A Delay-Constrained Cross-Layer Model Using Network Coding

by

David C. Adams

B.S., Electrical Engineering

United States Air Force Academy (2012)

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

of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2014

Signature of Author ..... Signature redacted .....
Department of Electrical Engineering and Computer Science
May 9, 2014

Certified by ..... Signature redacted .....
Muriel Médard
Professor of Electrical Engineering
Thesis Supervisor

Certified by ..... Signature redacted .....
Christopher Yu
Technical Staff, Draper Laboratory
Thesis Supervisor

Certified by ..... Signature redacted .....
Jinfeng Du
Postdoctoral Associate
Technical Advisor

Accepted by ..... Signature redacted .....
Leslie A. Kolodziejski
Chair, Department Committee on Graduate Studies

The views expressed in this article are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

A Delay-Constrained Cross-Layer Model Using Network Coding

by

David C. Adams

Submitted to the Department of Electrical Engineering and Computer Science
on 21 May 2014 in partial fulfillment of the requirements for the degree of
Masters of Science in Electrical Engineering and Computer Science

Abstract

Traditionally, most packet-switched networks have only one wireless hop: the link between the end users and their access point. However, there is increasing interest in using wireless links to reach the edge of the network. Having more than one wireless link is a game changer. Network layer architecture is predicated on the assumption that the lower layers are reliable, but this comes at a high cost in terms of data rate on a band-limited, lossy wireless channel. This cost is tolerable over one underutilized link, but when the network demands high-capacity wireless links, it may be time to rethink the way the packet-switched network interacts with its underlying infrastructure.

The aim of this thesis is to provide a general model that can be used to frame a wide variety of cross-layer coding problems. We do not explicitly consider the channel code, medium access, or modulation; instead, we leverage the maturity of these fields to observe the general effect they produce on higher layers. We focus our model on applications where delay is constrained, which forces us to consider coding results in the regime where code length is non-asymptotically large. In trying to extend our analysis to multi-hop flows, we develop an analytical tool that can be useful in wider applications. This tool simplifies certain network flows to a distribution on the amount of information available to the destination; it is a step towards characterizing network information flows in the non-asymptotic regime. Finally, we will use the model to design encoding schemes, given practically-motivated constraints.

Thesis Supervisor: Muriel Médard
Title: EECS Professor

Thesis Supervisor: Christopher Y. Yu
Title: Draper Laboratory Technical Staff

Acknowledgements

First off, I would like to thank MIT and Draper collectively for giving me the chance to be here, and Muriel and Chris in particular for their patience and guidance. I also believe that without Jinfeng, I would have no idea what this thesis would look like. I'd also like to thank Chris Rose for scaring me into productivity. There are several people at the US Air Force Academy that were willing to voice their confidence in me when I applied to MIT: Col. Jeffrey Butler, LtCol. Anne Clark, and Dr. Randall Musselman. I hope I have made you all proud.

I appreciate the support of all of my friends during my time in Cambridge, but in particular I want to recognize two groups. First, I'd like to thank my lab mates in the Network Coding and Reliable Communication Group for their encouragement, consolation, and tutoring - essentially for providing me whatever support I needed each day when I walked into the office. Second, I'd like to thank my roommates for challenging me to work hard. Finally, I want to thank my family for instilling in me the values that have carried me this far. Love you, Mom, Dad, and Maggie.

Contents

List of Figures

Chapter 1

Background

Up until very recently, networks have largely been considered to be wireline structures, possibly with a wireless link comprising the last hop to the user. As a consequence, the field of wireless communications has been somewhat insulated from the work on higher layers of the network. The goal of wireless communications has simply been to provide a low error rate, usually at the cost of data rates. This cost may be prohibitive for certain applications. For example, Google and DARPA[1] both have programs to deliver service to denied areas via mobile, airborne relays. In addition to all of the inauspicious characteristics of terrestrial wireless backhaul channels, the channels connecting these mobile units are further debased by power constraints, imprecise antenna alignment, and Doppler fading. The use of wireless networks has highlighted the inefficiencies of the traditional role of wireless architecture in a larger network, and we are beginning to see work that merges the properties of noisy links with the behavior of packet-switched networks - cross-layer modeling.

Most cross-layer work thus far has focused on configuring structures at each layer to optimize network performance. Cross-layer optimization has been investigated, but generally depends heavily on the underlying model and the objective function over which the optimization occurs. It should come as no surprise, then, that since the proliferation of the idea of network coding (first proposed by Ahlswede *et al.*. in [1]) the body of work on two-layered coding covers a continuum of varied models. The differences between these models can be subtle but consequential.

For example, we may consider two papers published three years apart in two journals in the field of wireless communication, by Berger *et al.* [2] and Courtade and Wesel [3]. Both papers address the question of how best to allocate redundancy beween two coding

[1]We refer to Google's Project Loon and DARPA's Mobile Hotspot program.

layers operating over a fading channel. While [2] concludes that the optimal way to combat severe fading is to use a stronger network code, [3] concludes that optimality demands stronger channel codes when fading is severe. The discrepancy is due to differences in model construction and the objective function used in each paper and is illustrative of the difficulty of deriving general results from a cross-layer model.

In light of the difficulties involved in trying to glean insight from two layers of coding, we propose a model and set of associated tools whose generality lends itself to use in a wide range of circumstances. Using tools we tailor for this model, we envisage how to negotiate design trade-offs in a system with practically-motivated constraints; in particular we consider an end-to-end delay constraint.

1.1 Literature Review

This thesis borrows from findings in different areas of research. The references have been loosely grouped into the following categories based on their primary relevance to this work.

1.1.1 Related Work on Network Coding

From Chapter 3 onward, random linear network coding (RLNC) will be central to the analysis in this thesis, and coding at the packet level features heavily in our approach to cross-layer modeling. The work of Ahlswede *et al.* in [1] introduced the idea of network coding and proved that it could achieve network capacity by satisfying the max-flow min-cut theorem. In the wake of [1], Li *et al.* proved in [4] that linear network codes were a sufficient subclass of network codes to achieve capacity, again by showing that the max-flow was achievable using this subclass. Koetter and Médard provided a framework for analysis of linear network coding based on linear algebra in [5]. This algebraic framework offered a more succinct way of describing encoding and re-encoding operations than the routing-centered view it supplanted. Ho *et al.* in [6] further established that randomly generated linear codes were capacity-achieving with high probability.

While [1, 4, 5, 6] cast network coding as an alternative to traditional routing, Cai and Yeung, two of the authors of the original network coding paper in [1], also began early on looking at the error-correcting properties of these codes in [7]. Their observations have spurred a whole other line of investigation of network coding as a competitor to rateless erasure codes [8].

RLNC has been investigated by many to determine what, if any, quality-of-service gains it offers. We mention only a few works here. Ghaderi *et al.* in [9] and Eryilmaz *et al.* in [10] both showed that network coding (and rateless codes) reduce the number of transmissions required to broadcast data to several sources when used in conjunction with automatic repeat request (ARQ). Ghaderi *et al.* interpreted this result to mean that network coding provides increased reliability, while Eryilmaz *et al.* observed that this reliability translates into shorter delay on average. In a similar vein, Rezaee *et al.* consider in [11] a retransmission scheme where feedback in a broadcast wireless network is scheduled based on the expected time of transmission for all destinations. Heterogeneity among channels is determined to be detrimental to quality of experience, but the number of users is not found to be detrimental under homogeneous conditions.

1.1.2 Related Work on One-Hop Cross-Layer Optimization

Many works have considered the performance of RLNC when used in conjunction with channel coding, and we mention a few of them here. Although their work predates the discovery of RLNC, Barg *et al.* in [12] considers concatenated codes where the outer code is a random linear code over some large finite field. The work involves finding exponential bounds on the probability of error of these codes.

Vehkaperä and Médard in [13] propose a two-layer concatenated code framework with a total delay constraint, where error exponents for random coding, derived from [14], are used to analyze both inner and outer codes. This framework investigates a throughput-delay tradeoff, clarifies the embedding of error functions in a multi-level approach, and models the fact that data may be lost at either layer. The main result is presented as a solution to the question of optimal erasure-coding rate, but the model used is the progenitor of the one presented in this thesis.

In [15], Koller *et al.* consider RLNC operating over packets that are each protected by random error-correction code and transmitted over a binary symmetric channel (BSC) for a one-hop broadcast network. The expected number of transmissions required to decode is minimized, and it is noted that this optimization problem is not equivalent to optimizing the expected number of accurate bits per transmission.

Berger *et al.* in [2] focus on a flat block-fading channel, where perfect channel decoding is assumed for all rates below capacity and perfect packet-erasure coding is assumed when a fixed overhead requirement is met. The tradeoff between spectral efficiency and end-to-end probability of error is explored; one particularly useful insight is the investigation of this

trade-off with few choices of physical layer transmission modes.

Courtade and Wesel in [3] also assume perfect channel and erasure coding as in [2], but consider both fast and slow flat block-fading with a delay constraint. This paper employs a Gaussian approximation of end-to-end outage probability, and optimizes code rates at each layer for power consumption under end-to-end rate and reliability constraints. Numerical solutions indicate that allocating some redundancy to a rateless erasure code improves performance, but that the optimal erasure code rate is bounded away from zero as the transmission time increases.

Swapna *et al.* in [16] work on RLNC over K packets broadcasting to n users over independent time-correlated erasure channels. They conclude that K must scale as $O(\log(n))$ to ensure non-zero throughput. Under such conditions, the mean and variance of total transmission time depend on channel correlation.

A slightly different perspective is considered by Wu and Jindal in [17]. Instead of erasure correction, the authors use ARQ in conjunction with error correction over a block-fading channel. The authors optimize for goodput, and find that the resulting error rate for most realistic operating conditions is on the order of 10%. While this is certainly a different setup than is considered here, the results are very germane to those in Chapter 5, and indicate that physical layer error tolerance can be relaxed when used in conjunction with a variety of packet-level erasure correction measures.

1.1.3 Work on Network Coding for Larger Networks

While [12, 13, 15, 17] elucidate different facets of the cross-layer problem, they all consider only single-link networks where generalization to larger topologies is nontrivial. Taking that one step further, [2, 3, 16] all consider a wireless broadcast, but the wireless component of the network is still only a single hop. We also recognize some works that consist of multiple hops, albiet not necessarily wireless ones.

Dana *et al.* in [18] considered the use of linear network coding in networks consisting of wirelessly connected nodes, assuming the destination node has side information detailing the network link on which each packet loss occurs. Constructing a network model based on correlated erasure links and broadcast from one node to its neighbors, the authors found the capacity of such networks and showed that linear network codes achieved it.

Lun *et al.* in [19] propose a framework to translate a lossy unicast or multicast network into a lossless packet network with only point-to-point arcs (for wireline) and hyper-arcs (for wireless). Assuming that the number of packets is large and that the arrivals of packets

at each node are independent Poisson processes, the probability of RLNC decoding error is characterized by the delay, rate, and the network capacity. The average throughput of each individual link becomes the sole figure of merit characterizing system performance, but this result requires that the number of packets be large and packet arrivals form a Poisson process; additionally, delay is based entirely on the depth of the network, independently of propagation time. Unlike [18], no side information is required to achieve capacity. Lower layers are not considered in [19], but we will rely heavily on its results in Chapter 4 as we consider coding over several links.

While [19] does not consider the practical constraint of buffer size at intermediate nodes, Haeupler and Médard show in [20] that RLNC is still capacity-achieving even if intermediate nodes may only store one coded packet. As in [19], this result only describes the behavior of the long-term average. Xiao *et al.* in [21] investigate the delay in packet erasure networks where RLNC is used in a rateless fashion, and the delay is optimized based on the tradeoff between codeword lengths on physical layer and on network layer. The most interesting aspect of [21] for this thesis is the technique of bonding together all of the links in the minimum cut and modeling them as one link with a resultant erasure rate. While we will also try to simplify networks in Chapter 4, our approach will consider more than the minimum cut and describe more than the average network behavior.

1.2 Outline and Contributions

In Chapter 2, we introduce and justify our two-layer network model, defining language and notation that will be used throughout this thesis. A table of all the parameters is provided, along with the performance metrics derived from the model. Chapter 3 presents analytical expressions to approximate the error probability of our network code. In particular, we focus on a single link when there is some finite constraint on delay. The end of the chapter briefly discusses the performance of each of these approximations. In Chapter 4, we extend our error analysis to more complicated network topologies and introduce a method of reducing simple but nontrivial networks to smaller equivalent structures. Finally, in Chapter 5, we study how to design a two-layer coding scheme using our model. We motivate an example setup with limited physical-layer configurations and maximize over our choice of two code rates for both a single link and a relay network. The maximum-achieving configurations are analyzed to gain intuition about the role each code plays in error mitigation. The results indicate that it can be preferable to allow a high rate of physical-layer decoding error when

erasure-correction is used.

The focal point of this work is the model itself. To the best of the author's knowledge, it is not found in any other work; its closest relative is probably in [13], from which it was adapted. The chief insight of the model contained in this thesis is presented in Chapter 5, where we can see how the addition of erasure coding affects the performance of the network and the operating point of error-correction codes.

In addition to these chief contributions, the network reduction procedures detailed in Chapter 4, which are developed here from first principles, may be of interest in system design or further research into packet-erasure networks. The original motivation was to combine network links like components of a circuit to reach a sort of "Thévenin equivalent" network; perhaps future work will be able to complete this task for arbitrary networks.

A Word About Notation

We will be dealing with a large number of parameters in the following chapters. There will be a few notational conventions that should help clarify things. Physical layer (PHY) parameters will be denoted by Greek letters, which may be a jarring departure from traditional coding notation involving k, n, and R. Capital letters will be used for random variables; lower case versions of these letters will denote realizations of the random variables. Bold capitals will be used for matrices, and any character with an arrow over it denotes a vector. Sets and metrics will use bold or script fonts. All of this is done in an attempt to avoid using multiple subscripts when analyzing time-indexed random variables occurring at several links simultaneously.

Chapter 2

The Two-Layer Network Model

Part of the difficulty in developing useful cross-layer models is that we cannot assume that the layers interface as cleanly as we can when considering one layer in isolation. We begin to address this problem by noting that network properties generally propagate up the stack, while the lowest layers are often agnostic of what is happening above them. It makes sense, then, for us to start at the PHY layer before building up from there.

We generally adhere to standard information theory vocabulary at the PHY layer. Our encoder accepts a certain quantity of information, called the *message*, from the higher layer, and encodes this into a *codeword*. The number of times the channel is used to transmit a codeword is called the *block length*. This gives us a concept of a *PHY rate*[1] as the amount of information transmitted in one channel use.

The primary unit of information at the network (NET) layer is the *packet*. We model packets as elements of some large finite field; to all operations at this layer, packets should appear atomic. Some percentage of each packet may be required for a *header*, containing overhead like addressing or decoding information; we call the rest of the packet the *payload*. Each segment of data over which the NET encoding operation is performed is called a *generation*; a generation is analogous to the message at the PHY layer. The NET encoder adds redundancy to produce a number of *coded packets* larger than the generation size in packets. We measure redundancy at this level with the *NET rate*, a dimensionless quantity comparing the generation size in packets to the number of coded packets output. The NET rate only holds any real meaning over one link, since this comparison changes from link to link with the number of coded packets.

[1] Or perhaps spectral efficiency, as our notion of the PHY-layer includes both forward error correction and modulation.

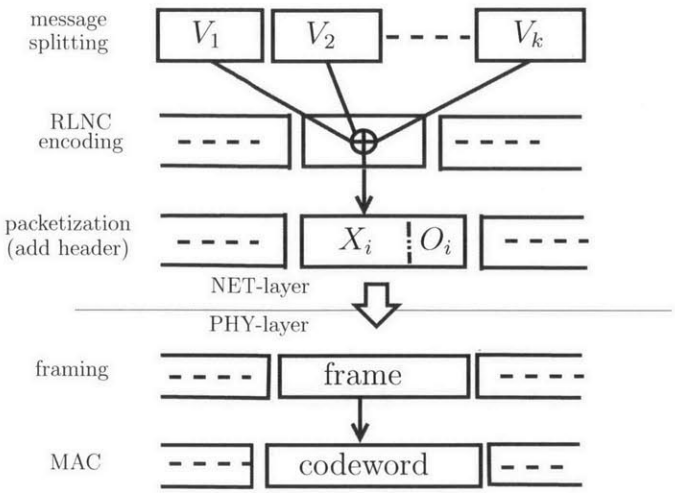

Figure 2.1: Two-layer encoding process. Each generation of data is segmented into k pieces, which are coded using an erasure-correction code into packets. Headers are appended to the packets, then the packets are framed and handed off to the medium access control (MAC), which will perform error correction and mapping to a signal constellation.

At the boundary of the two layers of each link a certain number of packets are *framed* into a message and passed to the PHY to be coded into a single codeword. The number of packets framed by a codeword does not necessarily have to be an integer.

2.1 Model Parameters

We now assign notation to the quantities described above. As before, we commence with the lower layer, then discuss the higher layer as a separate model entirely, and finally join the two.

2.1.1 Physical Layer

We allow for a general treatment of PHY codes and modulation schemes, using the following definitions:

- $\mathcal{H}$: the channel model being used; a constellation of output symbol distributions conditional on input
- $\mathcal{M}$: the output alphabet of the encoder
- ν: the number of channel uses comprising one codeword; block length

- ρ: the rate of the encoder in bits per channel use.

If we denote the encoding operation with the function $f(\cdot)$, then we have a mapping for our encoder,

$$f(\cdot): \mathbb{F}_2^{\nu\rho} \mapsto \mathcal{M}^{\nu}, \tag{2.1}$$

and a similar map for decoding, featuring the addition of the singleton set $\{\mathrm{e}\}$, indicating an erasure,

$$f^{-1}(\cdot): \mathcal{M}^{\nu} \mapsto \mathbb{F}_2^{\nu\rho} \cup \{\mathrm{e}\}. \tag{2.2}$$

2.1.2 Network Layer

At the NET layer, our encoder is treated with equal generality, captured by five parameters:

- ℓ: length of packet in bits
- η: the percentage of the packet occupied by the payload
- k: generation size in packets
- n: number of coded packets generated for each generation
- ξ: probability of packet erasure.

As with the PHY layer, the encoder is described with an unspecified function $g(\cdot)$,

$$g(\cdot): \left(\mathbb{F}_2^{\ell}\right)^{k} \mapsto \left(\mathbb{F}_2^{\ell}\right)^{n}, \tag{2.3}$$

or, in the case where packets are transmitted on two links,

$$g(\cdot): \left(\mathbb{F}_2^{\ell}\right)^{k} \mapsto \left\{(\mathbb{F}_2^{\ell})^{n_1}, (\mathbb{F}_2^{\ell})^{n_2}\right\}. \tag{2.4}$$

If our destination node pools coded packets from all links in the set $\mathcal{I}$, then our decoding operation would be

$$g^{-1}(\cdot): \left(\mathbb{F}_2^{\ell} \cup \{\mathrm{e}\}\right)^{\sum_{i\in\mathcal{I}} n_i} \mapsto \left(\mathbb{F}_2^{\ell}\right)^{k}. \tag{2.5}$$

It should briefly be mentioned that ξ is not only a function of the PHY-layer settings, but also may be dependent upon the size of the packets themselves. For example, we may be concerned with a multiple access model where longer packets translate to a higher incidence of collision. We will discuss the structure of ξ in greater detail later.

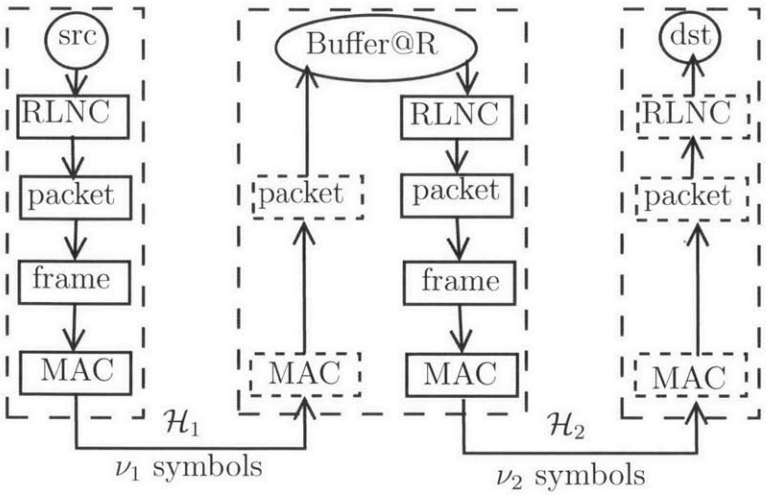

Figure 2.2: Flow chart of a two-hop network where the delay constraint restricts the number of symbols sent on link i to ν_i over the corresponding channel $\mathcal{H}_i$.

