A DIGITAL INTERFACE FOR WIRELESS NETWORKS

BY

ANAND MURALIDHAR

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2011

Urbana, Illinois

Doctoral Committee:

Professor P. R. Kumar, Chair
Professor Bruce Hajek
Professor Pierre Moulin
Assistant Professor Todd P. Coleman

ABSTRACT

This dissertation addresses the problem of determining the capacity of wireless networks and how to operate them. Within this context we present results on Gaussian relay, interference, and multicast networks.

Two new models for wireless networks are introduced here: the discrete network and the superposition network. As with a Gaussian network, one can construct either a discrete network or a superposition network. The discrete network is obtained by simply quantizing the received signals in the Gaussian model and by restricting the transmit signals to a finite alphabet. The superposition network, inspired by the Gaussian model, is a noiseless deterministic network, the inputs and outputs of the channels are discrete, and channel gains are signed integers.

The capacity of a class of Gaussian relay networks and its corresponding discrete or superposition network is always within a bounded gap, where the gap is independent of channel gains or signal-to-noise ratio (SNR), and depends only on the number M of nodes in the network. More importantly, a near-optimal coding strategy for either the discrete or the superposition network can be converted into a near-optimal coding strategy for the original Gaussian network. Hence, both these networks serve as near-optimal digital interfaces for operating the Gaussian network.

The discrete network is obtained from a Gaussian network by simply quantizing the received signals and restricting transmitted signals to a certain finite precision. Since its signals are obtained from those of a Gaussian network and its transmissions are transmittable as-is on a Gaussian network, the discrete network provides a particularly simple quantization-based digital interface for operating layered Gaussian relay networks. These are relay networks in which the nodes are grouped into layers, and only nodes of one layer can transmit to the nodes of the next layer. The cut-set upper bounds on the capacities of the Gaussian and the discrete network are within an SNR-independent bounded gap of $O(M \log M)$ bits. Moreover, a simple linear network code is a near-optimal coding strategy for the discrete relay network, achieving all rates within $O(M^2)$ bits of its cut-set bound, where the bound is independent of channel gains or SNR. The linear code can be used as-is on the Gaussian network after quantizing its received signals. It achieves all rates within $O(M^2)$ bits of the capacity for Gaussian relay networks. The linear network code improves on existing approximately-optimal coding schemes for the relay network by virtue of its simplicity and robustness, and it explicitly connects wireline network coding with codes for Gaussian networks.

The approximation of Gaussian networks by other previously proposed deterministic networks is also studied in this dissertation, and two main results are presented, one positive and the other negative. The gap between the capacity of a Gaussian relay network and a corresponding linear deterministic network can be unbounded. The key reasons are that the linear deterministic model fails to capture the phase of received signals, and there is a loss in signal strength in the reduction to a linear deterministic network. On the positive side, Gaussian relay networks with a single source-destination pair are indeed well approximated by the superposition network. The difference between the capacity of a Gaussian relay network and the corresponding superposition network is bounded by $O(M \log M)$ bits, where the gap is again independent of channel gains or SNR. As a corollary, multiple-input multiple-output (MIMO) channels cannot be approximated by the linear deterministic model but can be by the superposition model. A code for a Gaussian relay network can be designed from *any* code for the corresponding superposition network simply by pruning it, suffering no more than a rate loss of $O(M \log M)$ bits that is independent of SNR.

Similar results hold for the $K \times K$ Gaussian interference network, MIMO Gaussian interference networks, MIMO Gaussian relay networks, and multicast networks, with the constant gap depending additionally on the number of antennas in case of MIMO networks.

To my adviser and my family

ACKNOWLEDGMENTS

This dissertation is a culmination of four years of work on problems of a fairly abstract nature, but with a rich history. As any Ph.D. student will confirm, working on such problems can be a daunting prospect, especially if one begins the graduate program as a bumbling student. This is further compounded by the highs and lows of research, where a bright sunny week can be followed by several months of the dour midwestern winter, both literally and metaphorically. In the midst of all these uncertainties, I was lucky to have an adviser whom I could always look up to for inspiration or encouragement. Professor P. R. Kumar has been an inspiring figure in my life, with his constant encouragement, genuine concern, subtle humor, and an incredible ability to immediately identify the crux of a research problem. It is often said that asking the right question is a critical component of research, and Prof. Kumar, a true master of this art, has tried his best to pass on this skill to the rest of us. Working with Prof. Kumar is one of the highlights of my Ph.D. experience and he has had a strong positive influence on me, which I hope to carry forward in my life.

