon,n\right)2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]} \qquad \text{(A.27)} \\
&= \lim_{n\to\infty}\sum_{j\in\mathcal{J}(n)}\left\lceil\epsilon\left|T\left(P^{j,n}\right)\right|\right\rceil\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]}
\end{aligned}$$

where in (A.27), (A.26) vanishes because for any $p\in[0,1]$, $0\le -p\log_2 p\le\frac{1}{2}$.

Note that

$$0\le\left\lceil\epsilon|T\left(P^{j,n}\right)|\right\rceil-\epsilon|T\left(P^{j,n}\right)|\le 1,$$

$$\sum_{j\in\mathcal{J}(n)}\epsilon\left|T\left(P^{j,n}\right)\right|\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]}=\epsilon H(U) \qquad \text{(A.28)}$$

where (A.28) follows from (2.10) since

$$\begin{aligned} H(U) &= \frac{1}{n}H\left(\underline{U}_{[n]}\right) = \frac{1}{n}\sum_{\underline{u}_{[n]}\in\mathcal{U}^n} -P_{\underline{U}_{[n]}}\left(\underline{u}_{[n]}\right)\log_2 P_{\underline{U}_{[n]}}\left(\underline{u}_{[n]}\right) \\ &= -\frac{1}{n}\sum_{j\in\mathcal{J}(n)} \left|T\left(P^{j,n}\right)\right| 2^{-n\left[H\left(P^{j,n}\right)+D\left(P^{j,n}\|Q\right)\right]}\log_2\left(2^{-n\left[H\left(P^{j,n}\right)+D\left(P^{j,n}\|Q\right)\right]}\right) \\ &= \sum_{j\in\mathcal{J}(n)} \left|T\left(P^{j,n}\right)\right|\left[H\left(P^{j,n}\right)+D\left(P^{j,n}\|Q\right)\right]2^{-n\left[H\left(P^{j,n}\right)+D\left(P^{j,n}\|Q\right)\right]}. \end{aligned}$$

Thus

$$\begin{aligned} 0 \le \mathcal{H}\left(U^a(\epsilon)\right) - \epsilon H(U) &\le \lim_{n\to\infty}\sum_{j\in\mathcal{J}(n)}\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]2^{-n\left[H\left(P^{j,n}\right)+D(P^{j,n}\|Q)\right]} \\ &\le \lim_{n\to\infty}\left|\mathcal{P}_n\left(\mathcal{U}\right)\right| e_{\max}\left(Q,\mathcal{U}\right)2^{-ne_{\min}(Q,\mathcal{U})} \qquad \text{(A.29)} \\ &\le e_{\max}\left(Q,\mathcal{U}\right)\lim_{n\to\infty}(n+1)^{|\mathcal{U}|}2^{-ne_{\min}(Q,\mathcal{U})} \qquad \text{(A.30)} \\ &= 0 \qquad \text{(A.31)} \end{aligned}$$

where (A.29) is due to (A.1), (A.2), (2.11); (A.30) is due to (2.9); and (A.31) is due to Lemma A.1.1. □

A.5 Proof of Lemma 2.2.4

Proof. We first show that $\mathcal{H}\left(\underline{U}^{\mathcal{S}}, U^a(\epsilon)\right)$ is continuous in ϵ. Assume $0 < \epsilon' < \epsilon < 1$. Note from Lemma A.1.1 that

$$\begin{aligned} e_{\min}\left(Q_{U^{\mathcal{S}}},\mathcal{U}^{\mathcal{S}}\right) &> 0 \\ e_{\max}\left(Q_{U^{\mathcal{S}},U},\mathcal{U}^{\mathcal{S}}\times\mathcal{U}\right) &< \infty. \qquad \text{(A.32)} \end{aligned}$$