2.1.3 Code Coupling

We require that the PHY be capable of marking erroneously decoded packets as erasures, using a cyclic redundancy check or some other error-detection method. Coupling the encoding functions given in (2.1) and (2.3), we see that the NET encoder simply spits out a string of bits that then feeds the PHY encoder. The bit strings may be different lengths, however, so we compose our encoding operations by indicating the correct number of codewords needed,

$$W \mapsto (\mathbb{F}_2^{\ell})^k \overset{g(\cdot)}{\mapsto} (\mathbb{F}_2^{\ell})^n \overset{\phi(\cdot)}{\mapsto} (\mathbb{F}_2^{\nu\rho})^{\frac{n\ell}{\nu\rho}} \overset{f(\cdot)}{\mapsto} (\mathcal{M}^{\nu})^{\frac{n\ell}{\nu\rho}}, \tag{2.6}$$

where $\phi(\cdot)$ is an isomorphism that repartitions the bit string passed from the NET encoder to the PHY encoder. We require that $\frac{n\ell}{\nu\rho}$ be an integer in (2.6), but we could also substitute $\left\lceil \frac{n\ell}{\nu\rho} \right\rceil$ and append padding bits to the output of the NET encoder.[2] Furthermore, we can generalize the mapping in (2.6) to reflect heterogeneity among point-to-point links in the network. If $\mathcal{I}$ is the set of links over which the source node can transmit, we can say that

$$W \mapsto (\mathbb{F}_2^{\ell})^k \overset{g(\cdot)}{\mapsto} (\mathbb{F}_2^{\ell})^{\sum_{i\in\mathcal{I}} n_i} \overset{\phi(\cdot)}{\mapsto} \left\{ (\mathbb{F}_2^{\nu_i\rho_i})^{\frac{n_i\ell}{\nu_i\rho_i}} \overset{f_i(\cdot)}{\mapsto} (\mathcal{M}_i^{\nu_i})^{\frac{n_i\ell}{\nu_i\rho_i}}, \quad i \in \mathcal{I} \right\}. \tag{2.7}$$

Merging the two models at either layer provides a nearly complete picture, but there are some refinements needed to facilitate analysis. First, let us define a parameter that captures packet loss due to NET-layer effects like collision, congestion, buffer overflows, etc.; we denote

[2] In this case, $\phi(\cdot)$ can be a one-to-many mapping.

this p_t.[3] Also, we have ℓ bits per packet and $\nu\rho$ bits per codeword (or per message), so we can define a framing parameter,

$$\alpha \triangleq \frac{\nu\rho}{\ell} \text{ packets per codeword} \tag{2.8}$$

to characterize succinctly the virtual channel that the NET-layer code sees. In the next section, we will briefly consider the effect of α on error propagation.

As the title of this thesis suggests, our primary concern is to study scenarios where the delay is constrained, meaning that the destination must attempt to decode each generation after a set amount of time. This does not mean that we constrain the average amount of time it takes to decode each generation. Given a real system with a set bandwidth and signal constellation, any delay constraint given to us in units of time could be converted into a constraint on the number of channel uses our PHY code is permitted per generation. If this channel-use constraint is τ, then from our composite mapping in (2.6), we derive that $n\ell/\rho \leq \tau$. This relationship between n, ℓ, and ρ leads us to make two observations about the interaction between these parameters. First, the choice of ℓ and η should be constant across the network; this will keep us from needing to decode both layers at every node, as shown in Figure 2.2. Second, our upper bound on n for any link grows with ρ; there is not necessarily a trade-off of redundancy between the two layers because k can be set independently of n.[4]

Code coupling is the aspect of the model that truly makes it cross-layer. In Chapters 3 and 4, we will look at links that assume 1-to-1 code coupling, but a richer set of interactions will be used in our cross-layer design in Chapter 5. Summarizing, we have added the following parameters to flesh out our delay-constrained model:

- α: packet framing parameter
- τ: end-to-end delay constraint in channel uses
- p_t: rate of packet loss due to NET-layer effects.

2.2 Performance Metrics

We now consider the question of characterizing the probability of decoding error at each layer as a function of the parameters defined in Section 2.1. Additionally, we define goodput, which

[3] p_t is inherited from [13]

[4] In this regard, our work differs from previous works that formulate an optimization problem based on allocating a fixed amount of redundancy across layers, e.g. [3].

serves as a measure of usable data and inherently reflects the trade-off between NET-layer rate and probability of error.

2.2.1 Layered Error Probability

Treating each layer in isolation, we consider the problem of finding the probability of error in terms of that layer's parameters. If $W \in \mathbb{F}_2^{\nu\rho}$ is the message given to the PHY-layer encoder, and $\hat{W} \in \mathbb{F}_2^{\nu\rho} \cup \{\mathrm{e}\}$ is the estimate of the message after decoding, our probability of error is given by

$$\mathbb{P}\left\{\hat{W} \neq W\right\} = \mathbb{P}\left\{\hat{W} = \mathrm{e}\right\} = p_e\left(\nu, \rho, \mathcal{H}\right), \tag{2.9}$$

[5]

where $p_e\left(\nu, \rho, \mathcal{H}\right)$ is a deterministic function that is specified by the exact choice of $f(\cdot)$ and $f^{-1}(\cdot)$, and may in fact be unknown.

Likewise, at the NET layer we have a generation of data $V \in (\mathbb{F}_2^{\ell})^k$ and corresponding decoder estimate $\hat{V} \in \left(\mathbb{F}_2^{\ell} \cup \{\mathrm{e}\}\right)^k$. If we denote the set of edges in the flow between our source and destination node $\mathcal{E}$, then let $\vec{n} = \langle n_1, \ldots, n_{|\mathcal{E}|}\rangle$ and $\vec{\xi} = \langle \xi_1, \ldots, \xi_{|\mathcal{E}|}\rangle$ be vector representations of the parameters of the links in $\mathcal{E}$, probability of end-to-end error is thus,

$$\mathbb{P}\left\{\hat{V} \neq V\right\} = P_e\left(k, \vec{n}, \vec{\xi}\right). \tag{2.10}$$

Again, we use an unspecified deterministic function that depends on choice of $g(\cdot)$ and $g^{-1}(\cdot)$. The parameter ξ is difficult to manipulate, however, as ξ is actually a function itself. PHY error probability, p_t, and packet length will determine ξ:

$$\xi \triangleq \mathbb{P}\left\{\text{Packet erasure}\right\} = P_p\left(\ell, p_t, p_e\left(\nu, \rho, \mathcal{H}\right)\right). \tag{2.11}$$

Substituting (2.11) into (2.10), we obtain a function describing error over just a single-link network:

$$\mathbb{P}\left\{\hat{V} \neq V\right\} = P_e\left(k, n, P_p\left(\ell, p_t, p_e\left(\nu, \rho, \mathcal{H}\right)\right)\right). \tag{2.12}$$

Although (2.12) only reflects the dynamics of encoding over a single link, we may represent the error probability as a function of many links, each with a different PHY code - in fact, this will be covered in detail in Chapter 4. For now, we restrict ourselves to the notation in (2.12).

Note that choice of framing parameter, α, can have an impact on the probability of packet

[5]The first equality holds only with perfect error detection (usually via cyclic redundancy check).

erasure, as shown in Figure 2.3. In the simplest case, where $\alpha = 1$, one packet fits into one codeword, and uncorrelated decoding errors at the PHY translate into a discrete, memoryless virtual channel for the NET layer. When α is an integer greater than 1, a decoding error at the PHY layer results in the erasure of all α coded packets contained within that codeword. This integer case is nearly as analytically simple as the 1-to-1 case, but the virtual packet erasure channel will have a particular kind of memory. This is shown in Figure 2.3a.

If α is a non-integer, we intuit that this will somewhat raise our rate of packet loss above the baseline derived from $p_e(\nu, \rho, \mathcal{H})$. To see this, note in Figure 2.3b that a packet is divided between two codewords, one of which has been erroneously decoded. There is no error correction between the two layers, so the entire packet must be marked as an erasure.

Likewise, setting $\alpha < 1$ will also raise our rate of packet erasure. If a packet spans multiple codewords, as in Figure 2.3c, then any one erroneously decoded codeword will corrupt the entire packet. For encoders that are decoupled, then, it seems sensible to set $\alpha \geq 1$; additionally, in a network with a set $\mathcal{E}$ of links where each link $i \in \mathcal{E}$ may have a different PHY code, it is advisable that $\ell \leq \min_i\{\nu_i \rho_i\}$. This is not often the case in practice, because η can be nearly 1 when the packets are large.

It follows from our discussion of α that when $\alpha \neq 1$, the probability that a particular packet is erased may depend on the erasure of other packets or vary from packet to packet, even when $\mathcal{H}$ is discrete and memoryless. Thus, using ξ in (2.11) is actually abuse of notation, as a scalar may not fully characterize the distribution of packet arrivals. However, we will often construe our virtual erasure channel as memoryless and time-invariant, so a scalar ξ for each link will suffice.

2.2.2 Goodput

To examine the trade-off between throughput (the amount of data per generation) and error, we consider the quantity of information to be successfully decoded at the destination:

$$\Gamma \triangleq \mathbb{E}\left[\text{bits of information received per generation}\right]. \tag{2.13}$$

In the event of successful decoding, the destination receives k packets containing $\eta\ell$ bits of information each. Let us assume that an incorrectly decoded generation of data provides the destination with no data. Then,

$$\Gamma = k\eta\ell\left(1 - \mathbb{P}\left\{\hat{V} \neq V\right\}\right) = k\eta\ell\left(1 - P_e\left(k, n, P_p\left(\ell, p_t, p_e\left(\nu, \rho, \mathcal{H}\right)\right)\right)\right). \tag{2.14}$$

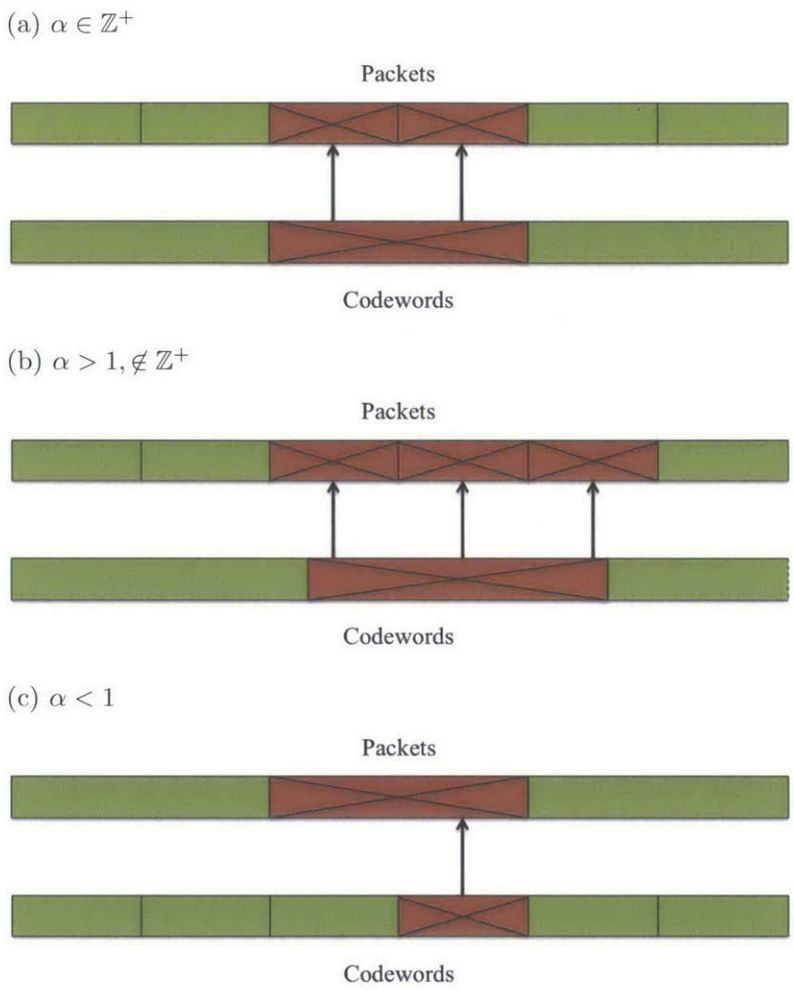

Figure 2.3: Illustrations of error propagation from the PHY layer to the NET layer for different values of the framing parameter, α.

Although there are many possible names for Γ we could use, such as "effective rate", and "throughput", we opt to use the term *goodput*. The expression in (2.14) illustrates some of the intricacies of cross-layer optimization. Γ contains no fewer than eight parameters and conveys how much information is decoded per generation, but it is still unclear what relevance a generation has to more traditional measures of delay like time or channel uses. When an end-to-end delay constraint τ is imposed, however, we could use $\frac{1}{\tau}\Gamma$ as a measure of average information delivery rate.

Note that we are considering a lower-bound case by assuming that no information is decoded when $V \neq \hat{V}$. Disregarding the idea of partial data reception, however, permites us to avoid some of the difficulties that come from evaluating the expectation in (2.13) when

$f(\cdot)$	PHY encoding operation
$g(\cdot)$	NET encoding operation
$\mathcal{H}$	PHY channel model; distribution of output symbols, conditional on inputs
k	number of packets before encoding
ℓ	packet length
$\mathcal{M}$	PHY encoder output alphabet
n	number of packets after encoding
p_t	probability of packet erasure not due to PHY-layer decoding
α	packet loading ratio
η	percent of packet that is payload
ν	block length of PHY code
ρ	rate of PHY code
ξ	probability of packet erasure

Table 2.1: Global list of parameters for two-layer encoding scheme.

the receiver stores information between frames.[6]

2.3 Chapter Summary

To recapitulate, we have considered a general model of a two-layer digital communication system. This model is skeletal; we build an outline based on first principles. The abstraction of the functions for probability of error enable this model to be a basis upon which different coding schemes or optimization strategies can be compared. For convenience, a list of all of the parameters defined is given in Table 2.1, and the placeholders for different error events are given in Table 2.2.

In particular, the embedded functions in (2.12) indicate that, given a set of constraints and an objective function, designing a two-layer system is a far different process than single-layer optimization. For example, if we choose to limit our total channel uses and decide to minimize our overall probability of error P_e, our selection of ρ is difficult: increasing ρ increases n, which lowers P_e, but increasing ρ also increases p_e, which in turn increases P_e.

This model may be reduced to a more recognizable model at either layer. For example, let us select parameters to reduce our framework to a standard channel coding problem over

[6]Sometimes, average information rate is computed as the amount of information in a generation divided by the expect amount of time until decoding is possible, which is not necessarily equal to (2.13). For example, [15].

a binary symmetric channel. Let $\mathcal{M} = \{0, 1\}$ (or, equivalently, $\mathbb{F}_2$). For $a, b \in \mathcal{M}$, define the channel,

$$\mathcal{H}: \; P_{B|A}(b|a) = \begin{cases} 1-\delta, & a = b \\ \delta, & \text{o/w} \end{cases}.$$

Let $p_t = 0$, $\alpha = 1$, $\ell = \nu\rho$, $k = 1$, $n = 1$, $\eta = 1$, and $g(x) = x$. All we have left are the typical PHY parameters: rate, blocklength, and encoding operation. Then the reduced coding operation can be represented with the mapping,

$$V \mapsto \mathbb{F}_2^{\nu\rho} \overset{f(\cdot)}{\mapsto} \mathbb{F}_2^{\nu}.$$

Also, our error functions are now $P_e(1, 1, P_p(\cdot)) = P_p(\nu\rho, 0, p_e(\cdot)) = p_e(\nu, \rho, \mathcal{H})$, so all that is left is the PHY channel-coding problem.

We can also adjust our model to only represent the network coding problem. Let $n = k + \Delta$, $\mathcal{M} = \{0, 1\}$, $\rho = 1$, $\nu = n\ell$, $f(x) = x$, operating over a perfect channel $\mathcal{H}$. Let p_t reflect the steady-state probability of packet erasure, then let Δ be the amount of redundancy in packets. The encoding operation is now described by

$$V \mapsto (\mathbb{F}_2^{\ell})^k \overset{g(\cdot)}{\mapsto} (\mathbb{F}_2^{\ell})^{k+\Delta},$$

which represents the network coding construction. These two example cases show that our new model is compatible with single-layer applications.

$p_e(\nu, \rho, \mathcal{H})$	probability of PHY decoding error
$P_p(\ell, p_t, p_e(\cdot))$	probability of packet erasure
$P_e(k, n, P_p(\cdot))$	probability of NET decoding error

Table 2.2: Error functions for the two-layer encoding scheme.

Chapter 3

Error Approximations for Random Linear Network Coding

In Chapter 2, we outlined a model for a single-link network with two layers of coding, treating probability of error as generally as we could. In this chapter, we consider exclusively the NET layer probability of error while using a random linear network code (RLNC) over a virtual channel with independent and identically distributed (iid) packet erasures. In particular, we derive functions to approximate P_e over a single link using techniques from combinatorics, large deviations theory, the central limit theorem (CLT), and recent results on channel coding in the finite block-length regime. We will briefly investigate whether large-deviation or CLT-based approaches produce better approximations of true probability of error in different rate regions, finding that the CLT produces a better approximation around capacity, but not at low rates. Perhaps surprisingly, we show that the finite blocklength result can be derived quickly by combining the CLT with the basic structure of a RLNC.

3.1 Combinatorial Results for Random Linear Network Coding

Before exploring our error approximations, a brief exposition of RLNC will serve to highlight useful properties of the code. Although RLNC is general enough to model multiple flows of information occurring within the same network, we limit our scope to a single flow. Our treatment of the material is descended primarily from [19].

3.1.1 Introduction of Random Linear Network Coding

At the source node, we begin with k messages, each represented as a vector of length ℓ_q over field $\mathbb{F}_q$, the set $\{v_1, v_2, \ldots, v_K\}$. Since, as presented in Chapter 2, a packet consists of ℓ bits, and because an element in $\mathbb{F}_q$ can be represented in $\log_2 q$ bits, we see that,

$$q^{\ell_q} \geq 2^{\ell}, \text{ or } \ell_q = \left\lceil \frac{\ell}{\log_2 q} \right\rceil. \tag{3.1}$$

To generate a coded packet, k coefficients $\{a_1, a_2, \ldots, a_k\}$ are drawn according to a uniform distribution over $\mathbb{F}_q$, and the coded packet is constructed by a linear combination of $\{v_1, v_2, \ldots, v_k\}$:

$$x = a_1 v_1 + a_2 v_2 + \cdots + a_k v_k. \tag{3.2}$$

If we think of each packet v_i as the i^{th} column in a $l \times k$ matrix over $\mathbb{F}_q$, which we denote as $\mathbf{V}$, then we can think of the construction of n coded packets as the linear operation,

$$\mathbf{VG} = \mathbf{X}, \tag{3.3}$$

where $\mathbf{G}$ is a $k \times n$ random matrix over $\mathbb{F}_q$ with each element drawn iid from a uniform distribution, so that $\mathbf{X}$ is a horizontal concatenation of the n coded packets, $x_1, x_2, \ldots, x_n$.

Across the link, the coded packets that have not been erased are received, giving a subset of $m \leq n$ packets from $\{x_1, \ldots, x_n\}$. We assume that all of the coefficients are known at the receiver, either because they were sent with the packet, or because we used a pseudorandom number generator to make them, and the seed is known at the receiver.[1] Then we can replace $\mathbf{G}$ in (3.3) with $\mathbf{G}'$ and $\mathbf{X}$ with $\mathbf{X}'$, where the prime indicates that erased payloads (in $\mathbf{X}'$) and encoding vectors (in $\mathbf{G}'$) are replaced with 0. If $\text{rank}(\mathbf{G}') = k$, we are able to recover $\mathbf{V}$ through Gaussian elimination. If a coded packet increases the rank of $\mathbf{G}'$, meaning that the coding vector of the packet is linearly independent of the coding vectors already seen by the destination, then we may refer to the packet as a *degree of freedom* (DoF).

In the sections that follow, we will not distinguish between different types of erasures (i.e. whether they originate from PHY decoding or NET-layer effects), using a virtual channel with iid packet erasures of probability ξ and the property that

$$\mathbb{P}\left\{\hat{\mathbf{V}} \neq \mathbf{V}\right\} = \mathbb{P}\left\{\text{rank}(\mathbf{G}') < k\right\} \tag{3.4}$$

[1] Using the pseudorandom number generator seed is really only feasible over a point-to-point connection.

to derive explicit expressions for $P_e(k, n, \xi)$. In general, the assumption of an iid erasure channel is inaccurate for a number of reasons. An iid wireless channel is already an idealized model, but even if we make that assumption, the choice of α may remove the independence of packet erasures as discussed in Section 2.2.1. Additionally, NET layer erasures resulting from buffer overflow or congestion are not generally iid, because it takes some time for a buffer to empty or traffic flows to reroute.