I would like to thank my committee members, Professors Hajek, Moulin, and Coleman for agreeing to be a part of the committee, finding time in their schedules for my defense, and for their encouraging words and suggestions during the preliminary exam and later. I would also like to thank Professors Meyn, Moulin, and Coleman for their encouragement during my qualifying exam; they certainly gave me the confidence to carry on with my research. I thank all the faculty in the Coordinated Science Lab and in other departments with whom I have had an opportunity to interact. I am also grateful to the staff at Coordinated Science Lab, especially Jana Lenz and Dan Jordan, for their patience and help with all aspects of life in the Coordinated Science Lab.

Few people have mattered more to me over the last fours years than my friends. After spending time with them at such close quarters, it is hard to distinguish between them and my family. I will always cherish the jokes, long lunches, volley games, movies, and the endless hours spent in their company. VJ, Hemant, Gayu, Sreeram, Rajan, Sathe, JK, RG, Kunal, Anjan, Jagan, Vivek, Sindy, Amruta, and Arti — life in Champaign would be a dull affair without any of you.

I am really grateful to my Macbook for withstanding the vagaries of my behavior and for silently tolerating the coffee I spilt on it. The silence is especially deafening since its fan has stopped working. My car has been an excellent companion for the last couple of years, and I will be parting with her with a very heavy heart. I feel that the unnatural sounds that emanate from her interiors are her way of reaching out to me. Also, a big thanks to my home of the last two years, my den at 507. It has played a crucial role in keeping me motivated towards the end, with the afternoon naps and the evening chai, and its proximity to CSL.

I would like to thank my parents, family, and Rexy. My parents have been an integral part of my journey with their unconditional emotional support. It would have been difficult to cope with the tough times in my life if they were not by my side, and I am grateful to them for their confidence in my abilities. Rexy, though you will not be able to read this, there were few days in the last four years when I did not think of you.

I consider myself very fortunate to have had this opportunity to complete the Ph.D. program at the University of Illinois, and I express my deepest gratitude to all the people who have helped me in this journey. Finally, I would like to thank the eternal spirit that guides us all, and helps us in making the right choice.

The material in this dissertation is based upon work partially supported by NSF under contracts ECCS-0701604, CNS-07-21992, CNS-0626584, and CNS-05-19535, CCF-0939370, and CNS-0905397, AFOSR under contract FA9550-09-0121, and USARO under contracts

W911NF-08-1-0238 and W-911-NF-0710287. I would like to thank the above agencies for supporting my research. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the above agencies.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Computing the capacities of wireless networks is a formidable problem. It has been a central theme of network information theory, where research has attempted to characterize the information-theoretic limits of data transmission in various abstract models of networks. Of equal importance, and also of great interest, has been the problem of construction of optimal coding schemes that achieve the promised rates. However, except for some examples of networks such as the MAC channel [1], [2] and the broadcast channel [3], [4], it has so far not been possible to accurately compute the capacities of even simple networks such as the relay channel [5], [6] involving a small number of nodes.

While the information theoretic capacity indicates a limit on the rates of communication in the system, it does not say anything about the optimal strategy for operating these networks. Even in a simple wireless network, there are a multitude of options and techniques available for cooperation among nodes, and so it is of great interest to determine optimal or provably near-optimal operating strategies for wireless networks of interest. This logjam in designing strategies for wireless networks has motivated the work in [7] where the stringent goals of network information theory are relaxed by allowing for answers which are within a bounded gap of the capacity. By bounded gap is meant a constant that is independent of the channel gains or SNR, and is a function of only the number of nodes in the network.