Define

$$\begin{aligned} L(\epsilon,\epsilon',n) &\triangleq -\frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\Big[P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right) \\ &\qquad -P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\Big] \\ DH_n(\epsilon,\epsilon') &\triangleq \frac{1}{n}\left[H\left(\underline{U}^{\mathcal{S}}_{[n]},\underline{U}^{a}_{[n]}(\epsilon)\right)-H\left(\underline{U}^{\mathcal{S}}_{[n]},\underline{U}^{a}_{[n]}(\epsilon')\right)\right] \end{aligned}$$

Then

$$\begin{aligned} DH_n(\epsilon,\epsilon') &= \frac{1}{n}\left[H\left(\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)\right)-H\left(\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')\right)\right] \\ &= L(\epsilon,\epsilon',n) \\ &- \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\sum_{j\in\mathcal{J}(n)}\sum_{k\in\mathcal{K}(j,n)} DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]},\tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{U}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]},\underline{u}^{j,k}_{[n]}\right). \end{aligned}$$

We now bound $L(\epsilon,\epsilon',n)$. Note that, for all $n>n_0=\lceil\frac{1}{e_{\min}\left(Q_{U^{\mathcal{S}}},\mathcal{U}^{\mathcal{S}}\right)}\rceil$ and all $\underline{u}^{\mathcal{S}}_{[n]}$,

$$P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\le P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\le P_{\underline{U}^{\mathcal{S}}_{[n]}}\left(\underline{u}^{\mathcal{S}}_{[n]}\right)\le 2^{-ne_{\min}\left(Q_{U^{\mathcal{S}}},\mathcal{U}^{\mathcal{S}}\right)}<\frac{1}{2}$$

where the first inequality follows from (A.15). Since the function $g(x)=-x\log_2 x$ is monotonically increasing on $[0,\frac{1}{2})$, for all $n>n_0$ and all $\underline{u}^{\mathcal{S}}_{[n]}$,

$$-\Bigg[P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) -P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\Bigg]\le 0$$

which implies $L(\epsilon,\epsilon',n)\leq 0$. We can also lower bound $L(\epsilon,\epsilon',n)$ as follows:

$$\begin{aligned}
L(\epsilon,\epsilon',n) &= \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\Bigg[P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) \\
&\qquad -P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\Bigg] \\
&+ \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)-P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) \\
&= \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left[P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\log_2\left(\frac{P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)}{P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)}\right)\right] \\
&+ \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)-P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) \\
&\geq \frac{1}{n}\left(\sum_{\underline{u}^{\mathcal{S}}_{[n]}}P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\log_2\frac{\left(\sum_{\underline{u}^{\mathcal{S}}_{[n]}}P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)}{\left(\sum_{\underline{u}^{\mathcal{S}}_{[n]}}P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)} \qquad \text{(A.33)} \\
&+ \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)-P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) \\
&= \frac{1}{n}\left(P_{\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\xi\right)\right)\log_2\frac{\left(P_{\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\xi\right)\right)}{\left(P_{\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\xi\right)\right)} \\
&+ \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)-P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) \\
&\geq 0 \qquad \text{(A.34)} \\
&+ \frac{1}{n}\sum_{\underline{u}^{\mathcal{S}}_{[n]}}\left(P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon')}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)-P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right)\right)\log_2 P_{\underline{U}^{\mathcal{S}}_{[n]},\underline{\tilde{U}}^{a}_{[n]}(\epsilon)}\left(\underline{u}^{\mathcal{S}}_{[n]},\xi\right) \\
&\geq -\frac{1}{2}e_{\max}\left(Q_{U^{\mathcal{S}},U},\mathcal{U}^{\mathcal{S}}\times\mathcal{U}\right)\left|DP^{n,\epsilon,\epsilon'}\left(\cdot,\cdot\right)\right|_1 \qquad \text{(A.35)}
\end{aligned}$$

where (A.33) follows from the log-sum inequality [CT91, p. 29], (A.34) is due to (A.17), and (A.35) is due to (A.2), Lemma A.1.1, and (A.23). Thus