3.1.2 Combinatorial Results for Error

An exact expression for probability of error of the code described above has been given in previous works (for example, [19]), but we find it helpful to include the derivation here as well, as our arguments in this and later chapters will be contingent upon the structure of RLNC. Let S be the number of packets to be successfully received at the end of the link, and let D be the number of DoF available at the decoder. Note that $D = \text{rank}(\mathbf{G}')$.

Because our channel is iid, S follows a simple binomial distribution:

$$\mathbb{P}\{S = s\} = \binom{n}{s}(1-\xi)^s \xi^{n-s}, \quad s \in \{0, 1, \ldots, n\}. \tag{3.5}$$

Given knowledge that $S = s$, we know that our $\mathbf{G}'$ has $n - s$ columns that are all zero; thus, we only need to find the probability that a $k \times s$ submatrix of $\mathbf{G}'$ has rank k. We can think of this as the probability that each row of that submatrix is linearly independent of the rows above it:

$$\mathbb{P}\{D = k | S = s\} = \begin{cases} \prod_{i=0}^{k-1}(1 - q^{i-s}), & s \geq k \\ 0 & s < k \end{cases}, \tag{3.6}$$

so that

$$\begin{aligned} \mathbb{P}\{D = k\} &= \sum_{s=k}^{n} \mathbb{P}\{S = s\}\mathbb{P}\{D = k | S = s\} \\ &= \sum_{s=k}^{n} \binom{n}{s}(1-\xi)^s \xi^{n-s} \prod_{i=0}^{k-1}(1 - q^{i-s}). \end{aligned} \tag{3.7}$$

This formulation is exact, but it is also cumbersome to compute. We can relax it a little by applying an upper bound on probability of success given by Liva, Paolini, and Chiani in

[22]. The result from their paper states that,

$$1 - \frac{1}{q-1}q^{k-s} < \prod_{i=0}^{k-1}(1-q^{i-s}) \leq 1 - q^{k-s-1}. \tag{3.8}$$

We apply the second inequality to obtain a lower bound for probability of error,

$$P_e = 1 - \mathbb{P}\{D = k\} \geq 1 - \sum_{s=k}^{n} \binom{n}{s} (1-\xi)^s \xi^{n-s} \left(1 - q^{k-s-1}\right). \tag{3.9}$$

This lower bound is tight when q is large, which can be seen by taking the ratio of the R.H.S of (3.8) to the L.H.S:

$$\frac{1 - q^{k-s-1}}{1 - \frac{1}{q-1}q^{k-s}} = \frac{q^{-k+s+1} - 1}{q^{-k+s+1} - \frac{q}{q-1}}. \tag{3.10}$$

This will be the standard against which we compare our other approximations, none of which depend explicitly on q. Figure 3.1 indicates the scale of the difference between the upper and lower bounds on P_e given in (3.8).

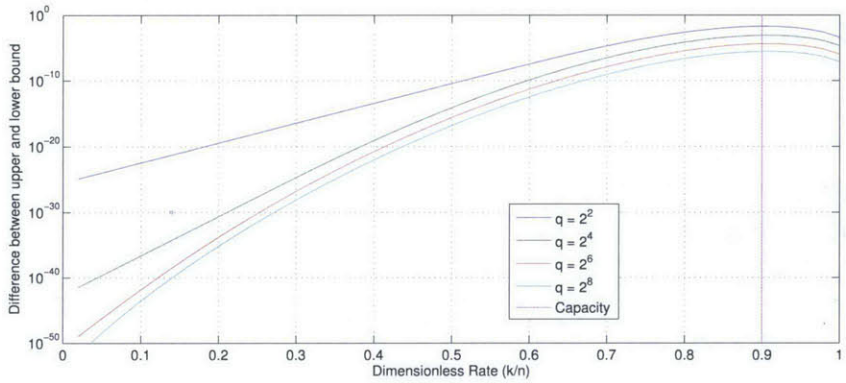

Figure 3.1: Difference between upper and lower bounds on P_e from (3.8) where $n = 50$ and $\xi = 0.1$; for $q = 2^8$, the maximum value is 2.8×10^{-6}.

3.2 Large Deviations Results

To apply well-known results from the theory of large deviations, we begin with the same model as in our combinatorial approach, but make some simplifications. Previously, we were concerned with whether the span of the coding vectors received at the destination was equal to $\mathbb{F}_q^k$; now we assume that any k coding vectors selected at the buffer of our destination will

be linearly independent, so that any k or more coded packets will be sufficient to decode, i.e. that $\mathbb{P}\{D = S\} = 1$. We justify this assumption by recalling the L.H.S of (3.8), which lower-bounds $\mathbb{P}\{D = k|S = s\}$ and is very close to 1 when q is large or s is much larger than k.

We start by tracking the failure of each packet to cross the link. Let

$$X_i \triangleq \mathbb{1}\{\text{packet } i \text{ is erased}\}, \tag{3.11}$$

so that $\{X_i,\, 1 \leq i \leq n\}$ is an iid random process where each X has the Bernoulli distribution

$$p_X(x) = \begin{cases} \xi & x = 1 \\ 1 - \xi & x = 0 \end{cases}. \tag{3.12}$$

Then, the number of packets received is $n - \sum_{i=1}^{n} X_i$. Since we assume that any k packets will allow us to decode,

$$P_e = \mathbb{P}\left\{n - \sum_{i=1}^{n} X_i < k\right\} = \mathbb{P}\left\{\frac{1}{n}\sum_{i=1}^{n} X_i > \frac{n-k}{n}\right\}. \tag{3.13}$$

The standard Large Deviations Theory result is that for any sequence of iid random variables $\{Y_i,\, i \geq 1\}$,

$$\mathbb{P}\left\{\frac{1}{n}\sum_{i=1}^{n} Y_i \geq \gamma\right\} = e^{-nE(\gamma)+o(n)}. \tag{3.14}$$

The definition of the exponent $E[\gamma]$ depends on the value of γ relative to the support of Y. Since the sum of n realizations of Y cannot exceed the sum of n of the maximum possible realization of Y,

$$E(\gamma) = \begin{cases} 0 & \gamma < \mathbb{E}[Y] \\ \psi^*(\gamma) & \mathbb{E}[Y] \leq \gamma < \sup\{y : p_Y(y) > 0\} \\ \text{Undef.} & \text{o/w} \end{cases}, \tag{3.15}$$

where $\psi_Y(\lambda)$ is the cumulant generating function (CGF) of Y, defined as

$$\psi_Y(\lambda) \triangleq \log \mathbb{E}\left[e^{\lambda Y}\right] \tag{3.16}$$

and $\psi_Y^*(\gamma)$ is the convex conjugate of the CGF, defined as

$$\psi_Y^*(\gamma) \triangleq \sup_{\lambda \in \mathbb{R}} \lambda\gamma - \psi_Y(\lambda). \tag{3.17}$$

Now we seek to apply (3.14) to the sequence $\{X_i\}$ to find an approximation for probability of error. We see from (3.13) that we should set γ to $\frac{n-k}{n}$. The convex closure of the support of X is $[0,1]$, so $E\left(\frac{n-k}{k}\right)$ is defined when $0 \leq \frac{n-k}{n} < 1$; it is unknown in the trivial case when $k = 0$, but when $k = n$,

$$\frac{n-k}{k} = 0 \leq \mathbb{E}[X],$$

so $E\left(\frac{n-k}{k}\right) = 0$. The next step is then to compute $\psi_X^*\left(\frac{n-k}{n}\right)$ for $0 < \frac{n-k}{n} < 1$.

We start by finding the CGF of X, $\psi_X(\lambda)$; also note that $\exp(\psi_X(\lambda))$ yields the moment generating function (MGF) of X:

$$\psi_X(\lambda) = \log \mathbb{E}\left[e^{\lambda X}\right] = \log\left(\xi e^{\lambda} + 1 - \xi\right). \tag{3.18}$$

To find the supremum indicated in (3.17), we need the derivative of the CGF. Then,

$$\begin{aligned}
\psi_X' &= \frac{\partial}{\partial \lambda} \log \mathbb{E}\left[e^{\lambda X}\right] \\
&= \frac{1}{\mathbb{E}\left[e^{\lambda X}\right]} \frac{\partial}{\partial \lambda} \mathbb{E}\left[e^{\lambda X}\right] \\
&= \frac{1}{\mathbb{E}\left[e^{\lambda X}\right]} \mathbb{E}\left[\frac{\partial}{\partial \lambda} e^{\lambda X}\right] \\
&= \frac{\mathbb{E}\left[X e^{\lambda X}\right]}{\mathbb{E}\left[e^{\lambda X}\right]} \\
&= \frac{\xi e^{\lambda}}{\xi e^{\lambda} + 1 - \xi},
\end{aligned} \tag{3.19}$$

where the third equality follows from the linearity of the expectation operator. Then our

supremum-achieving value λ^* is found as follows:

$$\begin{aligned}
0 &= \frac{\partial}{\partial\lambda}\left(\frac{n-k}{n}\lambda - \psi_X(\lambda)\right)\Big|_{\lambda=\lambda^*}, \\
\psi_X'(\lambda^*) &= \frac{n-k}{n}, \\
\frac{\xi e^{\lambda^*}}{\xi e^{\lambda^*} + 1 - \xi} &= \frac{n-k}{n}, \\
1 + \frac{1-\xi}{\xi}e^{-\lambda^*} &= \frac{n}{n-k}, \\
e^{-\lambda^*} &= \left(\frac{k}{n-k}\right)\left(\frac{\xi}{1-\xi}\right), \\
\lambda^* &= \log\left[\left(\frac{1-\xi}{\xi}\right)\left(\frac{n-k}{k}\right)\right]. \qquad (3.20)
\end{aligned}$$

Re-writing (3.17) using (3.20), we see that,

$$\begin{aligned}
E\left(\frac{n-k}{n}\right) &= \frac{n-k}{n}\lambda^* - \psi_X(\lambda^*), \\
&= \frac{n-k}{n}\log\left[\left(\frac{1-\xi}{\xi}\right)\left(\frac{n-k}{k}\right)\right] - \log\left[\xi\left(\frac{n-k}{k}\right)\left(\frac{1-\xi}{\xi}\right) + 1 - \xi\right], \\
E\left(\frac{n-k}{n}\right) &= \frac{n-k}{n}\log\left[\left(\frac{1-\xi}{\xi}\right)\left(\frac{n-k}{k}\right)\right] - \log\left[\frac{n(1-\xi)}{k}\right]. \qquad (3.21)
\end{aligned}$$

The result in (3.21) holds so long as $\mathbb{E}[X] \leq \frac{n-k}{n} < 1$. Since $\mathbb{E}[X] = \xi$, this is equivalent to the condition that $0 < \frac{k}{n} \leq 1 - \xi$. The R.H.S of (3.21) is equal to the divergence of two Bernoulli distributions $D(\xi'||\xi)$, evaluated where $\xi' = \frac{n-k}{n}$. Following these results back to (3.14) and (3.15), we obtain

$$P_e \approx e^{nE\left(\frac{n-k}{n}\right)}, \qquad (3.22)$$

where

$$E\left(\frac{n-k}{n}\right) = \begin{cases} 0 & \text{if } \frac{k}{n} > 1 - \xi \\ \frac{n-k}{n}\log\left[\left(\frac{1-\xi}{\xi}\right)\left(\frac{n-k}{k}\right)\right] - \log\left[\frac{n(1-\xi)}{k}\right] & \text{if } 0 < \frac{k}{n} \leq 1 - \xi \end{cases}. \qquad (3.23)$$

Note that for rates above capacity, the best exponent we can get in (3.22) is 0. The exponent in (3.23) has the property that k and n need not be known individually; all that matters is the ratio k/n. However, if the two parameters are both known, we can get a slightly better approximation by taking out the slack in the strict inequality in (3.13). Since

$\sum X_i$ will take on integer values,

$$P_e = \mathbb{P}\left\{ n - \sum_{i=1}^{n} X_i < k \right\} = \mathbb{P}\left\{ n - \sum_{i=1}^{n} X_i \leq k-1 \right\}$$
$$= \mathbb{P}\left\{ \frac{1}{n}\sum_{i=1}^{n} X_i \geq \frac{n-k-1}{n} \right\}. \tag{3.24}$$

Since (3.22) and (3.23) hold for any k, we can circumvent applying (3.14) to a new γ by simply substituting $k-1$ where we see k, obtaining

$$P_e \approx e^{nE\left(\frac{n-k+1}{n}\right)}, \tag{3.25}$$

where

$$E\left(\tfrac{n-k+1}{n}\right) = \begin{cases} 0 & \text{if } \frac{k}{n} > 1-\xi+\frac{1}{n} \\ \frac{n-k+1}{n}\log\left[\left(\frac{1-\xi}{\xi}\right)\left(\frac{n-k+1}{k-1}\right)\right] - \log\left[\frac{n(1-\xi)}{k-1}\right] & \text{if } \frac{1}{n} < \frac{k}{n} \leq 1-\xi+\frac{1}{n} \end{cases}. \tag{3.26}$$

Figure 3.2 shows (3.26) evaluated for different erasure probabilities and rates.

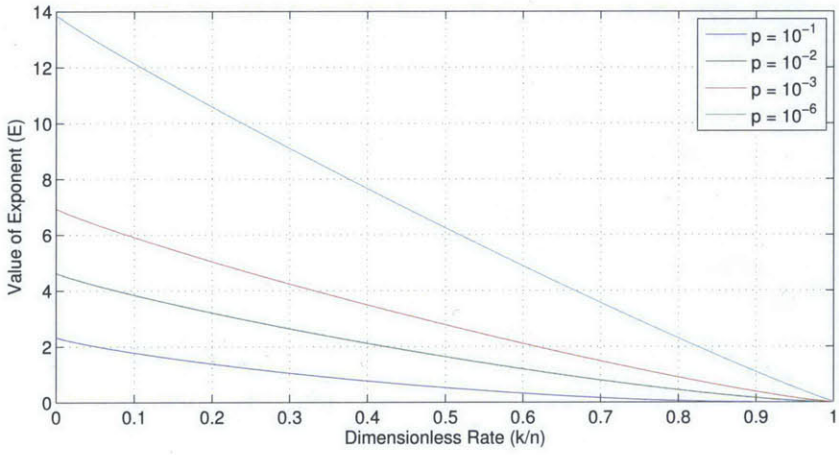

Figure 3.2: Large deviations exponents for various channels as functions of NET rate; $n = 500$

3.3 Central Limit Theorem-Based Results

As an alternative method of approximating error, we turn to some recent results pertaining to codes with finite blocklengths. While the large deviations results also apply to codes with finite blocklengths, the term "finite block-length" itself refers to a collection of results [23, 24] on channel capacity. In [23], Hayashi explored the second-order coding rate of several channels, building on the work by Strassen [25] that applies the CLT to channel coding. In [24], Polyanskiy, Poor, and Verdú, also recalling Strassen, explored more channels and provided the fundamental limit of a channel code in the non-asymptotic regime.[2]

We apply one of the results from [24] concerning the finite blocklength of discrete, memoryless channels, and modify it to relate our probability of error to the NET-layer rate. These non-asymptotic results are built around a random variable called the information density, which is a log likelihood ratio of the conditional and unconditional probabilities of the channel output. Channel capacity and channel dispersion are related to the first and second moments, respectively, of information density for particular distributions on the channel inputs. After delving into the finite blocklength results, we perform a direct application of the CLT to approximate error.

3.3.1 Finite Blocklength Approximation

We define our input alphabet (the set of coded packets we could conceivably send on the channel) as $\mathcal{A}$. Our channel either delivers these packets intact or erases them, so the output alphabet of the channel is $\mathcal{B} = \mathcal{A} \cup \{\mathrm{e}\}$. The input alphabet size will not ultimately matter, so we will just say that $|\mathcal{A}| = m$. Finally, our iid packet erasure channel can be represented as a conditional probability measure $P_{B|A} : \mathcal{A} \times \mathcal{B} \mapsto [0, 1]$, given by the following function:

$$P_{B|A}\{b|a\} = \begin{cases} 1-\xi & \text{when } b = a \\ \xi & \text{when } b = \mathrm{e} \\ 0 & \text{o/w} \end{cases} . \tag{3.27}$$

The channel we have is thus a m-ary erasure channel[3] (m-EC), and it is commonly known to have capacity $C = (1-\xi)\log m$.

[2] [24] itself is encyclopedic; the combined number of lemmas and theorems contained reaches several dozen.

[3] Customarily this would be a Q-ary erasure channel, but we use a different variable to avoid confusion with the Q-function.

To introduce the notion of channel dispersion, we first define information density:

$$i(A;B) \triangleq \log \frac{P_{B|A}\{B|A\}}{P_B\{B\}}, \tag{3.28}$$

where P_B is an output distribution resulting from the composition of $P_{B|A}$ on an input distribution P_A; thus, $i(A;B)$ is a derived random variable. Although not typically defined as such, we see that Shannon capacity is a function of the first moment of information density: $C = \sup_{P_A} \mathbb{E}\left[i(A;B)\right]$. If we define Π as the set of all capacity-achieving distributions on $\mathcal{A}$ for our channel, then we can define the channel dispersion as

$$V_{min} \triangleq \min_{P_A \in \Pi} \mathrm{Var}\left(i^2(A;B)\right) = \min_{P_A \in \Pi} \mathbb{E}\left[\log^2 \frac{P_{B|A}\{B|A\}}{P_B\{B\}}\right] - C^2, \tag{3.29}$$

which is a somewhat simplified version of what is presented in (239-244), (270) of [24]. For the m-EC, the only capacity-achieving distribution on $\mathcal{A}$ is uniform, so we get the output distribution

$$P_B\{b\} = \begin{cases} m^{-1}(1-\xi), & b \neq \mathrm{e} \\ \xi & b = \mathrm{e}. \end{cases} \tag{3.30}$$

Now, we can quickly find that

$$\begin{aligned} \mathbb{E}\left[\log^2 \frac{P_{B|A}\{B|A\}}{P_B\{B\}}\right] &= \mathbb{E}_A\left[\mathbb{E}_{B|A}\left[\log^2 \frac{P_{B|A}\{B|A\}}{P_B\{B\}} \mid A\right]\right] \\ &= \mathbb{E}_A\left[(1-\xi)\log^2\left(\frac{m(1-\xi)}{1-\xi}\right) + \xi \log^2\left(\frac{\xi}{\xi}\right)\right] \\ &= (1-\xi)\log^2 m, \end{aligned} \tag{3.31}$$

which produces the m-EC channel dispersion

$$V_{min} = \left[(1-\xi) - (1-\xi)^2\right]\log^2 m = (\xi - \xi^2)\log^2 m. \tag{3.32}$$

At this point we should introduce Theorem 53 from [24], which states that, for a binary erasure channel and a $P_e \in (0, \frac{1}{2}]$, the maximum achievable message alphabet size $M^*(n, P_e)$ obeys the equation

$$\log M^*(n, P_e) = nC - \sqrt{nV_{min}}Q^{-1}(P_e) + O(1), \tag{3.33}$$

where Q^{-1} is the inverse Q-function. Although (3.33) describes a binary erasure channel instead of an m-EC, we know from Theorem 49 in [24] that (3.33) is achievable for a m-EC. Proof of the converse is given in the Appendix.

For a m-EC, $M^*(n, P_e) = m^{k^*}$, where k^* need not be an integer. Evaluating (3.33), we see that

$$\begin{aligned}\log M^*(n, P_e) &= k^* \log m = nC - \sqrt{nV_{min}}Q^{-1}(P_e) + O(1) \\ &= n(1-\xi)\log m - \sqrt{n(\xi - \xi^2)}\log(m)Q^{-1}(P_e) + O(1)\end{aligned}$$

so that

$$\frac{k^*}{n} = (1-\xi) - \sqrt{\frac{\xi - \xi^2}{n}}Q^{-1}(P_e) + o(\frac{1}{n}). \tag{3.34}$$

Rearranging, we obtain the following expression for probability of error in terms of k^*:

$$P_e = Q\left(\frac{1 - \xi - \frac{k^*}{n} + o(\frac{1}{n})}{\sqrt{\xi - \xi^2}}\sqrt{n}\right). \tag{3.35}$$

By removing the $o(1)$ term, we derive an approximation of error for rates around capacity, so that for $\frac{k}{n} \leq 1 - \xi$ (this inequality is needed to satisfy the assumptions on (3.33)),

$$P_e \approx Q\left(\frac{1 - \xi - \frac{k}{n}}{\sqrt{\xi - \xi^2}}\sqrt{n}\right). \tag{3.36}$$

Having been derived for codes achieving capacity, it might seem this approximation deteriorates as $\frac{k}{n}$ drops well below capacity. In fact, the difference between (3.36) and our tight combinatorial bound in (3.9) actually decreases with rate for the case shown in Figure 3.3a. However, Figure 3.3b depicts the absolute value of the log ratios between each approximation and the combinatorial bound. We see that the CLT approximation becomes less useful as an order-of-magnitude estimation of P_e as rate gets farther from capacity.