So motivated, we address the issues both of approximating the capacity of relay and interference networks, and of developing and rigorously establishing near-optimal coding strategies to operate the networks. The starting point of the work in this dissertation is the Gaussian model for wireless networks, and its aim is to develop other models that can serve as digital interfaces for operating Gaussian networks. We desire that the new models have two properties. First, they must well approximate the capacity of the Gaussian network up to a bounded gap; i.e., the capacity of the Gaussian network and the capacity of the network constructed with the new model must be within a bounded gap. Second, there must exist a procedure to adapt near-optimal coding strategies for the new models to derive near-optimal coding strategies for the original Gaussian network. Such a near-optimal code will achieve the capacity of the original Gaussian network to within a bounded gap, with the gap independent of channel gains or SNR. A bounded gap approximation is valid at all SNRs and is most relevant in the high power or low noise regimes.

We introduce two channel models for wireless networks that have both these properties. The first model, the *discrete model*, is derived from the Gaussian model and is obtained by simply quantizing the received signals and by restricting the transmit signals to a finite alphabet. The second model is a noiseless deterministic model and is called the *superposition* model. This model is inspired by the Gaussian model, and the transmit and received signals in this model are also restricted to a finite discrete set. We can construct either a discrete or a superposition network that corresponds to a Gaussian network. Both well approximate a class of Gaussian networks in the sense described above.

In addition to approximating capacity, it is also of primary interest to determine near-optimal coding strategies. The above program of approximation also extends to coding schemes. The discrete network provides a quantization-based digital interface for the class of layered Gaussian networks. In order to operate the Gaussian network on the discrete interface, one simply quantizes the received signals and restricts the transmit signals to a finite alphabet. We construct a simple coding scheme for the layered discrete relay network called the *linear network code*, which achieves all rates within $O(M^2)$ bits of the capacity of the discrete network, when there are M nodes in the network. This linear network code is also near-optimal for the layered Gaussian network and can be used on it operating through the discrete interface. The linear network code is a generalization of random linear network coding for wireline networks [8, 9], or for linear deterministic networks [7]. To the best of our knowledge, this is the first instance of an explicit connection between network coding for wireline and noiseless networks and network coding for Gaussian networks, with a bounded gap guarantee on performance.

The superposition model also serves as a surrogate for Gaussian networks in a stronger sense. One can rigorously *lift* any coding scheme from the superposition network to a corresponding coding scheme for the Gaussian network without more than a bounded loss in the rate. Such a lifting procedure approximation can potentially simplify the construction of near-optimal coding schemes for Gaussian relay and interference networks. This approximation permits one to address the alternative, possibly simpler problem of finding the capacity of the deterministic network and subsequently lifting it to obtain a near-optimal coding scheme for the original Gaussian relay network of interest.

1.1 Dissertation outline

In Chapter 2, we summarize earlier work on relay networks and interference networks. There is a rich body of work on both these networks and we mention some of the relevant results. We survey the work on approximating the capacities of relay networks and on designing near-optimal coding strategies. The capacity and coding results in network coding are also surveyed, and in Chapter 5, we relate network coding for wireline networks with coding for Gaussian networks. We survey the results in approximating the capacity region of the 2×2 interference channel. We also mention some results on the capacity region of large interference networks.

In Chapter 3, we describe the various models used in this dissertation. There are two aspects of a wireless network that must be modeled. First, we describe the network model, which is a graph that captures the connectivity in the network. We consider two main types of networks: relay networks and interference networks. In relay networks there is one source, many relays, and a single destination. In interference networks there are many source-destination pairs. We also define multicast networks. Then we model the wireless channel. We describe a variety of channel models for the various impairments in a wireless network. Some of these are noisy models, i.e., the received signal is corrupted by the noise, while others are noiseless deterministic models where the received signal is a deterministic function of the transmitted signals. All the models are either derived from or inspired by the Gaussian model. We introduce two new models in this dissertation: the discrete model and the superposition model. Both these models serve as near-optimal digital interfaces for operating certain classes of Gaussian networks, though the discrete model is a more natural interface since its signals are obtained by simply quantizing the signals in the Gaussian network. We can construct a discrete network or a superposition network corresponding to every Gaussian network. We also define the linear deterministic model [7] in this chapter.