$$\begin{aligned}
& -\frac{1}{2} e_{\max}\left(Q_{U^{\mathcal{S}},U}, \mathcal{U}^{\mathcal{S}} \times \mathcal{U}\right) \left| DP^{n,\epsilon,\epsilon'}(\cdot,\cdot) \right|_1 \\
\leq\ & DH_n(\epsilon,\epsilon') \\
\leq\ & e_{\max}\left(Q_{U^{\mathcal{S}},U}, \mathcal{U}^{\mathcal{S}} \times \mathcal{U}\right) \sum_{\underline{u}^{\mathcal{S}}_{[n]}} \sum_{j \in \mathcal{J}(n)} \sum_{k \in \mathcal{K}(j,n)} DP^{n,\epsilon,\epsilon'}\left(\underline{u}^{\mathcal{S}}_{[n]}, \tau_1\left(\underline{u}^{j,k}_{[n]}\right)\right) \quad \text{(A.36)} \\
=\ & \frac{1}{2} e_{\max}\left(Q_{U^{\mathcal{S}},U}, \mathcal{U}^{\mathcal{S}} \times \mathcal{U}\right) \left| DP^{n,\epsilon,\epsilon'}(\cdot,\cdot) \right|_1, \quad \text{(A.37)}
\end{aligned}$$

where (A.36) is due to (A.32), and (A.37) is due to (A.23). Thus

$$\left| \lim_{n\to\infty} DH_n(\epsilon,\epsilon') \right| = \left| \mathcal{H}\left(\underline{U}^{\mathcal{S}}, U^a(\epsilon)\right) - \mathcal{H}\left(\underline{U}^{\mathcal{S}}, U^a(\epsilon')\right) \right| \leq e_{\max}\left(Q_{U^{\mathcal{S}},U}, \mathcal{U}^{\mathcal{S}} \times \mathcal{U}\right)(\epsilon - \epsilon')$$

by Lemma 2.2.2. Thus $\mathcal{H}\left(\underline{U}^{\mathcal{S}}, U^a(\epsilon)\right)$ is continuous in ϵ.

Finally, $\mathcal{H}\left(\underline{U}^{\mathcal{S}} | U^a(\epsilon)\right)$ is continuous in ϵ due to the continuity of $\mathcal{H}\left(U^a(\epsilon)\right)$ and $\mathcal{H}\left(\underline{U}^{\mathcal{S}}, U^a(\epsilon)\right)$ along with the chain rule for entropy. The endpoints are contained because $\underline{U}^a_{[n]}(0)$ is a point mass and thus $\frac{1}{n} H\left(\underline{U}^a_{[n]}(0)\right) = 0$ and $\underline{U}^a_{[n]}(1)$ is bijective with U, and thus $\frac{1}{n} H\left(\underline{U}^a_{[n]}(1)\right) = H(U)$. □

A.6 Proof of Lemma 2.4.1

Here we consider the ML decoding error exponent for source decoding x when side information y is known at the decoder. Denote $P_e(y)$ to be the error probability conditioned upon receiving y. Then from [Gal76] we have that

$$\begin{aligned}
\frac{-1}{n} \log P_e(y) \geq E_{x|y}(R,y) &\triangleq \max_{0 \leq \rho \leq 1} \rho R - E_{0,x|y}(\rho, y) \\
E_{0,x|y}(\rho,y) &\triangleq (1+\rho) \log \left[\sum_x Q(x|y)^{\frac{1}{1+\rho}} \right] \quad \text{(A.38)}
\end{aligned}$$

For future reference, let us define the tilted distributions

$$Q_\rho(x|y) \triangleq \frac{Q(x|y)^{\frac{1}{1+\rho}}}{\sum_x Q(x|y)^{\frac{1}{1+\rho}}} \tag{A.39}$$

$$Q_\rho(y) \triangleq \frac{P(y)\left[\sum_x Q(x|y)^{\frac{1}{1+\rho}}\right]^{1+\rho}}{\sum_y P(y)\left[\sum_x Q(x|y)^{\frac{1}{1+\rho}}\right]^{1+\rho}}. \tag{A.40}$$