3.3.2 Central Limit Theorem Approximation

Although the finite block-length approach is based on second-moment analysis, like Shannon's original notion of capacity, it does not indicate the structure of the capacity-achieving code. This leads us to consider a more straightforward approach to a second-order error

(a) (b)

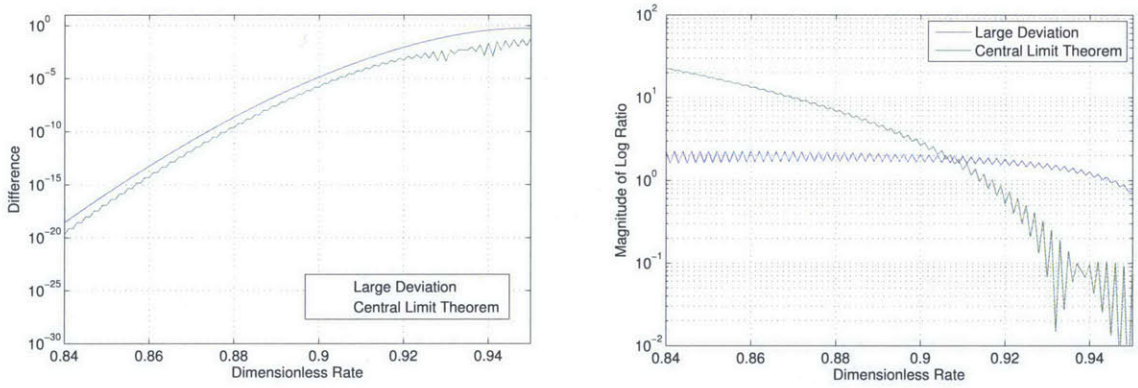

Figure 3.3: Measures of the difference between the combinatorial bound and finite block-length and large deviations approximations for RLNC error rates as a function of $\frac{k}{n}$, where $n = 500$, $\xi = 0.05$, and $q = 2^8$. The curves shown are computed as $|P_{e,apprx}(x) - P_{e,comb}(x)|$ in (a) and $\left|\frac{\log P_{e,apprx}(x)}{\log P_{e,comb}(x)}\right|$ in (b), where $x = k/n$.

approximation involving the structure of RLNC just as we did in Section 3.2. We let X_i retain its definition and distribution from before, and note that

$$\mathbb{E}[X] = \xi, \quad \text{and} \quad \sigma_X^2 = \xi - \xi^2. \tag{3.37}$$

Again we use the property that, for RLNC,

$$P_e = \mathbb{P}\left\{ n - \sum_{i=1}^{n} X_i \ < k \right\}. \tag{3.38}$$

Applying the CLT to this observation, we can see that in the limit, probability of error can be found using a normal CDF:

$$\begin{aligned}
\lim_{n\to\infty} 1 - P_e &= \lim_{n\to\infty} \mathbb{P}\left\{ n - \sum_{i=1}^{n} X_i \ \geq k \right\} \\
&= \lim_{n\to\infty} \mathbb{P}\left\{ \frac{\sum_{i=1}^{n} X_i - n\mathbb{E}[X]}{\sqrt{n\sigma_X^2}} \leq \frac{n(1-\mathbb{E}[X]) - k}{\sqrt{n\sigma_X^2}} \right\} \\
&= \Phi\left(\frac{n(1-\mathbb{E}[X]) - k}{\sqrt{n\sigma_X^2}} \right)
\end{aligned} \tag{3.39}$$

so that for large but finite n,

$$P_e \approx 1 - \Phi\left(\frac{1-\xi-\frac{k}{n}}{\sqrt{\xi-\xi^2}}\sqrt{n}\right)$$
$$= Q\left(\frac{1-\xi-\frac{k}{n}}{\sqrt{\xi-\xi^2}}\sqrt{n}\right). \tag{3.40}$$

This result is the same as we obtained in (3.36), but the approach is much simpler than calculating the first and second moments of information density, which we are required to do for the finite block-length approach. The strength of the Polyanskiy, Poor, and Verdú approach is that it uncovers a limit fundamental to the channel, but for the specific case of RLNC we see that we can get the same result with less computation by simply applying the CLT.

3.4 Chapter Summary

In this chapter, we outlined four methods of getting an estimate for probability of error using RLNC over packet erasure channels. We found that simply applying the CLT to our packet erasures yielded the same result as calculating the finite blocklength capacity using channel dispersion.

To determine whether the large deviations- or CLT-based approach is more accurate in any particular region, we have plotted some sample curves for each along with the tight combinatorial lower bound from (3.9) in Figures 3.4, 3.5, and 3.6. We can see that, for large n and rates well below capacity, the large deviations approximation follows the combinatorial bound very closely. However, for the region right around capacity, the CLT approximation seems to always be the closer of the two; this property will have interesting consequences in Chapter 5. Figure 3.5 shows to some extent the effect of a smaller field size on these approximations, as the probability of error for any rate increases slightly. Figure 3.6 is interesting in that the large deviations and CLT approximations intersect, which shows clearly the regions in which each approximation is closer to the combinatorial bound.

Finally, we should reiterate that these approximations are derived by simply defining the event that a threshold number of iid events has occurred and evaluating the probability of that event. The same methodology could be applied to an ideal maximum-distance separable (MDS) code with an iid probability of symbol error. For an ideal MDS code of distance d, if $\sum X_i$ indicates the number of errors in a codewords, probability of error is given by

$\mathbb{P}\{\sum X_i \geq d/2\}$. If these errors are iid, we can apply the results derived in this chapter by just substituting $d/2$ for k.

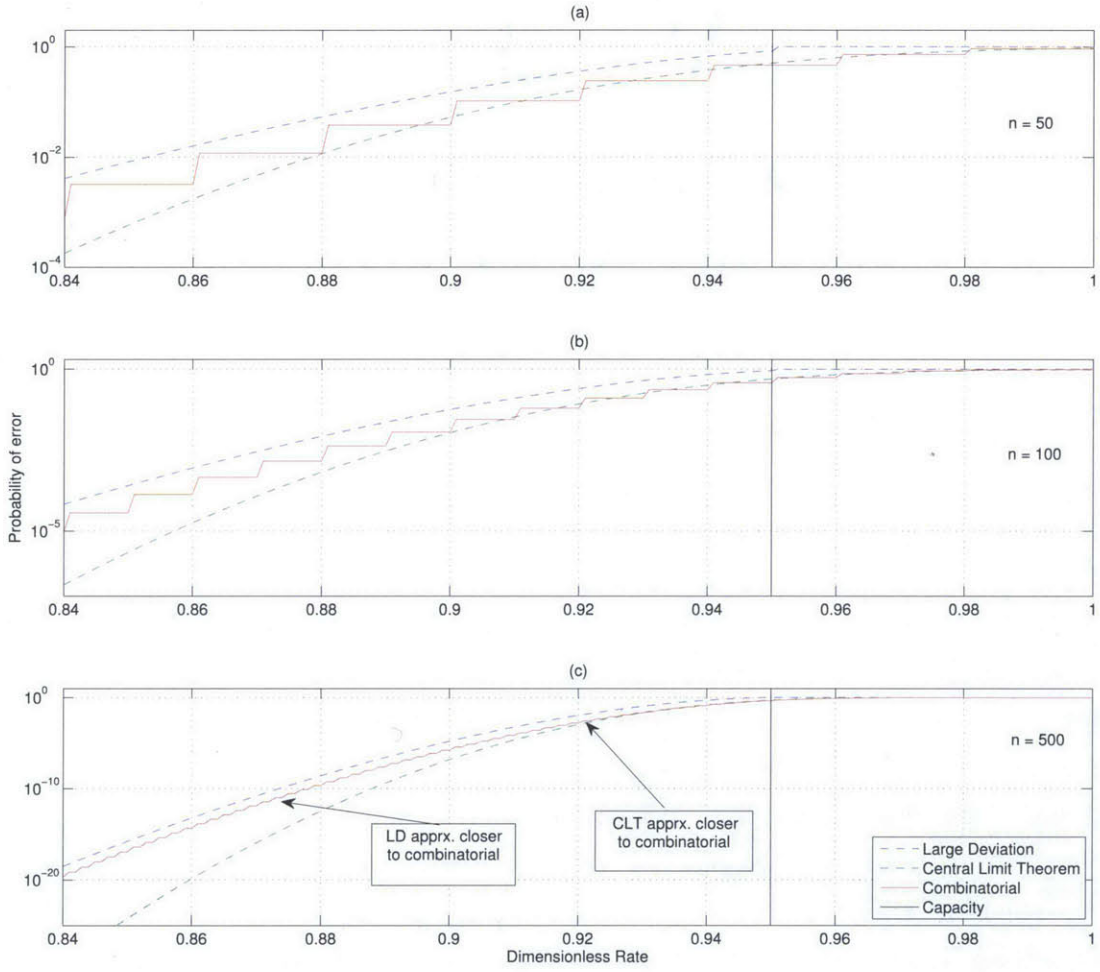

Figure 3.4: Comparison of combinatorial, large deviations, and CLT error approximations for RLNC when $\xi = 0.05$ and $q = 2^8$. In (a), $n = 50$; in (b), $n = 100$; in (c), $n = 500$.

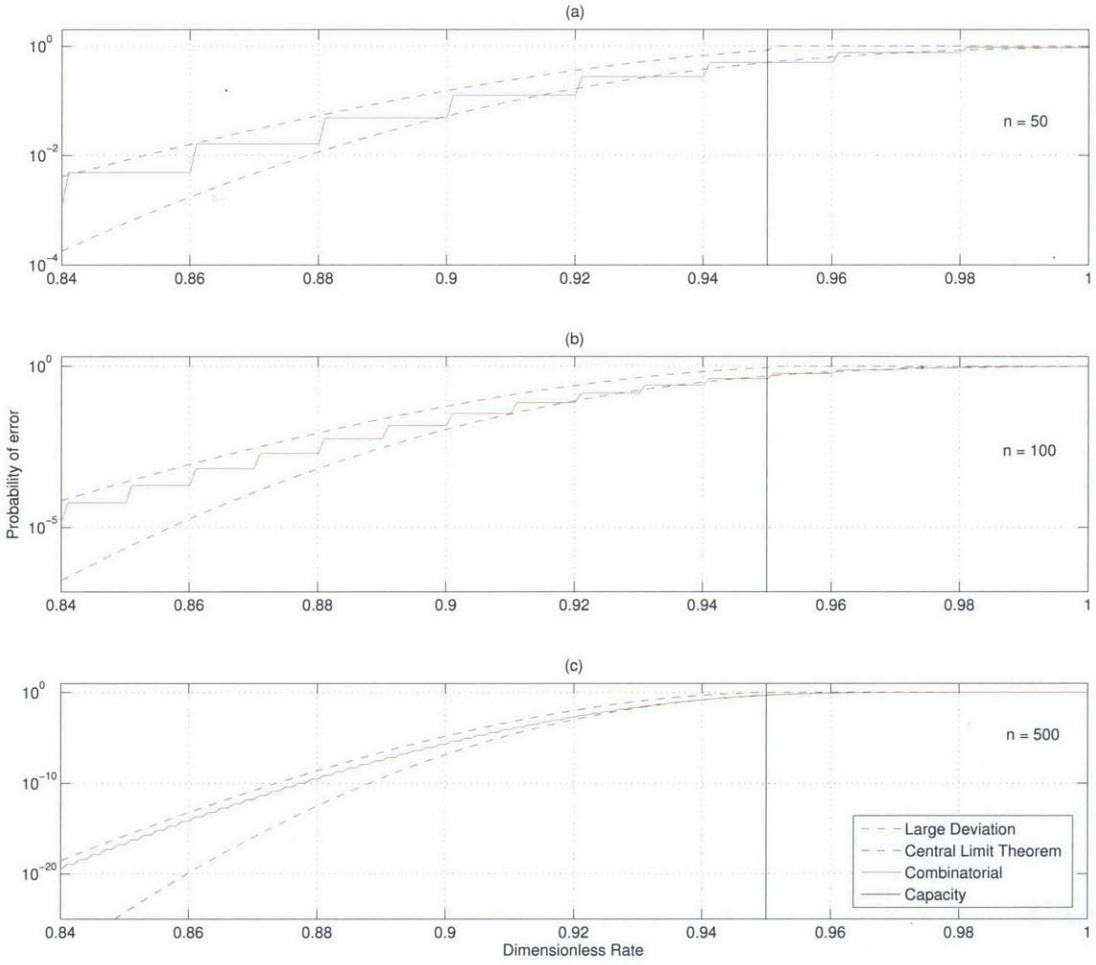

Figure 3.5: Comparison of combinatorial, large deviations, and CLT error approximations for RLNC when $\xi = 0.05$ and $q = 2^3$. In (a), $n = 50$; in (b), $n = 100$; in (c), $n = 500$. Notice that the smaller field size has shifted the combinatorial curve up.

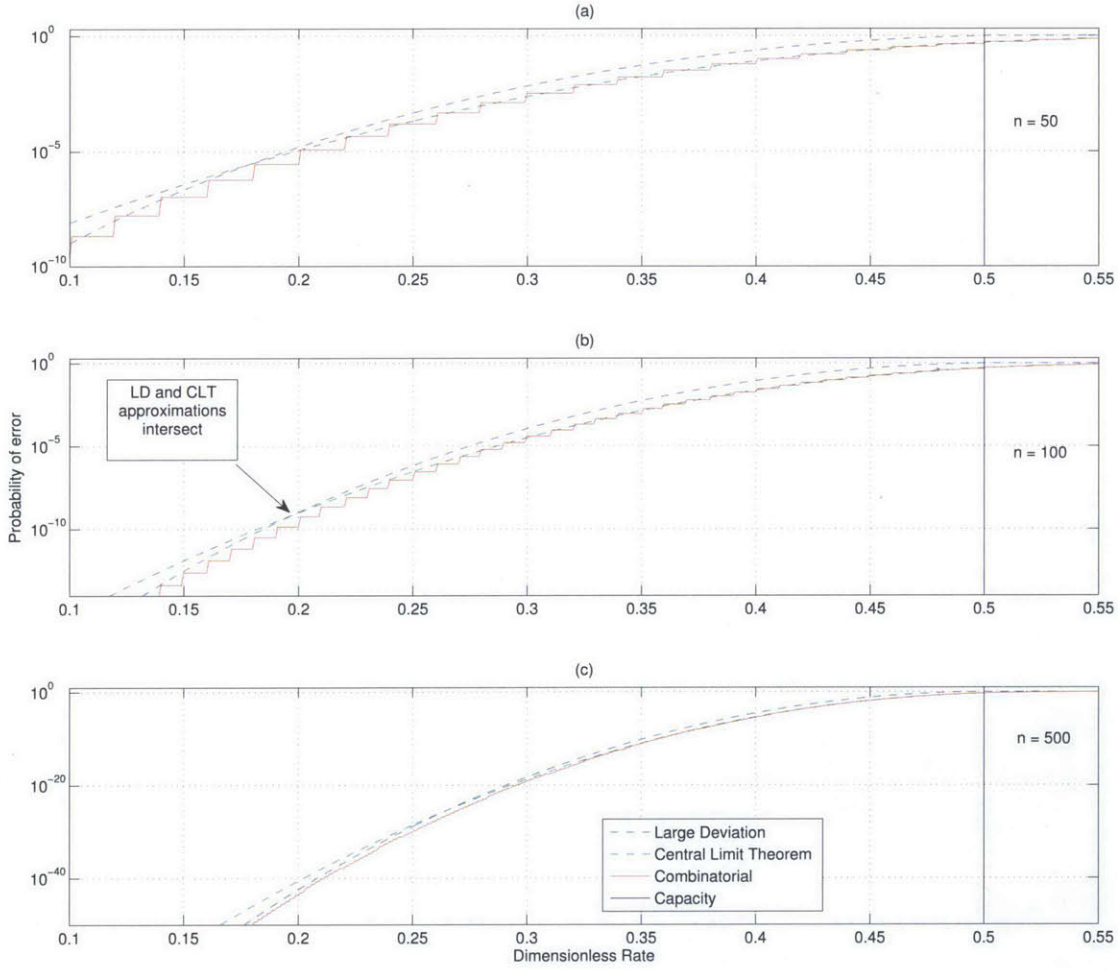

Figure 3.6: Comparison of combinatorial, large deviations, and CLT error approximations for RLNC when $\xi = 0.5$ and $q = 2^8$. In (a), $n = 50$; in (b), $n = 100$; in (c), $n = 500$. The CLT approximation is less accurate than the large deviation approximation for rates well below capacity.

Chapter 4

Network Reduction

In the last chapter, we derived several explicit functions to approximate the probability that D, the number of DoF to arrive at the destination, is less than sufficient for decoding for a point-to-point connection. In this chapter, we consider larger networks. Since RLNC affords the ability to re-encode at intermediate links, generating additional redundancy to compensate for packet losses that have already been incurred, the analysis is not a simple generalization of point-to-point systems.

Recall from Section 3.1.1 that coded packets are a linear combination of our uncoded packets; that is that

$$x_i = a_{i,1}v_1 + a_{i,2}v_2 + \cdots + a_{i,k}v_k. \tag{4.1}$$

Re-encoding is virtually the same operation as the original coding, except the input consists of the already-coded x_i vectors. These are prescribed a random coefficient from $\mathbb{F}_q$ and added together; for example, we could get

$$\begin{aligned}\gamma &= b_1x_1 + b_2x_2 \\ &= b_1\left(a_{1,1}v_1 + a_{1,2}v_2 + \cdots + a_{1,k}v_k\right) + b_2\left(a_{2,1}v_1 + a_{2,2}v_2 + \cdots + a_{2,k}v_k\right) \\ &= (b_1a_{1,1} + b_2a_{2,1})v_1 + (b_1a_{1,2} + b_2a_{2,2})v_2 + \cdots + (b_1a_{1,k} + b_2a_{2,k})v_k,\end{aligned} \tag{4.2}$$

We see that re-encoding yields more linear combinations of our original packets, and the new coded packets have the same properties as the original coded packets.

We make a couple of assumptions to begin our network analysis. We assume that our re-encoding operations are non-degenerate; if the coding vectors of the packets received at any node span a subspace $\mathcal{U}$ of $\mathbb{F}_q^k$, then the new vectors of the re-encoded packets also span $\mathcal{U}$. Furthermore, we assume that linear dependence is never an issue when we have received

at least k packets; across any link, we will hold that the encoding vectors of any k packets span $\mathcal{U}$.

4.1 Tandem Link Equivalence

Initially, we assume that, on any path from our source node to our destination node, no two links are transmitting the same generation of data at the same time. If all of the coded packets containing information about one generation of data have been transmitted across one link before any coded packets are sent on the next link in a series, the distribution of D is easy to find. In Section 4.1.2, we will argue that our method is valid even when links send the same generation of data concurrently.

4.1.1 Reduction Operation

Our network consists of two directed links connecting three nodes, as in Figure 4.1. We call these links 1 and 2. Link $i \in \{1, 2\}$ permits us to send n_i packets per generation. Now define a random variable describing the number of successfully transmitted packets over the link in a generation:

$$S_i \triangleq \#\{\text{packets received on link } i\}. \tag{4.3}$$

Assuming that we have an explicitly defined channel model, we can obtain a probability mass function (PMF) for S_i, which is described by the vector $\vec{\lambda}_i \in [0, 1]^{n_i+1}$, where

$$\vec{\lambda}_i[s] = \mathbb{P}\{S_i = s\}, \quad s \in \{0, 1, \ldots, n_i\}. \tag{4.4}$$

We also find it useful to define another vector representing the complementary cumulative distribution function (CCDF) of S_i, $\vec{\Lambda}_i^c \in [0, 1]^{n_i+1}$:

$$\vec{\Lambda}_i^c[s] = \mathbb{P}\{S_i \geq s\} = \sum_{j=s}^{n_i} \vec{\lambda}_i[j], \quad s \in \{0, 1, \ldots, n_i\}. \tag{4.5}$$

At this point, we recall the assumptions involving the linear independence of coding vectors and re-encoding. Referring to Figure 4.1, we recognize that the span of the coding vectors available at the destination is a subspace of the span of the coding vectors available at node R. Each packet arriving at R (up to the k^{th} packet) adds another degree of freedom and increases the dimension of the subspace spanned by the coding vectors in R's buffer.

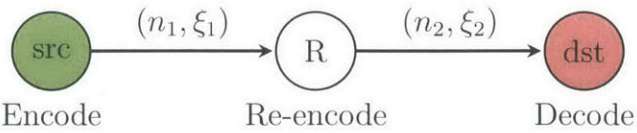

Figure 4.1: Basic two-link tandem network

At the destination, however, the largest set of linearly independent coding vectors available cannot exceed the number of DoF available at R or the number of packets received on the second link. Thus, the number of DoF available at the destination is the minimum number of packets received across either link.