Chapters 4, 5, and 6 deal with relay networks. In Chapter 4, the cut-set upper bound is defined, which is a fundamental limit on the capacity of a relay network. The cut-set bounds of the Gaussian, discrete, and superposition networks are shown to be within a bounded gap of $O(M \log M)$ bits when the total number of nodes in the network is M. The constant in the bounded gap is independent of channel gains or SNR, and only a function of the number of nodes in the network. These results are relevant since the cut-set bound for the Gaussian relay network is approximately achievable [7]. In later chapters, it is proved that it is approximately achievable for layered discrete relay networks and for superposition relay networks. We also prove that the difference between the cut-set bound of the Gaussian and the corresponding linear deterministic network can grow with SNR. Since the cut-set bound is the capacity of linear deterministic relay networks, it follows that the linear deterministic model cannot approximate the capacity of Gaussian networks.

In Chapter 5, the discrete model is shown to provide a quantization-based digital interface for layered Gaussian networks. First, layered relay networks are defined, and a coding scheme is constructed for layered discrete relay networks that involves linear operations over the binary vectors describing the discrete model. This linear network code achieves rates within $O(M^2)$ of the cut-set bound, and hence within $O(M^2)$ of the capacity of the network. It is proved that the linear code can be used on the layered Gaussian network by simply quantizing the received signals in the Gaussian network and restricting the transmit signals to a finite set, i.e., by operating the Gaussian network on the digital interface defined by the discrete model. The linear code is also near-optimal for the layered Gaussian relay network, and achieves rates within $O(M^2)$ of the cut-set bound. The $O(M^2)$ bound is independent of channel gains or SNR. We extend the results to the MIMO and multicast counterparts, with the bounded gap then additionally depending on the number of antennas for MIMO networks.

In Chapter 6, we prove that the superposition model is a near-optimal digital interface for operating Gaussian relay networks. We prove that any coding scheme for the superposition network can be "lifted" to the Gaussian network simply by adjoining codewords to create long codewords, and then pruning the set of long codewords. The loss in the rate of the code due to this lifting is shown to be at most $O(M \log M)$ bits. We also prove that the capacities of the Gaussian relay network and its superposition counterpart differ by at most $O(M \log M)$ bits. This implies that a near-optimal coding scheme for the superposition network can be lifted to obtain a near-optimal coding scheme for the Gaussian relay network. We also prove similar results for MIMO relay networks and multicast networks.

Chapter 7 develops a digital interface for interference networks. We show that the capacity regions of the $K \times K$ Gaussian interference channel and the superposition interference channel are within a bounded number of bits, independent of channel gains or SNR. Also, similarly to the case of the relay network, we develop a systematic way to lift any code for the superposition interference network to the Gaussian interference network, and establish that it does so with no more than a bounded loss in the rate. We also extend results to the MIMO counterpart of the interference network.

We conclude in Chapter 8 by summarizing the results of approximating Gaussian networks with the discrete and superposition models.

Results in this dissertation have previously appeared in [10], [11], and [12]. Some of the results are also present in [13].

CHAPTER 2

SUMMARY OF PREVIOUS WORK

In this chapter, we survey the previous research on relay networks, network coding, and interference networks.

2.1 Relay networks

The relay channel was introduced by Van der Muelen [5], and certain achievable rates were determined. Cover and El Gamal [6] subsequently studied the network in detail and developed two coding schemes which are now known as decode-and-forward and compress-and-forward. They also provided an upper bound on the capacity of the relay channel, referred to as the cut-set bound, which was developed in full generality for networks in [14].

The decode-and-forward scheme was extended to networks with many relays in [15] and [16], and compress-and-forward was extended to larger networks in [17]. All these papers introduced certain novelties in existing coding schemes, though the best upper bound on the capacity of relay networks continues to be the cut-set bound. These schemes do not generally provide a guarantee of near-optimality, in the sense that the gap between the achievable rates for any of these coding schemes and the cut-set bound could be arbitrarily high as a function of SNR. So, in effect, it has not been clear whether it is the cut-set bound that is weak or whether it is the coding schemes that are not exploiting all the features available in the wireless network.