Differentiating with respect to ρ to find a stationary point, we can relate $E_{x|y}(R,y)$ and R parametrically in terms of ρ:

$$R = \frac{\partial E_{0,x|y}(\rho,y)}{\partial \rho} = H(X_\rho|y)$$

where the second equality above can be verified with calculation, as mentioned in [Gal76]. Now let us consider averaging over y when y itself is memoryless. Again, from [Gal76], we have:

$$\frac{-\log P_e}{n} \geq E_{x|y}(R) \triangleq \max_{0\leq\rho\leq 1} \rho R - E_{0,x|y}(\rho),$$
$$E_{0,x|y}(\rho) \triangleq \log\left(\sum_y P(y)\left[\sum_x Q(x|y)^{\frac{1}{1+\rho}}\right]^{1+\rho}\right) \tag{A.41}$$

Differentiating with respect to ρ to find a stationary point, we can relate $E_{x|y}(R)$ and R in terms of ρ:

$$R = \frac{\partial E_{0,x|y}(\rho)}{\partial \rho} = H(X_\rho|Y_\rho) \tag{A.42}$$

$$\Rightarrow E_{x|y}(R) = \rho H(X_\rho|Y_\rho) - E_{0,x|y}(\rho)$$

$$\Rightarrow \frac{\partial}{\partial\rho}E_{x|y}(R) = \rho\frac{\partial}{\partial\rho}H(X_\rho|Y_\rho)$$

$$\Rightarrow E'_{x|y}(R) \triangleq \frac{dE_{x|y}(R)}{dR} = \frac{\partial E_{x|y}(R)}{\partial\rho} = \rho \tag{A.43}$$

$$\Rightarrow E''_{x|y}(R) = \frac{\frac{\partial E'_{x|y}(R)}{\partial\rho}}{\frac{\partial R}{\partial\rho}} = \frac{1}{\frac{\partial}{\partial\rho}H(X_\rho|Y_\rho)} \tag{A.44}$$

where the second equality in (A.42) can be verified with tedious calculations, as mentioned in [Gal76]. Note from (A.42),(A.39) (A.40), and (A.43) that

$$E'_{x|y}\left(H(X|Y)\right) = 0. \tag{A.45}$$

Now note that

$$\begin{aligned}\frac{\partial}{\partial\rho}\left\{H\left(X_\rho|Y_\rho\right)\right\} &= \sum_y \frac{\partial}{\partial\rho}\left\{Q_\rho(y)H\left(X_\rho|y\right)\right\} \\ &= \sum_y H\left(X_\rho|y\right)\frac{\partial Q_\rho(y)}{\partial\rho} + \sum_y Q_\rho(y)\frac{\partial}{\partial\rho}\left\{H\left(X_\rho|y\right)\right\} \qquad \text{(A.46a)}\end{aligned}$$

To address $\frac{\partial Q_\rho(y)}{\partial\rho}$, note that in general for any differentiable function g, we have $g'(\rho) = g(\rho)\frac{d}{d\rho}\left\{\log g(\rho)\right\}$. Thus

$$\begin{aligned}\log Q_\rho(y) &= \log P(y) + (1+\rho)\log\left[\sum_x Q(x|y)^{\frac{1}{1+\rho}}\right] \\ &\quad - \log\left[\sum_y P(y)\left[\sum_x Q(x|y)^{\frac{1}{1+\rho}}\right]^{1+\rho}\right] \\ &= \log P(y) + E_{0,x|y}\left(\rho, y\right) - E_{0,x|y}\left(\rho\right) \qquad \text{(A.47)} \\ \Rightarrow \frac{\partial \log Q_\rho(y)}{\partial\rho} &= H\left(X_\rho|y\right) - H\left(X_\rho|Y_\rho\right) \qquad \text{(A.48)} \\ \Rightarrow \frac{\partial Q_\rho(y)}{\partial\rho}\Big|_{\rho=0} &= Q(y)\left[H\left(X|y\right) - H\left(X|Y\right)\right] \qquad \text{(A.49)}\end{aligned}$$