For precision, let us formalize this notion. We will redefine

$$D \triangleq \#\{\text{DoF at destination node}\}, \tag{4.6}$$

and note that,[1]

$$D = \min\{S_1, S_2\}. \tag{4.7}$$

To find the distribution of D, we note the following property:

$$\begin{aligned}\mathbb{P}\{D \geq s\} &= \mathbb{P}\left[\{S_1 \geq s\} \cap \{S_2 \geq s\}\right] \\ &= \mathbb{P}\{S_1 \geq s\}\mathbb{P}\{S_2 \geq s\}, \quad s \in \{0, 1, \ldots, k\},\end{aligned} \tag{4.8}$$

where the second equality follows from the independence of the two links. We have now found the CCDF of a single link equivalent to the two tandem links. We can re-write this using our vector notation:

$$\vec{\Lambda}^c_{EQ}[s] = \vec{\Lambda}^c_1[s]\vec{\Lambda}^c_2[s], \quad s \in \{0, 1, \ldots, k\}. \tag{4.9}$$

Note that increasing the value of k to $k' > k$ will not affect the values of $\vec{\Lambda}^c_{EQ}[s]$ for $s \leq k$. Additionally, for generation size k, $\vec{\Lambda}^c_{EQ}[k]$ is the probability of decoding success across the tandem link. Thus, by evening the length of constituent vectors $\vec{\Lambda}^c_1$ and $\vec{\Lambda}^c_2$ by either zero-padding the shorter one or truncating the longer one - we denote this operation with an apostrophe ($'$) - we obtain a vector representing end-to-end reliability for any generation size, given by

$$\vec{\Lambda}^c_{EQ} = \vec{\Lambda'}^c_1 \odot \vec{\Lambda'}^c_2, \tag{4.10}$$

[1]Really, $D = \min\{S_1, S_2, k, \}$, but our choice of k will not affect the derivations to follow, and when D exceeds k, we are able to decode anyway.

where $\odot$ denotes element-wise multiplication of two vectors, known as the Hadamard product. We justify truncating the longer CCDF when performing this operation because for $s > n_i$, then $\mathbb{P}\{S_i \geq s\} = 0$, so the truncated $\vec{\Lambda}^{c}_{EQ}$ captures the entire support of D.

4.1.2 Simultaneous Transmissions

If, instead of waiting at the intermediate node of Figure 4.1 to receive all information before transmitting, we begin forwarding DoF to the destination as they arrive, we cannot say that (4.7) holds with equality. Rather, we know that,

$$D \leq \min\{S_1, S_2\} \tag{4.11}$$

and would like see how tight this inequality is. Because of the delay constraint, we must investigate this question in the non-asymptotic regime, so we describe S_1, S_2, and D as arrival processes, using $S_1(t)$, $S_2(t)$, and $D(t)$ to indicate the value of these processes at time t. Let us also define a stochastic process to represent the slack in (4.11), taking values in $\mathbb{Z}^+$,

$$Y(t) \triangleq \min\{S_1(t), S_2(t)\} - D(t). \tag{4.12}$$

There are two distinct cases we need to consider. Case 1 is when $S_1(t) \leq S_2(t)$. In this case,

$$Y(t) = S_1(t) - D(t). \tag{4.13}$$

We can see that $Y(t)$ increases every time a packet is delivered successfully on the first link and decreases when a packet is delivered successfully on the second link. If packets arrive at the same rate on each link, then $Y(t)$ could be described by the countable-state Markov chain in Figure 4.2. So long as $\xi_1 > \xi_2$, the Markov Chain is positive recurrent, and the drift is in the direction of the zero state. If packets are transmitted on link 1 at a higher rate than on link 2, the Markov chain no longer holds, but the events corresponding to changes in the value of $Y(t)$ remain the same: namely, increases in $S_1(t)$ and $S_2(t)$. So long as packets are *successfully* delivered on link 2 faster than on link 1, we would expect the distribution of $Y(t)$ to have a small tail. Otherwise, we would intuit that we are no longer interested in Case 1.

Case 2 is when $S_1(t) > S_2(t)$. In this case, the following equation holds,

$$Y(t) = S_2(t) - D(t). \tag{4.14}$$

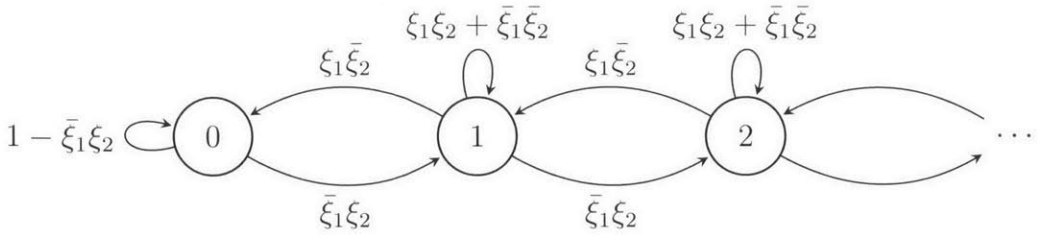

Figure 4.2: A countable-state Markov chain depicting the state of $Y(t) = S_1(t) - D(t)$, where $\bar{\xi}_i = 1 - \xi_i$.

When a packet is successfully delivered on link 2, $S_2(t)$ increases. If the delivered packet contains new information, then $D(t)$ also increases. Thus, $Y(t)$ only increases if a packet is delivered on link 2 whose coding vector is already in the span of the coding vectors that have already been delivered to the destination. In the worst-case scenario, when $D(t) = S_1(t)$, any packet delivered on link 2 will not increase $D(t)$. Thus, $Y(t)$ only increases when $S_2(t)$ increases if and only if

$$D(t) = S_1(t). \tag{4.15}$$

Combining (4.14) with (4.15) and the condition that $Y(t) \geq 0$, we see that $Y(t)$ can only increase when

$$\begin{aligned} Y(t) &= \min\{S_1(t), S_2(t)\} - D(t) \\ &= S_2(t) - D(t) \\ &= S_2(t) - S_1(t) \geq 0, \end{aligned} \tag{4.16}$$

so that,

$$S_2(t) \geq S_1(t). \tag{4.17}$$

This contradicts the assumption that $S_1(t) > S_2(t)$; thus, in Case 2, $Y(t) = 0$.

Since we need not worry about Case 2, the only time (4.11) may not be met with equality is when $S_1 \leq S_2$. Under these circumstances, we cannot say that $\mathbb{P}\{Y(t) = 0\}$ converges in probability, but the fact that $Y(t)$ forms a positive recurrent Markov Chain indicates that as infinitely many packets are sent, $Y(t)$ will reach 0 infinitely more times. If we solve for the steady-state distribution $\pi_i = \lim_{t\to\infty} \mathbb{P}\{Y(t) = i\}$ of the Markov Chain depicted in Figure 4.2, we obtain

$$\pi_i = \left(\frac{\bar{\xi}_1\xi_2}{\xi_1\bar{\xi}_2}\right)^i \pi_0, \tag{4.18}$$

so that by solving

$$\sum_{i=0}^{\infty} \pi_i = 1, \tag{4.19}$$

we obtain that

$$\pi_0 = \lim_{t\to\infty} \mathbb{P}\{Y(t) = 0\} = 1 - \frac{\bar{\xi}_1 \xi_2}{\xi_1 \bar{\xi}_2}. \tag{4.20}$$

This shows that, while $Y(t)$ does not converge in probability to 0, the tail of the distribution diminishes exponentially.

4.2 Parallel Link Equivalence

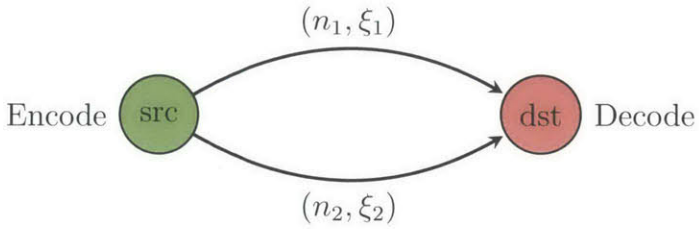

Figure 4.3: Basic two-link parallel network

Now we consider the scenario of two links in parallel, as shown in Figure 4.3. This problem can be regarded as a kind of dual to the tandem link problem. We carry over much of the terminology from the previous section; for link i, let n_i be the number of packets we can send for one generation and let S_i be the number of those packets that are successfully received. Also, let D again indicate the number of DoF received at the destination. Previously we assumed that any k packets received across one link would have linearly independent coding vectors; here we extend this assumption so that any k packets selected arbitrarily from links 1 and 2 will have linearly independent coding vectors. Thus,

$$D = S_1 + S_2. \tag{4.21}$$

Because S_1 and S_2 are independent, this means that we can derive a distribution for D by convolving the distributions of S_1 and S_2:

$$\mathbb{P}\{D = s\} = \sum_{j=0}^{s} \mathbb{P}\{S_1 = j\}\mathbb{P}\{S_2 = s - j\}. \tag{4.22}$$

Using vector notation to represent the convolution gives us the more compact form,

$$\vec{\lambda}_{EQ} = \vec{\lambda}_1 * \vec{\lambda}_2, \quad \vec{\lambda}_{EQ} \in [0,1]^{n_1+n_2+1}. \tag{4.23}$$

Notice that we can easily switch between $\vec{\lambda}$ and $\vec{\Lambda}^c$ using linear operators. From (4.5), we see that, if we consider $\vec{\lambda}$ as a column vector, then we can obtain $\vec{\Lambda}^c$ through the linear equation,

$$\vec{\Lambda}^c = \mathbf{U}_n \vec{\lambda}, \tag{4.24}$$

where $\mathbf{U}_n$ is an $n \times n$ upper triangular matrix where all nonzero entries are 1. $\mathbf{U}_n$ is full-rank and invertible, so we are able to execute the reverse operation though the following linear equation,

$$\vec{\lambda} = \mathbf{U}_n^{-1} \vec{\Lambda}^c. \tag{4.25}$$

The inverse of $\mathbf{U}_n$ has an equally simple structure: every entry in the main diagonal is a 1, and every entry in the diagonal above that is -1. To see this, instead of inverting $\mathbf{U}_n$, we just recognize that

$$\begin{aligned} \vec{\lambda}_i[s] &= \mathbb{P}\{S_i = s\} \\ &= \mathbb{P}\{S_i \geq s\} - \mathbb{P}\{S_i \geq s+1\} \\ &= \vec{\Lambda}_i^c[s] - \vec{\Lambda}_i^c[s+1]. \end{aligned} \tag{4.26}$$

Then we can re-write (4.23) in terms of the complementary cumulative distribution functions (CCDFs),

$$\vec{\Lambda}_{EQ}^c = \mathbf{U}_{n_1+n_2+1} \left[\left(\mathbf{U}_{n_1}^{-1} \vec{\Lambda}_1^c \right) * \left(\mathbf{U}_{n_2}^{-1} \vec{\Lambda}_2^c \right) \right], \quad \vec{\Lambda}_{EQ}^c \in [0,1]^{n_1+n_2+1}. \tag{4.27}$$

4.3 Network Throughput

The reduction operations outlined in the previous two sections will allow us to simplify some networks down to a single probability distribution. For example, the distribution of DoF received at the destination node of the relay network presented in Figure 4.4 is found by first reducing the tandem links in the upper path, then combining the result with the lower path.

Given a delay constraint in seconds, τ, and the distribution of D, we can calculate $\mathbb{E}[D]$, the expected number of DoF received at the destination node after τ seconds. It is simple to compute $\mathbb{E}[D]$, given that we already know the CCDF of D. Because D takes on values

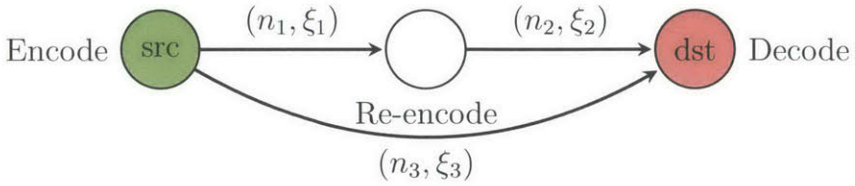

Figure 4.4: Relay network

in a subset of $\mathbb{Z}^+$, we can use the following calculation of expectation:

$$\begin{aligned}
\mathbb{E}[D] &= \sum_{d=0}^{\infty} d \cdot \mathbb{P}\{D = d\} \\
&= \sum_{d=0}^{\infty} 1 - \mathbb{P}\{D < d\} \\
&= \sum_{d=1}^{\infty} \vec{\Lambda}_D^c[d] \\
&= \sum_{d} \vec{\Lambda}_D^c[d] - \vec{\Lambda}_D^c[0] \\
&= \sum_{d} \vec{\Lambda}_D^c[d] - 1. \qquad (4.28)
\end{aligned}$$

Additionally, having the CCDF of D allows us to select a rate that achieves a certain constraint on probability of error, which is a common objective for practical communication system design. For any $\epsilon \in (0, 1]$, we code together k_ϵ^* packets, where k_ϵ^* is

$$k_\epsilon^* \triangleq \max\{k : \mathbb{P}\{D \geq k\} \geq 1 - \epsilon\} = \max\{k : \vec{\Lambda}_D^c[k] \geq 1 - \epsilon\}. \qquad (4.29)$$

4.3.1 Relation to Cut-set Capacity

Let us define the quantity $\frac{\mathbb{E}[D]}{\tau}$ as the time-limited throughput of the network. As we let $\tau \to \infty$, we can show that this time-limited throughput converges to the cut-set capacity of a network. To see this, let us imagine the same network in two different ways. Let our network $\mathcal{G}$ be a set of nodes $\mathcal{V}$ and links $\mathcal{E}$ with source node v_s and destination node v_d. Furthermore, let us assume that the links $\mathcal{E}$ are arranged so that the flow from v_s to v_d is a collection of tandem and parallel link structures, so that the reduction operations defined in Sections 4.1 and 4.2 can be iteratively applied to produce a single distribution on D. Let packet arrivals across each link $i \in \mathcal{E}$ be iid with erasure probability ξ_i. Let ℓ be fixed across

$\mathcal{G}$. Let each link $i \in \mathcal{E}$ have PHY-layer rate of ρ_i bits per second.[2]

In the first scenario, there is no delay constraint, so we find the cut-set capacity as described in [19]. Each packet contains ℓ bits of information, and the PHY-layer code on link i has rate ρ_i bits per second. Thus, data is transmitted at the rate of $\frac{\rho_i}{\ell}$ bits per second, and $\frac{\rho_i}{\ell}(1-\xi_i)$ is the average rate that data flows across link i, $\forall i \in \mathcal{E}$. The min-cut-max-flow theorem tells us that the capacity of the flow from v_s to v_d is determined by the minimum cut; if $\Xi \subseteq \mathcal{V}$ is the minimum cut-set, then capacity of the flow from v_s to v_d is given by

$$C_{\mathcal{G}} = \frac{1}{\ell} \sum_{i \in \Xi} \rho_i (1 - \xi_i). \tag{4.30}$$

In the second scenario, we have a delay constraint of τ seconds and want to find the value of $\frac{\mathbb{E}[D]}{\tau}$ by iteratively reducing parallel and tandem links. As mentioned in Chapter 2, $n_i \leq \tau \rho_i / \ell$. Ignoring integrality constraints, we let n_i satisfy this relationship with equality. The operations achieving the network reduction could be distilled to representing D as a function of two operations, $\min\{S_i, S_j\}$ and $S_i + S_j$ for $i, j \in \mathcal{E}$. We now scale all S_i by $\frac{1}{\tau}$ to derive $\frac{D}{\tau}$. Note that

$$\mathbb{P}\left\{\frac{S_i}{\tau} \geq s\right\} = \mathbb{P}\{S_i \geq s\tau\} \ . \tag{4.31}$$

In Chapter 3, we used the variable X to track packet losses on any link; $X = 1$ if the packet is erased. We reuse this notation, but since we have multiple links, we distinguish links with a superscript, using $\{X_j^{(i)} : i \in \mathcal{E},\ 1 \leq j \leq n_i\}$. Since S_i is the sum of all packets *not* erased,

$$S_i = n_i - \sum_{j=1}^{n_i} X_j^{(i)}. \tag{4.32}$$

For each link $i \in \mathcal{E}$, the sequence $\{X_j^{(i)}\}$ is iid, so for any $\epsilon > 0$ we apply the weak law of large numbers,

$$\lim_{n_i \to \infty} \mathbb{P}\left\{\left|\frac{1}{n_i}\sum_{j=1}^{n_i} X_j^{(i)} - \mathbb{E}[X^{(i)}]\right| > \epsilon\right\} = 0.n \tag{4.33}$$

We see that this could be re-written using S_i; furthermore, because $n_i = \frac{\tau \rho_i}{\ell}$, we can replace

[2] In Chapter 2 we define ρ in bits per channel use, but multiplying ρ by the bandwidth produces the rate in bits per second.

our limit in n_i with a limit in τ. We also use the fact that $\mathbb{E}[X^{(i)}] = \xi_i$ to obtain

$$\lim_{n_i \to \infty} \mathbb{P}\left\{\left|\frac{n_i - S_i}{n_i} - \xi_i\right| > \epsilon\right\} = 0$$
$$\lim_{n_i \to \infty} \mathbb{P}\left\{\left|\frac{S_i}{n_i} - (1 - \xi_i)\right| > \epsilon\right\} = 0$$
$$\lim_{\tau \to \infty} \mathbb{P}\left\{\left|\frac{\ell S_i}{\tau \rho_i} - (1 - \xi_i)\right| > \epsilon\right\} = 0$$

Since ρ_i and ℓ are both strictly positive,

$$\lim_{\tau \to \infty} \mathbb{P}\left\{\left|\frac{S_i}{\tau} - \frac{\rho_i}{\ell}(1 - \xi_i)\right| > \frac{\rho_i}{\ell}\epsilon\right\} = 0. \tag{4.34}$$

Thus we see that $\frac{S_i}{\tau}$ converges in probability to $\frac{\rho_i}{\ell}(1 - \xi_i)$. For our two network reduction operations, then,

$$\lim_{\tau \to \infty} \min\left\{\frac{S_i}{\tau}, \frac{S_j}{\tau}\right\} = \frac{1}{\ell}\min\{\rho_i(1 - \xi_i), \rho_j(1 - \xi_j)\} \tag{4.35}$$

and

$$\lim_{\tau \to \infty} \left(\frac{S_i}{\tau} + \frac{S_j}{\tau}\right) = \frac{1}{\ell}\left(\rho_i(1 - \xi_i) + \rho_j(1 - \xi_j)\right). \tag{4.36}$$

At the end of the network reduction, we have the cut-set,

$$\lim_{\tau \to \infty} \frac{D}{\tau} = \frac{1}{\ell}\sum_{i \in \Xi} \rho_i(1 - \xi_i). \tag{4.37}$$

By the law of large numbers, the ensemble average of $\frac{D}{\tau}$ converges in probability to its mean, so the R.H.S of (4.37) must be the value of $\mathbb{E}\left[\frac{D}{\tau}\right]$. Since this is equal to (4.30), we see that the time-constrained method is equal in the limit to the method applied in [19].

4.3.2 Impact of the CCDF

Compared to viewing a link as a bit pipe, deriving a CCDF at the destination provides more information about how a coding scheme affects the overall goodput of the system. If the PHY schemes on the individual links are fixed, then $\vec{\Lambda}_D^c$ can be interpreted as the reliability, which we define to be $1 - P_e$, given as a function of k, since $\vec{\Lambda}_D^c[k]$ is the end-to-end error

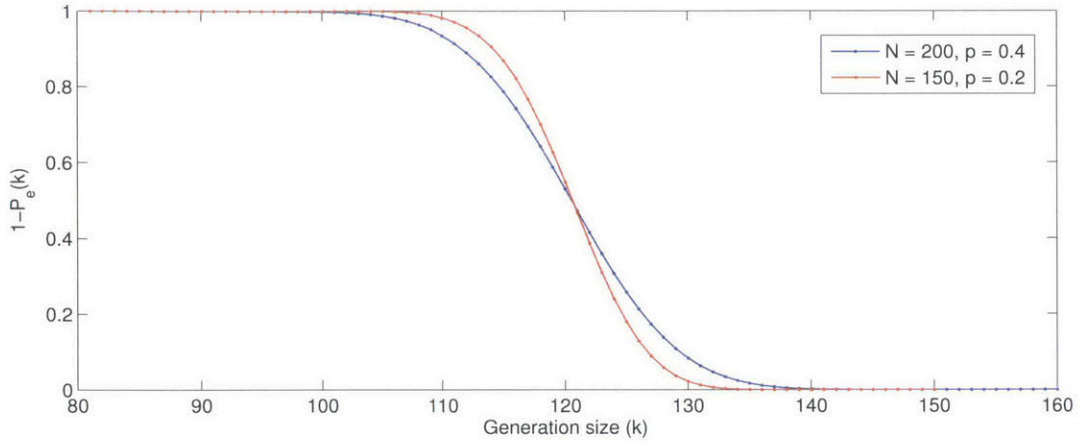

Figure 4.5: CCDFs of two network links with the same average rate but different packet constraint and erasure probability.

probability when we use a network code of generation size k. Additionally, we might find that two links might have different distributions despite having the same expected rate.

To motivate the use of the CCDF over the average rate alone, consider two links, 1 and 2, over which n_1 and n_2 channel uses are permitted, respectively, for each generation of data. For these links, let the average number of successfully-delivered coded packets per generation on each link, $\mathbb{E}[S_i]$ be equal:

$$n_1(1-\xi_1) = n_2(1-\xi_2). \tag{4.38}$$

Without loss of generality, let $n_1 > n_2$, so that $\xi_1 > \xi_2$. However, since $\mathrm{Var}(S_i) = n_i\xi_i(1-\xi_i)$, then S_1 has a higher variance than S_2. As a result, the slope of the CCDF of S_1 around $s = \mathbb{E}[S_1] = \mathbb{E}[S_2]$ will not be as steep as the slope of the CCDF of S_2, as shown in Figure 4.5. For operating points with $P_e \leq 0.5$, this means that link 2 is the more desirable link, as this configuration has a smaller probability of decoding error than the configuration of link 1.

4.4 Chapter Summary

In this chapter we have discussed how to apply our two-layer model to simple networks consisting of more than one point-to-point link. We have introduced a method of consolidating two fundamental network structures so that we can derive a full distribution for the arrival of DoF to the destination node from just two parameters of the PHY layer: rate and erasure probability. This will permit us to construct objective functions for the optimization of

the network in non-asymptotic scenarios where the delay constraint is small, such as video streaming or real-time data analysis. Furthermore, the PHY-layer architecture will determine a CCDF that is independent of k, preserving the decoupled nature of our two encoders. We should note, however, that our network reduction operations will not simplify arbitrary networks to a single distribution. Figure 4.6 depicts a very simple network that cannot be simplified in this way; no two links form a purely parallel or tandem structure, because of link 3. If we try to combine links 1 and 4, there is no way to account the flow of information $1 \rightarrow 3 \rightarrow 5$. No pair of links can be reduced without altering the number of DoF available at the head or tail of another link.

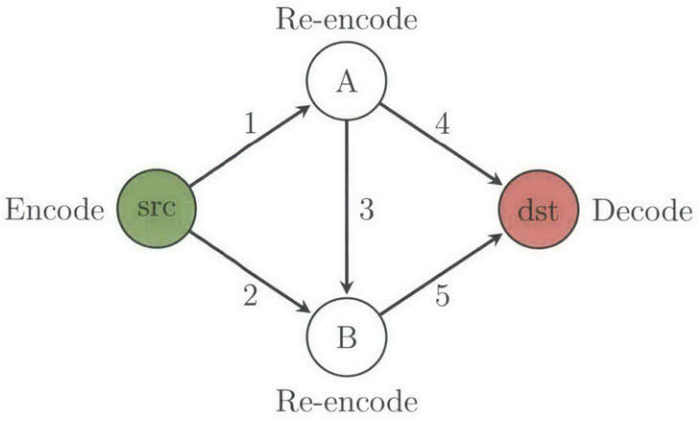

Figure 4.6: A simple network on which our network reduction operations fail

Chapter 5

Design Applications

In this chapter, we consider the design of wireless systems using our work presented in the previous chapters. We defined a model in Chapter 2, developed some approximation techniques to make analysis simpler in Chapter 3, and found a way to extend our analysis over multiple links in Chapter 4. As before, we consider the design of an encoder when there is a delay constraint. The fundamental limit of a flow is the min-cut, but just as channel capacity cannot be achieved for finite-blocklength codes, network capacity may not necessarily be achievable with finite delay constraints. The design problem we consider involves picking the best code from a limited number of options on each layer, which is a relevant problem for many wireless systems, where transceivers use channel-state information to select a transmission mode from a small number of choices. Simulating this process with our model involves optimizing for the design criterion of our choice over code rates subject to constraints imposed by the model and by the available options for codes. Decoupled encoders are a subset of joint network-channel coding, so the design problem considered here should not be confused with optimization of a two-layer encoding scheme over all codes, rates, modulation schemes, etc.; we are not investigating fundamental limits.

As in Chapter 2, (2.14), we are going to primarily consider goodput as the design objective:

$$\Gamma = k\eta\ell\left(1 - \mathbb{P}\left\{\hat{V} \neq V\right\}\right) = k\eta\ell\left(1 - P_e\left(k, n, P_p\left(\ell, p_t, p_e\left(\nu, \rho, \mathcal{H}\right)\right)\right)\right). \tag{5.1}$$

We could also design for $\mathbb{E}[D]$, the end-to-end number of DoF, but there are some issues with using this quantity as our objective. First, $\mathbb{E}[D]$ doesn't prescribe a generation size k, so it only characterizes the PHY. Second, as discussed in Section 4.3.2, the configuration achieving maximum $\mathbb{E}[D]$ may not be the configuration producing the maximum reliability for rates below capacity.

The next important question concerns which parameters we are free to choose. Again, we are concerned with scenarios involving delay constraints, so we assume some limit on τ. We also assume that our PHY-layer code has some fixed blocklength, and that network-level parameters such as packet length and percent overhead are fixed as well. For a single link, this leaves us to solve an optimization problem over two parameters: ρ and k. Over a larger network, we have to choose ρ_i for all $i \in \mathcal{E}$; for simplicity we may use the vector

$$\vec{\rho} = \langle \rho_1, \ldots \rho_{\mathcal{E}} \rangle. \tag{5.2}$$

One advantage of fixing ν is that n is determined by our choice of ρ, so that we avoid having to optimize over both n and ν.

5.1 Concavity Arguments for General Design Problem

In this thesis, the design problems we consider are small enough to solve using exhaustive search, but brute force is not a viable option when the network is comprised of many links. If the number of choices for PHY rate on link i is given by r_i, then there are on the order of $\prod_{i \in \mathcal{E}} r_i$ points that must be considered. If we hope to solve design problems for larger networks, we have to be able to use a more efficient optimization technique. The problem imposes natural integrality constraints on k and n, so we relax these constraints for all of the following arguments. In the relaxed problem, we show that Γ and $\mathbb{E}[D]$ are concave functions of k and $\vec{\rho}$ over networks that are reducible by our parallel and tandem link operations, and a good convex solver can be used the find solutions.

5.1.1 Concavity of Γ in k

We want to prove the convexity of $P_e(k, n, \xi)$. RLNC works over an erasure channel, the capacity of which is $1 - \xi$, so we will only consider rates less than this. With n fixed, this means that k is defined over the interval $[0, n(1-\xi)]$. The first step is to prove that, for any distribution on D, specified by the values of (n_i, ξ_i) pairs in the network, $P_e(k, \vec{n}, \vec{\xi})$ is convex in k.

Proposition 1. *For a distribution P_D on D that is non-decreasing in d over the interval $[0, \mathbb{E}[D]]$, $P_e(k, \vec{n}, \vec{\xi}) = \mathbb{P}\{D < k\}$ is convex in k over the support of P_D in that interval.*

Proof. Let dP_D be the Radon-Nikodym derivative of the distribution of D, and let $k_1 \leq k \leq k_2 \leq \mathbb{E}[D]$, where $k_1,\ k,\ k_2 \in \text{support}\{D\}$ and,

$$k = \theta k_1 + \bar{\theta} k_2, \quad \theta \in [0,1], \quad \bar{\theta} = 1 - \theta. \tag{5.3}$$

Because the PMF of D is non-decreasing in this interval, the average value of P_D over the range of k_1 to k will be lower than the average value of P_D over the range of k to k_2:

$$\frac{1}{k - k_1} \int_{k_1}^{k-1} dP_D \leq \frac{1}{k_2 - k} \int_{k}^{k_2 - 1} dP_D. \tag{5.4}$$

Multiplying both sides by $\frac{(k-k_1)(k_2-k)}{k_2-k_1}$, we obtain

$$\frac{k_2 - k}{k_2 - k_1} \int_{k_1}^{k-1} dP_D \leq \frac{k - k_1}{k_2 - k_1} \int_{k}^{k_2 - 1} dP_D. \tag{5.5}$$

Additionally, we may manipulate (5.3) to obtain

$$\theta = \frac{k_2 - k}{k_2 - k_1} \quad \text{and} \quad \bar{\theta} = \frac{k - k_1}{k_2 - k_1}. \tag{5.6}$$

Applying these identities to (5.5), we may obtain:

$$\theta \int_{k_1}^{k-1} dP_D \leq \bar{\theta} \int_{k}^{k_2 - 1} dP_D \tag{5.7}$$

$$(\theta + \bar{\theta}) \int_{k_1}^{k-1} dP_D \leq \bar{\theta} \int_{k_1}^{k_2 - 1} dP_D \tag{5.8}$$

$$\int_{0}^{k-1} dP_D - \int_{0}^{k_1 - 1} dP_D \leq \bar{\theta} \left(\int_{0}^{k_2 - 1} dP_D - \int_{0}^{k_1 - 1} dP_D \right) \tag{5.9}$$

$$\mathbb{P}\{D < k\} - \mathbb{P}\{D < k_1\} \leq \bar{\theta}\mathbb{P}\{D < k_2\} - (1 - \theta)\mathbb{P}\{D < k_1\} \tag{5.10}$$

$$\mathbb{P}\{D < k\} \leq \theta\mathbb{P}\{D < k_1\} + \bar{\theta}\mathbb{P}\{D < k_2\}. \tag{5.11}$$

This last statement satisfies the definition of convexity. □

Proposition 2. *Γ is concave over $k \in (0, a)$, where $a = \sum_{i \in \Pi} \{n_i(1 - \xi_i)\}$ and Π is the minimum cut-set.*

Proof. For simplicity, we use an abbreviation $P_e(k, \vec{n}, \vec{\xi}) = h(k)$.

$$\begin{aligned} \frac{\partial \Gamma}{\partial k} &= \frac{\partial}{\partial k}\left(k[1 - h(k)]\right) \\ &= 1 - h(k) - kh'(k), \end{aligned} \tag{5.12}$$

so that,

$$\frac{\partial^2 \Gamma}{\partial k^2} = -2h'(k) - kh''(k) \tag{5.13}$$

$$\leq 0. \tag{5.14}$$

Where the last inequality follows because $h'(k) \geq 0$ and $h''(k) \geq 0$, as $P_e(k, \vec{n}, \vec{\xi})$ is monotonic and convex (by Proposition 1) in k. □

We have proved that Γ is concave, but we must qualify that as concavity over the support of D. The assumptions of the proof describe only distributions where the PMF of D is strictly increasing below $\mathbb{E}[D]$. This excludes some distributions that are not pathological, or even unlikely in the system we have outlined. For example, consider the case when the framing parameter $\alpha = j > 1$ packets per codeword. Only every j^{th} integer is in the support of D, so strict monotonicity below $\mathbb{E}[D]$ does not hold over all integers. In this case, restricting a maximization problem to the subset of integers evenly divisible by j is justifiable: we will only receive packets in multiples of j.

5.1.2 Concavity of Γ in ρ, One Link

Convexity of end-to-end probability of error in the PHY-layer rate requires more examination than convexity in the NET-layer rate; first let us consider one link in isolation. Note that n increases with ρ, and p_e also increases in ρ as we trade off rate for probability of decoding error at the PHY layer. Since an increase in n will lower probability of error, while an increase in ξ will raise probability of error, it is unclear what effect an increase in ρ will have on $P_e(k, n, \xi)$. Let us characterize $n(\rho)$, $\xi(\rho)$, and $P_e(k, n, \xi)$:

1. $n(\rho)$ is affine non-decreasing in ρ once we relax the integrality constraint,
2. $\xi(\rho)$ is convex non-decreasing in ρ, owing to the time-sharing principle,
3. $P_e(k, n, \xi)$ is convex non-increasing in n and convex non-decreasing in ξ, owing to the time-sharing principle.

We expect that many explicit formulations of $P_e(k,n,\xi)$ will be jointly convex in n and ξ, but this is hard to argue based on first principles

Proposition 3. *$P_e(k,n,\xi)$ is convex in $\rho \in [0, C_{\mathcal{H}})$ for large but finite delay constraints where $\xi \leq 1/2$ and $C_{\mathcal{H}}$ is the Shannon capacity of the channel $\mathcal{H}$.*

Proof. From items 1-3 above, we obtain the following relations:

$$\begin{array}{llll} \dfrac{\partial P_e}{\partial n} \leq 0, & \dfrac{\partial P_e}{\partial \xi} \geq 0, & \dfrac{\partial n}{\partial \rho} > 0, & \dfrac{\partial \xi}{\partial \rho} \geq 0, \\ \dfrac{\partial^2 P_e}{\partial n^2} \geq 0, & \dfrac{\partial^2 P_e}{\partial \xi^2} \geq 0, & \dfrac{\partial^2 n}{\partial \rho^2} = 0, & \dfrac{\partial^2 \xi}{\partial \rho^2} \geq 0. \end{array} \tag{5.15}$$

We apply the chain rule to find the second derivative of P_e with respect to ρ:

$$\frac{\partial P_e}{\partial \rho} = \frac{\partial P_e}{\partial n}\frac{\partial n}{\partial \rho} + \frac{\partial P_e}{\partial \xi}\frac{\partial \xi}{\partial \rho}, \tag{5.16}$$

$$\frac{\partial^2 P_e}{\partial \rho^2} = \frac{\partial^2 P_e}{\partial n^2}\left(\frac{\partial n}{\partial \rho}\right)^2 + \frac{\partial P_e}{\partial n}\frac{\partial^2 n}{\partial \rho^2} + \frac{\partial^2 P_e}{\partial \xi^2}\left(\frac{\partial \xi}{\partial \rho}\right)^2 + \frac{\partial P_e}{\partial \xi}\frac{\partial^2 \xi}{\partial \rho^2} + 2\frac{\partial^2 P_e}{\partial n \partial \xi}\frac{\partial \xi}{\partial \rho}\frac{\partial n}{\partial \rho}. \tag{5.17}$$

All of the terms in (5.17) are non-negative by the relations in (5.15) except the last one; the sign of $\dfrac{\partial^2 P_e}{\partial n \partial \xi}$ is yet unknown. If n is large, a reasonable enough condition with a large delay constraint, then we can approximate error using the CLT as we did in Section 3.3.2, then we see that

$$P_e(k,n,\xi) \approx Q\left(\frac{n(1-\xi)-k}{\sqrt{n(1-\xi)\xi}}\right). \tag{5.18}$$

Since the Q-function decreases monotonically in its argument, we only need to prove for $f(n,\xi) = \dfrac{n(1-\xi)-k}{\sqrt{n(1-\xi)\xi}}$, that $\dfrac{\partial^2 f}{\partial n \partial \xi} \leq 0$. We first take the partial derivative w.r.t. n, then ξ, obtaining

$$f(n,\xi) = n^{1/2}\sqrt{\frac{1-\xi}{\xi}} - n^{-1/2}\frac{k}{\sqrt{(1-\xi)\xi}}, \tag{5.19}$$

$$\begin{aligned} \frac{\partial f}{\partial n} &= \frac{1}{2}n^{-1/2}\sqrt{\frac{1-\xi}{\xi}} + \frac{1}{2}n^{-3/2}\frac{k}{\sqrt{(1-\xi)\xi}} \\ &= \frac{1}{2}n^{-1/2}g_1(\xi) + \frac{k}{2}n^{-3/2}g_2(\xi), \end{aligned} \tag{5.20}$$

so that $\dfrac{\partial f}{\partial n}$ is the sum of two functions $g_1(\xi)$ and $g_2(\xi)$ weighted by positive coefficients. Now

we just need to find the derivaties of these two functions:

$$\begin{aligned} g_1(\xi) &= \left(\frac{1-\xi}{\xi}\right)^{1/2}, & g_2(\xi) &= \left(\xi - \xi^2\right)^{-1/2}, \\ g_1'(\xi) &= \frac{1}{2}\left(\frac{1-\xi}{\xi}\right)^{-1/2}\left(\frac{-1}{\xi^2}\right), & g_2'(\xi) &= -\frac{1}{2}(\xi-\xi^2)^{-3/2}(1-2\xi), \\ g_1'(\xi) &\leq 0, & g_2'(\xi) &\leq 0. \end{aligned} \tag{5.21}$$

Thus, $\dfrac{\partial^2 f}{\partial n \partial \xi} \leq 0$, so that $\dfrac{\partial^2 P_e}{\partial n \partial \xi} \geq 0$ and (5.17) is ≥ 0.

□

5.1.3 Concavity of Γ in ρ_i, Multiple Links

Proposition 4. *For any network $\mathcal{G}(\mathcal{V}, \mathcal{E})$ consisting of sets of parallel and tandem links, Γ is concave in $\rho_i \in [0, C_{\mathcal{H}_i})$, $\forall i \in \mathcal{E}$.*

Proof. Recall from Chapter 4, (4.27) that $\vec{\Lambda}_i^c[s] = \mathbb{P}\{S_i \geq s\}$ and, thus, for any choice of generation size k,

$$\mathbb{P}\{S_i \geq k\} = \vec{\Lambda}_i^c[k] = 1 - P_{e,i}(k, n(\rho_i), \xi(\rho_i)), \tag{5.22}$$

where $P_{e,i}(k, n(\rho_i), \xi(\rho_i))$ denotes the probability of decoding error for a single-link network equivalent to link i. Because we have shown, in the proof of Proposition 3, that $P_{e,i}(k, n(\rho_i), \xi(\rho_i))$ is convex in ρ_i, then $\vec{\Lambda}_i^c[s]$ is concave in ρ_i for any choice of k and $s \leq k$. Let us consider two links in tandem and in parallel. To make the dependencies on ρ_i explicit, let us represent $\vec{\Lambda}_1^c[s]$ by the indexed function $f_s(\rho_1)$; similarly, we say for the second link that $\vec{\Lambda}_2^c[s] = g_s(\rho_2)$.

Proving the convexity of $P_e(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho}))$ in either ρ_1 or ρ_2 is simple for tandem links; given a choice of k,

$$\begin{aligned} 1 - P_e\left(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho})\right) &= f_k(\rho_1) g_k(\rho_2), \\ P_e\left(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho})\right) &= -f_k(\rho_1) g_k(\rho_2) + 1. \end{aligned} \tag{5.23}$$

In either variable we simply multiply a concave function by a negative coefficient and add a constant, producing a convex function.

Parallel links are a bit less obvious. The number of DoF we receive is given by $D = S_1 + S_2$.

Then,

$$\begin{aligned}\mathbb{P}\{S_1 + S_2 \geq k\} &= \sum_j \mathbb{P}\{S_1 = j\}\mathbb{P}\{S_2 \geq k - j\} \\ &= \sum_j \left[\mathbb{P}\{S_1 \geq j\} - \mathbb{P}\{S_1 \geq j + 1\}\right] \mathbb{P}\{S_2 \geq k - j\} \\ &= \sum_j \left[f_j(\rho_1) - f_{j+1}(\rho_1)\right] g_{k-j}(\rho_2). \end{aligned} \tag{5.24}$$

Because the CCDFs are monotonically decreasing, $f_j(\rho_1) - f_{j+1}(\rho_1) \geq 0$ for any choice of ρ_1, thus the sum in (5.24) is the sum of concave functions, and is itself concave. Then,

$$P_e\left(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho})\right) = 1 - \mathbb{P}\{S_1 + S_2 \geq k\} \tag{5.25}$$

is convex in each ρ_i.