where (A.47) is due to (A.38), (A.41). As for $\frac{\partial H(X_\rho|y)}{\partial\rho}$, note that

$$\begin{aligned}H\left(X_\rho|y\right) &= -D\left(Q_\rho(x|y)\|\mathbb{U}\right) + \log|\mathcal{X}| \qquad \text{(A.50)} \\ \Rightarrow \frac{\partial}{\partial\rho}\left\{H\left(X_\rho|y\right)\right\} &= -\frac{\partial}{\partial\rho}\left\{D\left(Q_\rho(x|y)\|\mathbb{U}\right)\right\} \qquad \text{(A.51)}\end{aligned}$$

where $\mathbb{U}(x) = \frac{1}{|\mathcal{X}|}$. Dabak and Johnson show in [DJ02] that for any two probability distributions W_0 and W_1, the distribution W_t defined by

$$W_t(x) \triangleq \frac{W_0(x)^{1-t} W_1(x)^t}{\sum_a W_0(a)^{1-t} W_1(a)^t} \tag{A.52}$$

are related to the Fisher information $F(t)$ according to:

$$\frac{d}{dt}\left\{D\left(W_t \| W_0\right)\right\} = tF(t), \tag{A.53}$$

$$F(t) \triangleq \sum_x \left(\frac{d \log W_t(x)}{dt}\right)^2 W_t(x) \tag{A.54}$$

$$= \sum_x W_t(x) \left[\log \frac{W_1(x)}{W_0(x)}\right]^2 - \left[\sum_x W_t(x) \log \frac{W_1(x)}{W_0(x)}\right]^2 \tag{A.55}$$

We would like to characterize $Q_\rho(x|y)$ in terms of a W_t of the form (A.52):

$$\boxed{\begin{array}{c} 0 \le \rho \le \infty \\ \rho = 0 : Q_\rho(x|y) = Q(x|y) \\ \rho = \infty : Q_\rho(x|y) = \dfrac{1}{|\mathcal{X}|} \end{array}} \iff \boxed{\begin{array}{c} 0 \le t \le 1 \\ t = 0 : W_t(x) = W_0(x) \\ t = 1 : W_t(x) = W_1(x) \end{array}}$$

Thus by setting

$$\begin{aligned} W_0(x) &= Q_\infty(x|y) = \frac{1}{|\mathcal{U}|} \\ W_1(x) &= Q_0(x|y) = Q(x|y) \\ t &= \frac{1}{1+\rho} \end{aligned}$$

it follows that

$$\begin{aligned}
\frac{\partial H\left(X_\rho|y\right)}{\partial\rho} &= -\frac{\partial}{\partial\rho}\left\{D\left(Q_\rho(x|y)\|\mathbb{U}\right)\right\} \\
&= -\frac{dt}{d\rho}\frac{d}{dt}\left\{D\left(W_t\|W_0\right)\right\}\Big|_{t=\frac{1}{1+\rho}} \\
&= \frac{1}{(1+\rho)^3}F\left(\frac{1}{1+\rho}\right) \\
\Rightarrow \frac{\partial H\left(X_\rho|y\right)}{\partial\rho}\Big|_{\rho=0} &= F(1) \\
&= -H(X|y)^2 + \sum_x Q(x|y)\log^2[Q(x|y)] \qquad \text{(A.56)}
\end{aligned}$$

Thus it follows from (A.44),(A.46), (A.49), and (A.56) that

$$\boxed{E''_{x|y}\left(H(X|Y)\right) = \frac{1}{-H(X|Y)^2 + \sum_{x,y} Q(x,y)\log^2[Q(x|y)]}}$$