We derive an end-to-end probability of error by applying the reduction operations for tandem and parallel links iteratively. As we go about removing links from the network, the property of joint convexity is carried over to the resulting probability of error even as the number of links grows, so that finally we obtain

$$P_e(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho})) = 1 - \vec{\Lambda}^c_{EQ}[k], \tag{5.26}$$

which is convex in $\vec{\rho}$. Consequently,

$$\Gamma = \eta \ell k \left(1 - P_e(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho}))\right) = \eta \ell k \vec{\Lambda}^c_{EQ}[k] \tag{5.27}$$

is concave. □

Proposition 4 does not imply joint concavity in $\vec{\rho} = \langle \rho_1, \rho_2 \rangle$, but we can show that finding the maximizing point in the polyhedron $\mathcal{P} = \{\vec{\rho} : \rho_i \in [0, C_{\mathcal{H}_i}), \forall i \in \mathcal{E}\}$ can be done by finding the maximizing value of each ρ_i separately. From (5.27), this is equivalent to finding the values of ρ_i that minimize P_e. For tandem links, we obtain the gradient from (5.23):

$$\begin{aligned}\nabla P_e &= \left\langle \frac{\partial P_e}{\partial \rho_1}, \frac{\partial P_e}{\partial \rho_2} \right\rangle \\ &= -\left\langle f_k'(\rho_1) g_k(\rho_2),\ f_k(\rho_1) g_k'(\rho_2) \right\rangle. \end{aligned} \tag{5.28}$$

Because $f(\rho_1)$ and $g(\rho_2)$ are nonnegative, any value $\rho_1^* = \arg\min_{\rho_1 \in [0, \mathcal{H}_1)} P_e(k, \rho_1, \rho_2)$ is independent of ρ_2, as $f'(\rho_1^*) = 0$ or P_e will be monotonic in ρ_i, so that either $\rho_1 = 0$ or $\rho_1 = C_{\mathcal{H}_1}$ will achieve the minimum. The same argument holds for convexity in ρ_2. For parallel links, differentiating (5.24) gives the gradient

$$\nabla P_e = -\left\langle \sum_j \left[g_j(\rho_2) - g_{j+1}(\rho_2)\right] f'_{k-j}(\rho_1),\ \sum_j \left[f_j(\rho_1) - f_{j+1}(\rho_1)\right] g'_{k-j}(\rho_2) \right\rangle. \tag{5.29}$$

Again, since $g_j(\rho_2) - g_{j+1}(\rho_2)$ and $f_j(\rho_1) - f_{j+1}(\rho_1)$ must be nonnegative, we apply the same reasoning that we applied to (5.28) and see that for both operations, P_e can be maximized in ρ_1 and ρ_2 independently.

We have, then, proved the convexity of the relaxed end-to-end error probability in ρ_i and k without defining explicit functions for error probability at either layer. It is important to note that we have not proved *joint convexity*. The supremum of the support of k (erasure channel capacity) is a function of $\vec{\rho}$, which makes joint convexity difficult to prove. Consequently, using the gradient or Newton's method to minimize P_e might only produce a local minimum. However, we can still use well-known tools for convex optimization in each variable to reduce the complexity of the design problem.

5.1.4 Concavity of $\mathbb{E}[D]$

If we plan on using $\mathbb{E}[D]$ as the target function, this also has similar concavity properties as Γ; this will be easy to show, since we have already characterized the constituent terms of $\mathbb{E}[D]$.

Proposition 5. *For a network $\mathcal{G}(\mathcal{V}, \mathcal{E})$ consisting of sets of parallel and tandem links, $\mathbb{E}[D]$ is concave in $\rho_i \in [0, C_{\mathcal{H}_i})$, $\forall i \in \mathcal{E}$.*

Proof. Proposition 4 states that $P_e, (k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho}))$ is a concave function of $\rho_i \in [0, C_{\mathcal{H}_i})$, $\forall i \in \mathcal{E}$. Then from (4.28),

$$\begin{aligned} \mathbb{E}[D] &= \sum_s \vec{\Lambda}^c[s] - 1 \\ &= \sum_k \left(1 - P_e(k, \vec{n}(\vec{\rho}), \vec{\xi}(\vec{\rho}))\right) - 1. \end{aligned} \tag{5.30}$$

This is a finite sum of concave functions, and thus is itself concave. □

5.2 Design for a Single Link

We consider a single link over a BSC channel; the PHY layer uses a Reed-Solomon (RS) code to detect and correct errors, while the NET layer uses RLNC to code over erasures. In addition to channel effects, there is a steady state probability p_t that an incoming codeword is dropped owing to collision or congestion (this codeword may contain more than one packet). We assume limited access to the RS encoder: the block length is set at $\nu = 255$, but we have a choice of 7 payload sizes so that our rate can very nearly be $\rho \in \{1/8, \ldots, 7/8\}$. At the NET layer, our choice of k must be a positive integer $\leq n$. We assume a fixed value of η and we make $\ell = 255$ bytes, or 2040 bits, to keep things simple (this keeps α an integer for any choice of ρ). We let $\tau = 326,400$ channel uses[1], which allows, at the lowest RS code rate, 20 packets.

To perform the optimization, we evaluate $\max_{\rho,k} \Gamma$ at every value of δ for a BSC(δ). This maximum is not the solution to the relaxation discussed in Section 5.1; we now observe integrality constraints. We calculate Γ three times at each point, using each of the approximation techniques given in Chapter 3 to evaluate probability of error. To find the PHY-layer decoding error, we use a simple formulation for RS error: if any of the bits comprising a symbol are received in error, the symbol is received in error. Thus, symbol error probability is given by $p_s = 1 - (1-\delta)^{\log(\nu+1)}$. As given in any channel-coding reference such as [26, Sec. 6.5], probability of error is then given by

$$p_e(\nu, \rho, \delta) = \sum_{i=\left\lfloor \frac{\nu - \nu\rho}{2} \right\rfloor}^{\nu} \binom{\nu}{j} {p_s}^j (1-p_s)^{n-j}. \tag{5.31}$$

This function is a heuristic approach, based on our previous discussion. Thus, our choice of ν may not be that which yields the best Γ.

Figures 5.1, 5.2, and 5.3 track the maximizing values of ρ and k/n as functions of the channel, along with ξ and Γ. The plotted value of Γ is computed using the tight combinatorial lower bound (given in (3.9)), so that the performance of a design based on each of the approximations from Chapter 3 can be compared against the performance of a design based on a more accurate computation of goodput.

These figures indicate a system behavior that is quite different from single-layer design problems, or even cross-layer optimization problems where ρ is continuous. In the top plot of each figure, the PHY code rate behaves as expected; redundancy is increased as the channel

[1] With a double-sided 10 MHz band, this amounts to just over 65 ms.

degrades. On the second line, we see the erasure probability; the different p_t values produce different ranges of tolerable PHY-layer error rates in each of the figures. The sharp spikes right before the PHY rate switches over indicate regions where the coarse granularity of code rates impairs error-correction capability. The third line shows us how the RLNC responds to these packet erasure rates. There appears to be a "target" rate around $1 - p_t$ that is optimal when the RS code can manage its error. The last line shows that the effect of the sawing motion of the RLNC rate is to smooth goodput when the PHY transitions from one RS code to another. Notice that goodput is nearly linear in BSC crossover probability and thus roughly proportional to capacity.

Figure 5.1 is of particular interest as a special case, because here $p_t = 0$. Even when all errors are due to channel effects, NET-layer coding can compensate for poor granularity of PHY-layer rates. In effect, the use of RLNC can make a cheap off-the-shelf wireless transmitter perform like a high-end wireless transmitter.

5.3 Design for Relay

Our design problem for a relay network is very similar to the single-link scenario. Now we have a different p_t on each link. We have the same choice of 7 payload sizes for our RS code of length 255 so that our rate can very nearly be $\rho_i \in \{1/8, \ldots, 7/8\}$ on each link $i \in \{1, 2, 3\}$. At the NET layer, our choice of k must again be an integer $\leq n$. As before, η is fixed, $\ell = 2040$, but this time we change τ to 816,000[2] (this raises the number of codewords to 50 for the purpose of making the figures more readable). Instead of varying the severity of the BSC underlying each link, we fix p_t and BSC crossover probability δ on each link. In this model, most choices of p_t and δ will produce an obvious choice for each ρ_i so that designing the encoder for Γ, $\mathbb{E}[D]$ or k_ϵ^* results in the same system. However, we present results from a hand-picked case where different design objectives yield different choices of ρ_i. As before, integrality constraints are observed.

Figure 5.4 depicts the CCDFs of packet arrivals across each link for each possible choice of RS code rate. The number of the link corresponds to the numbering in Figure 4.4; that is, links 1 and 2 form a tandem set that is in parallel with link 3. The values of δ and p_t are arbitrary; we simply present a case where, based on the function for $p_e(\nu, \rho, \delta)$ given in (5.31), we see diversity in the CCDFs of D resulting from different design objectives. Notice that, over links 1 and 3, two CCDFs overlap as discussed in Section 4.3.2. Link 1 is not in

[2]For a 10 MHz double-sided band, that is 163 ms.

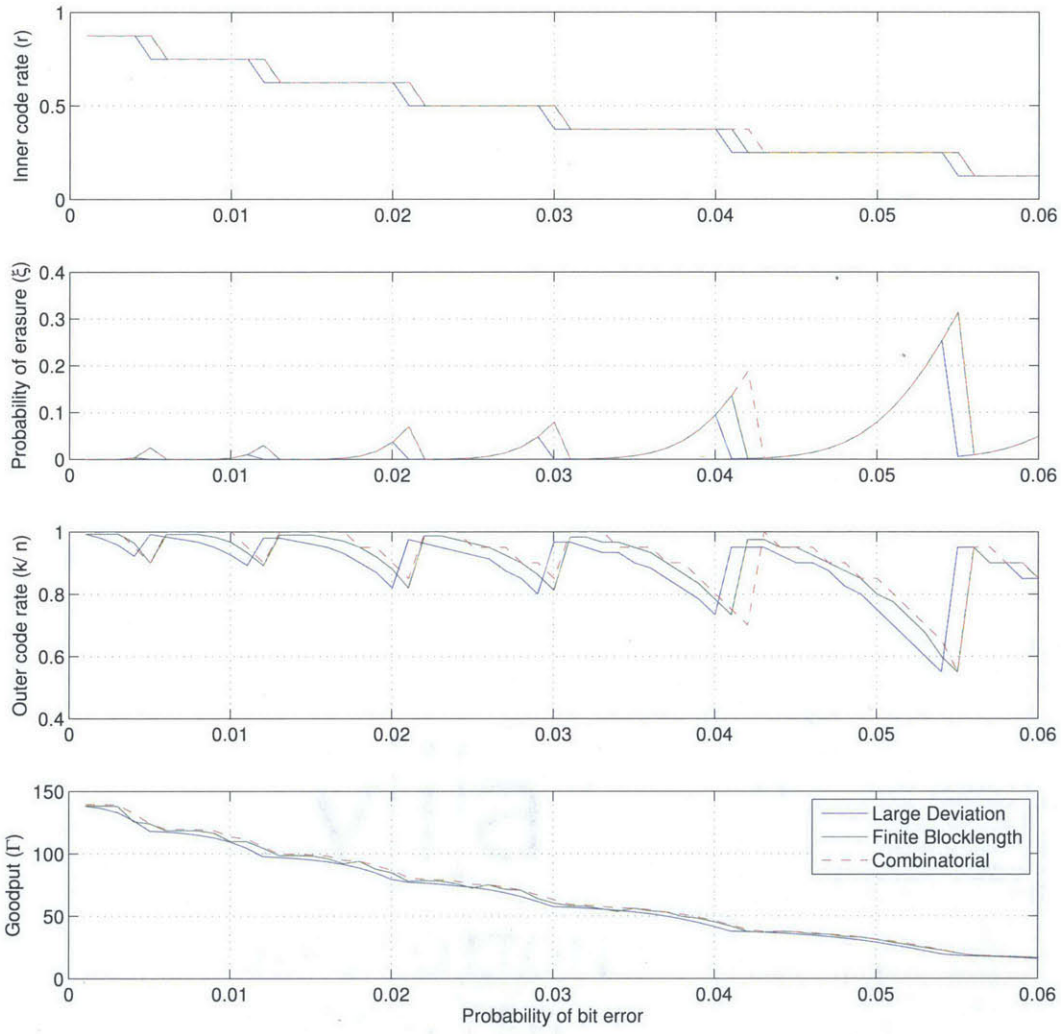

Figure 5.1: Single-link optimization results over a BSC when $p_t = 0$. The abscissa indicates the crossover probability of the underlying channel.

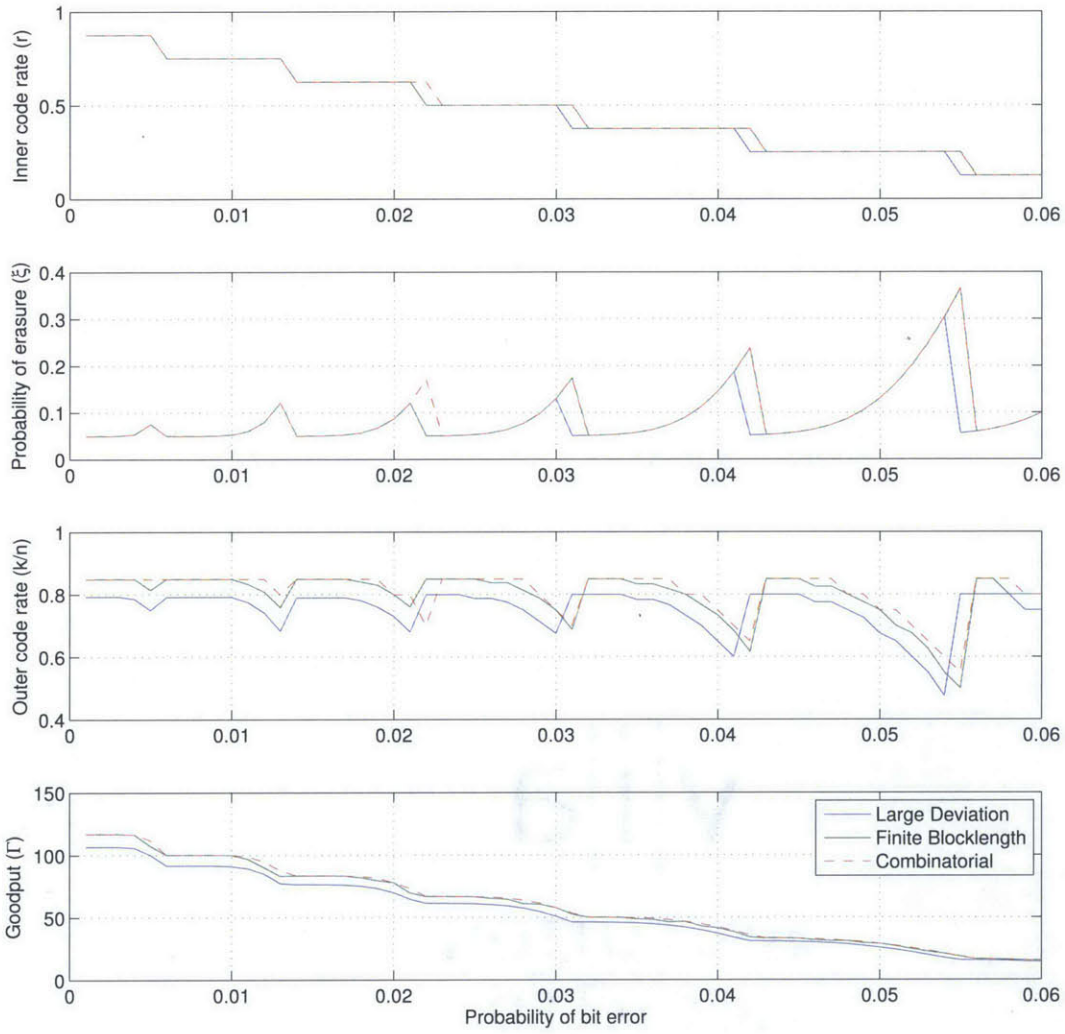

Figure 5.2: Single-link optimization results over a BSC when $p_t = 0.05$. The abscissa indicates the crossover probability of the underlying channel.

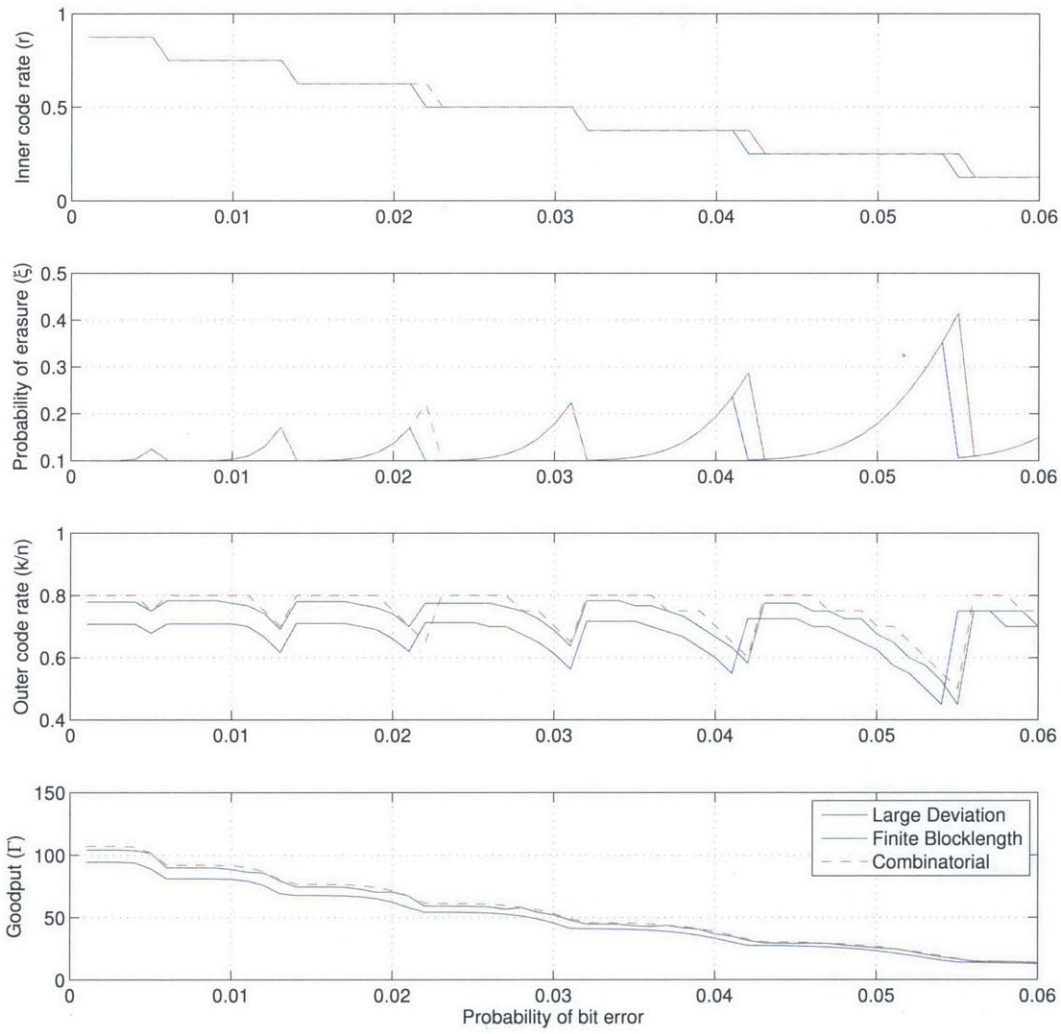

Figure 5.3: Single-link optimization results over a BSC when $p_t = 0.1$. The abscissa indicates the crossover probability of the underlying channel.

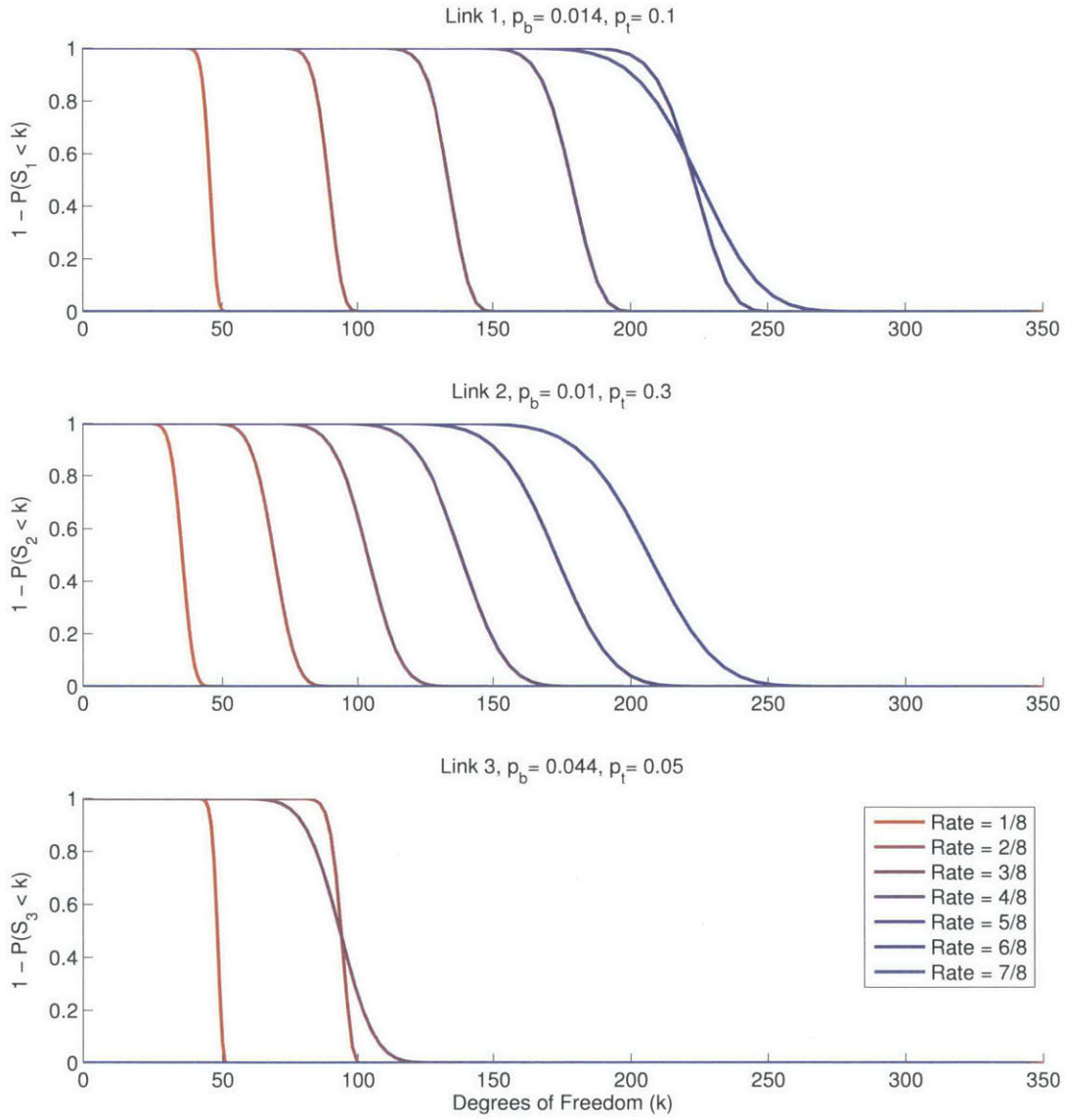

Figure 5.4: The approximate CCDFs of S_i for $i \in \{1, 2, 3\}$ for the relay optimization problem with link channels as described above each plot. These curves are approximate CCDFs because α is not necessarily 1 on any link, so the exact CCDFs are stepped, making the graphs more difficult to read. The curves shown are obtained by taking smoothing the discontinuities of each CCDF.

the min-cut, but since $\vec{\Lambda}_D^c = \vec{\Lambda}_1^c \odot \vec{\Lambda}_2^c$, the shape of the tail of the resultant CCDF will be affected by which of the two curves is selected. Additionally, notice that all of the possible CCDFs of link 2 are spaced closer together, far to the left of the same respective curves for link 1. This is because there is a high probability of packet loss over link 2, for which no choice of ρ_2 can compensate.