A.7 Proof of Lemma 2.4.2

Note that

$$\begin{aligned}
\log Q_\rho(x,y) &= \log Q_\rho(x|y) + \log Q_\rho(y) \\
&= \frac{1}{1+\rho}\left[\log Q(x|y) - E_{0,x|y}\left(\rho,y\right)\right] + \log P(y) + E_{0,x|y}\left(\rho,y\right) - E_{0,x|y}\left(\rho\right) \\
\Rightarrow \frac{\partial \log Q_\rho(x,y)}{\partial\rho} &= \frac{-1}{1+\rho}\frac{\partial E_{0,x|y}\left(\rho,y\right)}{\partial\rho} + \frac{-1}{(1+\rho)^2}\left[\log Q(x|y) - E_{0,x|y}\left(\rho,y\right)\right] \\
&\quad + \frac{\partial E_{0,x|y}\left(\rho,y\right)}{\partial\rho} - \frac{\partial E_{0,x|y}\left(\rho\right)}{\partial\rho} \\
&= \frac{-1}{1+\rho}\left[H(X_\rho|y) + \log Q_\rho(x|y)\right] + H(X_\rho|y) - H(X_\rho|Y_\rho) \qquad \text{(A.57)}
\end{aligned}$$

Thus

$$\begin{aligned}
\left(\frac{\partial \log Q_\rho(x,y)}{\partial \rho}\right)^2\Big|_{\rho=0} &= \left(\log Q(x|y) + H\left(X|Y\right)\right)^2 \\
&= H\left(X|Y\right)^2 + 2\log Q(x|y) H\left(X|Y\right) + \log^2\left[Q(x|y)\right] \\
\Rightarrow F(\rho)\Big|_{\rho=0} &= -H(X|Y)^2 + \sum_{x,y} Q(x,y)\log^2[Q(x|y)]. \qquad \square
\end{aligned}$$

Appendix B

Proofs of Chapter 3 Lemmas

B.1 Proof of Lemma 3.2.2

Suppose we have that a parity check matrix H has $d_{\text{univ}} = d = N\delta$. From the definition of d_{univ} in ((3.17)) it follows that for any nonzero $\underline{\tilde{u}} \in \text{Co}\,(H, \underline{0})$, the following holds:

$$w_h\,(\underline{\tilde{u}}) \geq d \quad \Leftrightarrow \quad w_h\,(\underline{\tilde{u}} \oplus \underline{1}) \leq n - d \tag{B.1}$$

$$\text{and } w_h\,(\underline{\tilde{u}}) \leq n - d \quad \Leftrightarrow \quad w_h\,(\underline{\tilde{u}} \oplus \underline{1}) \geq d. \tag{B.2}$$

Then if

$$w_h\,(\underline{u}) < \frac{1}{2}d \Leftrightarrow w_h\,(\underline{u} \oplus \underline{1}) > n - \frac{1}{2}d \tag{B.3}$$

is satisfied, we have

1.

$$\begin{aligned} w_h\,(\underline{u} \oplus \underline{\tilde{u}}) &\leq w_h\,(\underline{u}) + w_h\,(\underline{\tilde{u}}) \\ &< \frac{1}{2}d + n - d \text{ owing to (B.3),(B.2)} \\ &= n - \frac{1}{2}d \end{aligned}$$

2.

$$\begin{aligned} w_h\left(\underline{u} \oplus \tilde{\underline{u}}\right) &= n - w_h\left(\underline{u} \oplus \tilde{\underline{u}} \oplus \underline{1}\right) \\ &\geq n - \left[w_h\left(\underline{u}\right) + w_h\left(\tilde{\underline{u}} \oplus \underline{1}\right)\right] \\ &> n - \left[\frac{1}{2}d + n - d\right] \text{ owing to (B.3),(B.1)} \\ &= \frac{1}{2}d. \end{aligned}$$

Likewise, if

$$w_h\left(\underline{u} \oplus \underline{1}\right) < \frac{1}{2}d \Leftrightarrow w_h\left(\underline{u}\right) > n - \frac{1}{2}d \tag{B.4}$$

then

1.

$$\begin{aligned} w_h\left(\underline{u} \oplus \underline{1} \oplus \tilde{\underline{u}}\right) &\leq w_h\left(\underline{u} \oplus \underline{1}\right) + w_h\left(\tilde{\underline{u}}\right) \\ &< \frac{1}{2}d + n - d \text{ owing to (B.4),(B.2)} \\ &= n - \frac{1}{2}d \end{aligned}$$

2.

$$\begin{aligned} w_h\left(\underline{u} \oplus \underline{1} \oplus \tilde{\underline{u}}\right) &= n - w_h\left(\underline{u} \oplus \underline{1} \oplus \tilde{\underline{u}} \oplus \underline{1}\right) \\ &\geq n - \left[w_h\left(\underline{u} \oplus \underline{1}\right) + w_h\left(\tilde{\underline{u}} \oplus \underline{1}\right)\right] \\ &> n - \left[\frac{1}{2}d + n - d\right] \text{ owing to (B.4),(B.1)} \\ &= \frac{1}{2}d. \end{aligned}$$

Thus in either case, because of the following properties:

i. $\delta \leq \frac{1}{2}$ (this follows from the definition (3.17) of d_{univ}),

ii. The binary entropy function $h_b(\cdot)$ is monotonically increasing on $[0, \frac{1}{2})$ (see Figure B-1),

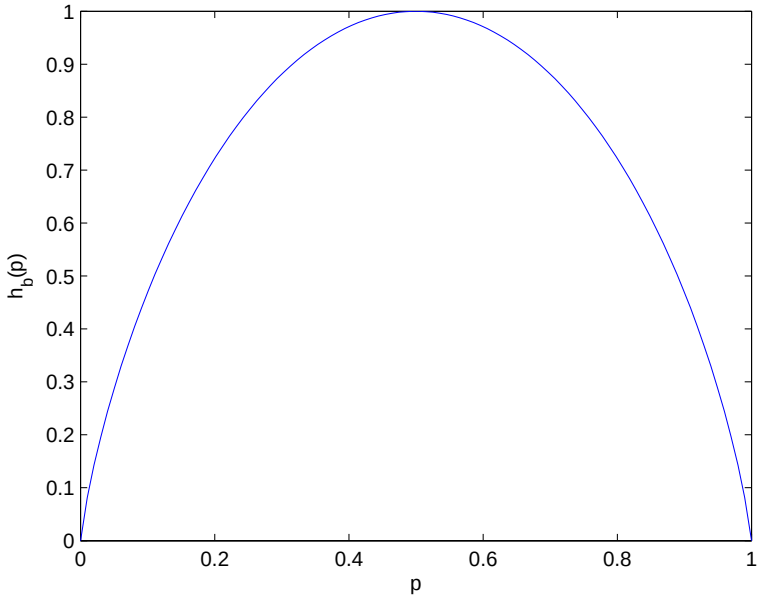


Figure B-1: The binary entropy function

iii. The binary entropy function $h_b(\cdot)$ is symmetric around $\frac{1}{2}$ (see Figure B-1),

we have that $h\left(P_{\underline{\tilde{u}}\oplus\underline{u}}\right) > h\left(P_{\underline{u}}\right)$. Thus if we define $\underline{s} = H\underline{u}$ then we have that $\underline{u}$ is the unique solution to

$$\min_{\underline{\hat{u}}\in \mathrm{Co}(H,\underline{s})} h\left(P_{\underline{\hat{u}}}\right).$$

The alternative statement in the lemma holds because the two statements are equivalent:

- $w_h\left(\underline{u}\right) < \frac{1}{2}d_{\mathrm{univ}}$ **or** $w_h\left(\underline{u}\oplus\underline{1}\right) < \frac{1}{2}d_{\mathrm{univ}}$,
- $h\left(P_{\underline{u}}\right) < h_b\left(\frac{1}{2}\delta\right)$.

This also follows from properties i.–iii. above. □