Turning our attention to Figure 5.5, we observe that differences in $\vec{n}$ and $\vec{\xi}$ for different choices of $\vec{\rho}$ have led to a different distribution on D depending on whether $\mathbb{E}[S_i]$, $\mathbb{E}[D]$, or Γ is to be maximized. Designing for $\mathbb{E}[S_i]$ over each link leads to high variance (and slightly higher mean) distributions of packet erasure, so the result is different from what we obtain when we design for $\mathbb{E}[D]$, which only selects the high variance option on link 3. Neither of these procedures prescribe a value for k; optimizing for Γ does prescribe a k, and the operating point maximizing Γ requires the low-variance option on both links 1 and 3; the associated resultant CCDF is again distinct.

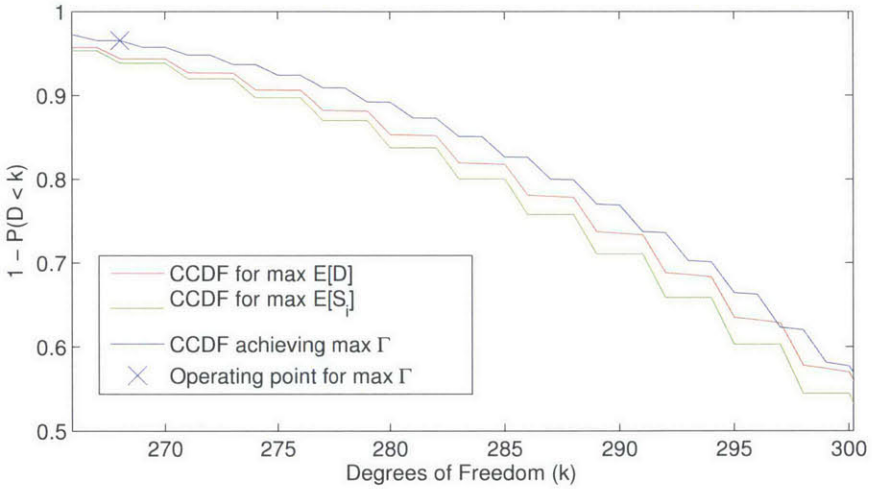

Figure 5.5: CCDFs reflecting the optimized relay code design for several different objective functions. Note that the steps in the function are uneven because the two parallel links are delivering a different number of packets in each codeword.

5.4 Chapter Summary

In this chapter, we have outlined a design problem constructed to resemble a link or set of links over which we are using limited[3] medium access control equipment with accurate

[3]Read "inexpensive".

channel state information. Using the channel state information, we pick values of ρ_i and k that give us the maximum Γ possible for this configuration.

What we have seen for this test-case problem is intuitive. Firstly, the PHY-layer code does not have to produce a probability of decoding error that is negligible; there is no need for p_e to be several orders of magnitude smaller than p_t. Secondly, the optimal NET-layer code rate does not decrease monotonically as the channel worsens; it palliates for the errors from the PHY when it has to, mitigating failure in the regions where no available choice of PHY mode performs well. Thirdly, the most impaired cut still has the most significant impact on end-to-end performance, despite the differences between our construction and the one in [19]. Finally, the second-order effects need to be considered if we are trying to satisfy an end-to-end error constraint.

Conclusion

To summarize, we proposed a model of two-layer network behavior intended to subsume many different scenarios. After that, we considered how to evaluate code performance with some analytically tractable error approximations over a single link. Expanding our scope, we introduced some tools to take these approximations and apply them to real, multi-link wireless networks[4]; these tools work well for a couple of fundamental network structures. Finally, we studied a design problem based on our model that evaluates the accuracy of our approximations and incorporates our tools for simplifying networks.

A Tale of Two Papers

In the introduction to this thesis we used two papers, [2] and [3] by Berger et. al. and Courtade and Wesel, respectively, to illustrate how "optimal" cross-layer behavior depends heavily on what the model used and the notion of optimality. Let us see if, using the contents of this thesis, we can give heuristic explanations for why these two papers reveal seemingly conflicting descriptions of optimal behavior.

Similarly to this thesis, most of the discussion in [3] involves a constraint on transmission time for a chunk of information, but in the section on asymptotic results, the authors conclude that, "rateless packet-level codes (e.g. Raptor codes) are not well-suited for fading channel applications". This conclusion is reached by minimizing transmit power given rate and reliability constraints - relating this to our setup in Chapter 5, the problem would be to find the worst BSC possible at which we could achieve some minimum rate and maximum probability of error - and letting the delay constraint go to infinity (and rate constraint to zero). Because the optimal error-correction code rate is bounded away from zero while the PHY rate goes to 0, Courtade and Wesel conclude that additional redundancy should be

[4]Although, there is nothing fundamentally flawed about using them for wireline networks, so long as RLNC is being used.

added to the PHY code instead of the NET code.

This conclusion is in line with our intuition about cross-layer systems. In the asymptotic regime, the rate of PHY decoding errors will be small; as long as the code is below capacity, probability of error will decrease as blocklength increases. Packet losses are only being generated by PHY-layer decoding errors, so as long as the rate of these errors is small, an erasure code will not produce any substantial performance gain.

By contrast, in the abstract of [2], Berger et. al write, "For severe fading channels, such as Rayleigh fading channels, the tradeoff leans towards more redundancy on erasure-correction coding across packets, and less so on error-correction coding within each packet." The authors consider a different optimization problem: maximizing reliability given a rate constraint. This optimization procedure is performed numerically. As the fading becomes worse (somewhat analogous to the deterioration of our channel), more redundancy is allocated to the NET-layer code. Again, we can reconcile this with our observations in Section 5.2. As fading worsens, the achievable region of rate/reliability pairs shrinks, so the minimum achievable probability of decoding error increases for any rate constraint. When the PHY rate is constrained so that the minimum p_e is on the order of 10^{-1}, it becomes more effective to give up on the PHY-layer code and compensate with the NET-layer code, as we observed in the sawing motion in Figures 5.1-5.3.

If we could isolate the central point of the last 5 chapters, it would be this: operate a given code in the steep portion of the curve characterizing the trade-off between throughput and probability of error. Shannon said that capacity is the border between reliable and unreliable communication; as we scale back blocklengths from infinity, we see that the transition between regions of high and low probability of error is sharp, but not discontinuous. There is a region where a small decrease in throughput corresponds with a large decrease in probability of error; coding on two layers allows us to slightly shift this region for any given code.

Appendix:
Proof of Converse for m-EC Capacity in the Finite Block-Length Regime

To prove the converse, we will first need the following lemma, which is an adaptation of [24, Theorem 38].

Lemma 1. *For a m-EC with erasure probability ξ, the average error probability of an (n, M, ϵ) code satisfies*

$$\epsilon \geq \sum_{s=\lfloor n-\log_m M \rfloor+1}^{n} \binom{n}{s} \xi^s (1-\xi)^{n-s} \left(1 - \frac{m^{n-s}}{M}\right) \tag{1}$$

even if the encoder knows the location of the erasures non-causally.

Proof. If we have a uniform distribution on channel inputs $A \in \{1, \ldots, M\}$, channel outputs $B \in \{1, \ldots, J\}$, and decoder decision $C \in \{1, \ldots, M\}$, then we can bound the average probability of correctly decoding a message,

$$\mathbb{P}\{C = A\} \leq \frac{J}{M}, \tag{2}$$

as in [24, (188)-(191)], which holds regardless of the symbol alphabet.

If the encoder knows *a priori* the location of erasures, an observed codeword with z erasures can take on m^{n-z} possible values. Thus, the probability of error given z erasures is lower bounded by $\max\{0, 1 - \frac{m^{n-z}}{M}\}$. Each message can be represented as $\log_m M$ symbols over a field of cardinality m, so an error only occurs $z \geq \lfloor n - \log_m M \rfloor + 1$. Since erasures follow a binomial distribution, (1) follows. □

We also state the Berry-Esseen Theorem used in [24].

Theorem 1 (Berry-Esseen). *Let $X_i, i = 1, \ldots, n$ be independent with*

$$\mu_i = \mathbb{E}[X_i] \tag{3}$$

$$\sigma_k^2 = \mathrm{Var}[X_i] \tag{4}$$

$$t_i = \mathbb{E}[|X_i - \mu_i|^3] \tag{5}$$

$$\sigma^2 = \sum_{i=1}^{n} \sigma_i^2 \tag{6}$$

$$T = \sum_{i=1}^{n} t_i. \tag{7}$$

Then for any $-\infty < \lambda < \infty$

$$\left| \mathbb{P}\left\{ \sum_{i=1}^{n} (X_i - \mu_i) \leq \lambda\sigma \right\} - Q(\lambda) \right| \leq \frac{6T}{\sigma^3}. \tag{8}$$

Additionally, we state (584) from [24], which says that for all $s = 0, \ldots, n$

$$\binom{n}{s} \xi^s (1-\xi)^{n-s} \leq \frac{G_1}{\sqrt{n}}, \tag{9}$$

where the existence of a constant G_1 is guaranteed by [27]. Now we are ready to prove the converse.

Poof of converse for (3.33) *for an m-EC.* Note that Lemma 1 holds for any (n, M, ϵ) code. Therefore, selecting an M that violates Lemma 1 will give us an upper bound on M^*. From (9), we see that,

$$\begin{aligned}
\sum_{s=\lfloor n-\log_m M \rfloor+1}^{n} \binom{n}{s} \xi^s (1-\xi)^{n-s} m^{n-s-\log_m M} &\leq \sum_{s=\lfloor n-\log_m M \rfloor+1}^{n} \frac{G_1}{\sqrt{n}} m^{n-s-\log_m M} \\
&= \frac{G_1 m^n}{M\sqrt{n}} \sum_{s=\lfloor n-\log_m M \rfloor+1}^{n} m^{-s} \\
&= \frac{G_1 m^n}{M\sqrt{n}} \left(\sum_{s=0}^{n} m^{-s} - \sum_{s=0}^{\lfloor n-\log_m M \rfloor} m^{-s} \right) \\
&= \frac{G_1 m^n}{M\sqrt{n}} \left(\frac{m^{-\lfloor n-\log_m M \rfloor} - m^{-n}}{m-1} \right) \\
&\leq \frac{G_1 m^n}{M\sqrt{n}} \left(\frac{m^{-n+\log_m M+1} - m^{-n}}{m-1} \right) \\
&= \frac{G_1}{M\sqrt{n}} \left(\frac{mM-1}{m-1} \right) \\
&= \frac{G_1}{\sqrt{n}} \left(\frac{m - \frac{1}{M}}{m-1} \right) \\
&\leq \frac{2G_1}{\sqrt{n}}.
\end{aligned} \tag{10}$$

Based on (10), we choose to define an M such that

$$\log_m M = n(1-\xi) - \sqrt{n\xi(1-\xi)}Q^{-1}\left(\epsilon + \frac{B+3G_1}{\sqrt{n}}\right), \tag{11}$$

where B is the Berry-Esseen constant associated with a binomial distribution, which is to say that $\frac{B}{\sqrt{n}}$ is the R.H.S of (8) when $X_i \sim Bern(\xi)$. To be more explicit, if S is the number of erasures, then from (8), we obtain

$$|\mathbb{P}\{S - n\mu \leq \lambda\sigma\} - Q(\lambda)| \leq \frac{B}{\sqrt{n}} \triangleq \frac{6T}{\sigma^3} = \frac{n6t_i}{\sqrt[3/2]{n\xi(1-\xi)}}. \tag{12}$$

For this proof, the actual value of t_i is irrelevant. We simply note that it is constant for choice of ξ. For our definition of S and choice of M,

$$\begin{aligned}
\sum_{s=\lfloor n-\log_m M\rfloor}^{n} \binom{n}{s}\xi^s(1-\xi)^{n-s} &\geq \mathbb{P}\{S \geq n - \log_m M\} \\
&= \mathbb{P}\{S - n\mu \geq n(1-\xi) - \log_m M\} \\
&= \mathbb{P}\left\{S - n\mu \geq n(1-\xi) - \log_m M\right\} \\
&= \mathbb{P}\left\{S - n\mu \geq \sqrt{n\xi(1-\xi)}Q^{-1}\left(\epsilon + \frac{B+3G_1}{\sqrt{n}}\right)\right\} \\
&= \mathbb{P}\left\{S - n\mu \geq \sigma\lambda\right\},
\end{aligned} \tag{13}$$

where $\sigma = \sqrt{n\xi(1-\xi)}$ and $\lambda = Q^{-1}\left(\epsilon + \frac{B+3G_1}{\sqrt{n}}\right)$. From (12), we obtain

$$-\mathbb{P}\{S - n\mu \leq \lambda\sigma\} + Q(\lambda) \leq \frac{B}{\sqrt{n}}. \tag{14}$$

Applying this to (13), we verify that

$$\sum_{s=\lfloor n-\log_m M\rfloor}^{n} \binom{n}{s}\xi^s(1-\xi)^{n-s} \geq \epsilon + \frac{3G_1}{\sqrt{n}}. \tag{15}$$

Further bounding (15) with (10), we obtain

$$\sum_{s=\lfloor n-\log_m M\rfloor+1}^{n} \binom{n}{s}\xi^s(1-\xi)^{n-s} \geq \epsilon + \frac{G_1}{\sqrt{n}} + \sum_{s=\lfloor n-\log_m M\rfloor+1}^{n} \binom{n}{s}\xi^s(1-\xi)^{n-s}m^{n-s-\log_m M}. \tag{16}$$

Combining the summations we are left with an extra term, so that

$$\sum_{s=\lfloor n-\log_m M\rfloor+1}^{n}\binom{n}{s}\xi^s(1-\xi)^{n-s}\left(1-\frac{m^{n-s}}{M}\right)\geq\epsilon+\frac{G_1}{\sqrt{n}}-\left[\binom{n}{s}\xi^s(1-\xi)^{n-s}\right]_{s=\lfloor n-\log_m M\rfloor}.$$

Applying (9) again produces

$$\sum_{s=\lfloor n-\log_m M\rfloor}^{n}\binom{n}{s}\xi^s(1-\xi)^{n-s}\left(1-\frac{m^{n-s}}{M}\right)\geq\epsilon, \tag{17}$$

which violates Lemma 1. Thus, we conclude that

$$\log_m M\leq n(1-\xi)-\sqrt{n\xi(1-\xi)}Q^{-1}\left(\epsilon+\frac{B+mG_1}{\sqrt{n}}\right) \tag{18}$$

$$=n(1-\xi)-\sqrt{n\xi(1-\xi)}Q^{-1}(\epsilon)+O(1), \tag{19}$$

where (19) follows from Taylor's formula.

□

Bibliography

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[2] C. R. Berger, S. Zhou, Y. Wen, P. Willett, and K. Pattipati, "Optimizing joint erasure- and error-correction coding for wireless packet transmissions," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4586–4595, 2008.

[3] T. A. Courtade and R. D. Wesel, "Optimal allocation of redundancy between packet-level erasure coding and physical-layer channel coding in fading channels," *IEEE Transactions on Communications*, vol. 59, no. 8, pp. 2101–2109, 2011.

[4] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, 2003.

[5] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE Transactions on Networking*, vol. 11, no. 5, pp. 782–795, 2003.

[6] T. Ho, M. Médard, R. Koetter, M. Effros, D. R. Karger, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4413 – 4430, 2006.

[7] N. Cai and R. W. Yeung, "Network coding and error correction," in *Proceedings of the IEEE Information Theory Workshop*, 2002.

[8] M. Luby, "LT codes," in *Proceedings of the IEEE Symposium on Foundations of Computer Science*, 2002.

[9] M. Ghaderi, D. Towsley, and J. Kurose, "Reliability gain of network coding in lossy wireless networks," in *Proceedings of IEEE INFOCOM*, 2008.

[10] A. Eryilmaz, A. Ozdaglar, M. Médard, and E. Ahmed, "On the delay and throughput gains of coding in unreliable networks," *IEEE Transactions on Information Theory*, vol. 54, no. 12, pp. 5511–5524, 2008.

[11] A. Rezaee, F. du Pin Calmon, L. M. Zeger, and M. Médard, "Speeding multicast by acknowledgment reduction technique (smart) enabling robustness of qoe to the number of users," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 7, 2012.

[12] A. Barg, J. Justesen, and C. Thommesen, "Concatenated codes with fixed inner code and random outer code," *IEEE Transactions on Information Theory*, vol. 47, no. 1, pp. 361 – 365, 2001.

[13] M. Vehkaperä and M. Médard, "A throughput-delay trade-off in packetized systems with erasures," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pp. 1858–1862, 2005.

[14] R. G. Gallager, *Information Theory and Reliable Communication.* New York, NY: Wiley, 1968.

[15] C. Koller, M. Haenggi, J. Kliewer, and D. J. Costello, "On the optimal block length for joint channel and network coding," in *IEEE Information Theory Workshop 2011*, pp. 528–532, 2011.

[16] B. T. Swapna, A. Eryilmaz, and N. B. Shroff, "Throughput-delay analysis of random linear network coding for wireless broadcasting," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6328 – 6341, 2013.

[17] P. Wu and N. Jindal, "Coding versus ARQ in fading channels: How reliable should the PHY be?," *IEEE Transactions on Communications*, vol. 59, no. 12, pp. 3363 – 3374, 2011.

[18] A. F. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of wireless erasure networks," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 789 – 804, 2006.

[19] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 3, no. 20, 2008.

[20] B. Haeupler and M. Médard, "One packet suffices - highly efficient packetized network coding with finite memory," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pp. 1151 – 1155, 2011.

[21] M. Xiao, M. Médard, and T. Aulin, "Cross-layer design of rateless random network codes for delay optimization," *IEEE Transactions on Communications*, vol. 59, no. 12, pp. 3311 – 3322, 2011.

[22] G. Liva, E. Paolini, and M. Chiani, "Performance versus overhead for fountain codes over $\mathbb{F}_q$," *IEEE Communications Letters*, vol. 14, no. 2, pp. 178 – 180, 2010.

[23] M. Hayashi, "Information spectrum approach to second-order coding rate in channel coding," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 4947 4966, 2009.

[24] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.

[25] V. Strassen, "Asymptotische absch atzugen in Shannon's informationstheorie," *Proceedings 3rd Conference in Information Theory, Prague*, pp. 689 – 723, 1962.

[26] S. Lin and D. J. Costello, *Error Control Coding.* Englewood Cliffs, New Jersey: Prentice-Hall, 1983.

[27] C.-G. Esseen, "On the concentration function of a sum of independent random variables," *Z. Wahrscheinlichkeitstheorie und Verw. Geb.*, vol. 9, no. 4, pp. 290–4966, 1968.