This impasse motivated the work of [7], which studied coding schemes with provable guarantees on performance. This was done through the approach of introducing a *linear* *deterministic model*, where the linearity is over a finite field. The linear deterministic model captures the broadcast and interference aspects of wireless networks. The capacity of linear deterministic relay networks was determined to be precisely the same as the cut-set bound, which was additionally shown to be achievable by random coding at the nodes. Further, coding schemes were developed for general deterministic relay networks in [7], where the received signal at a node is a function of the signals transmitted by the neighboring nodes. The coding schemes for the linear deterministic model motivated the construction of a coding scheme for Gaussian relay networks in [7] called *quantize-map-and-forward* (QMF) in which the relays quantize the received signals, buffer them, and randomly map them to a Gaussian codeword. QMF is provably near-optimal for relay networks in the sense that it achieves all rates within a bounded gap of $O(M \log M)$ bits from the cut-set bound. This result also establishes as a corollary that the cut-set bound for the Gaussian relay network is indeed approximately the capacity of the Gaussian relay network, up to a bounded gap of $O(M \log M)$ bits.

However, the above-mentioned results do not establish the closeness of the capacities of the linear deterministic network and the Gaussian network. In fact, the linear deterministic model does not approximate the capacity of Gaussian networks in general (see [7] and Section 3.2.2). The question that therefore arises is whether there is a procedure for constructing for every Gaussian relay network a corresponding deterministic network in such a way that the capacities of the Gaussian and the deterministic network are within a bounded gap. This has been done in [7] and is shown in this dissertation via different models. In the truncated deterministic model in [7], the channel gains and inputs are complex valued, while the complex channel output has integer real and imaginary parts. The model used here, the superposition model, is a somewhat more discrete model in the sense that channel gains and inputs are discrete valued.

The QMF scheme for the Gaussian network is certainly inspired by the coding scheme for the linear deterministic counterpart. But there is no rigorous procedure to design a coding scheme for the Gaussian network given a coding scheme for the linear deterministic network, and consequently also no procedure for lifting codes from one to the other while preserving near-optimality. Hence it is unclear if a deterministic model only serves to aid the intuition in the construction of coding schemes for Gaussian networks, or if there is a more rigorous fundamental connection between the capacities of the deterministic network and the Gaussian network. More importantly, it is not clear if a coding strategy for a deterministic model can be explicitly used for designing a coding strategy for the Gaussian network that achieves comparable performance. We answer these questions in the affirmative and show that the superposition model does indeed serve as a digital interface for operating Gaussian relay and interference networks, and codes can be simply lifted to the Gaussian network from its superposition counterpart. The study of coding schemes for the superposition network inspired the linear network code for relay networks presented here.

The bounded gap approximation for capacity of relay networks was further improved in [18] where the authors constructed an extension of compress-and-forward to relay networks that is approximately optimal. Here the relays are required to perform vector quantization in order to compress their received signals. In [19], the minimal compression rates for the relay nodes were computed, and the decoding procedure from [18] was further simplified.

The capacity results for relay networks have spurred research in finding low complexity coding schemes. In [20], the authors modify the QMF scheme by choosing low-density parity-check codes for encoding at the source and the relay instead of Gaussian codes. They present a simplified decoding algorithm based on Tanner graphs and show with simulations the viability of the proposed technique. In [21], a different approach is taken by constructing codes that are computationally tractable when compared to QMF. A concatenated code is presented for the relay network where the outer code is a polar code and the inner code is a modification of the random Gaussian code from [7]. This approach is shown to have computational complexity that is near-linear in the block-length of the code. In [22], the quantization and encoding in QMF is modified by using nested lattice codes at the source and relays.

In recent work in [23], the digital interface defined by the superposition model defined in this dissertation is used to develop a near-optimal coding scheme for a wireless broadcast network. In a wireless broadcast network, the source sends independent information to multiple destinations, with the relays facilitating the process of communication. The authors first solve the problem for the broadcast superposition network, lift it to the Gaussian network, and combine it with a Marton outer code.

2.2 Network coding

Network coding was introduced in the landmark paper [8]. The max-flow min-cut theorem was established for wireline networks with a single source and multiple destinations, which essentially implied that the cut-set bound was the capacity of these networks. The authors proved that a class of codes called β-codes, which involve random encoding at the source and intermediate nodes, achieve the capacity of the network with increasing block-length. There were other important contributions in [8]: namely, the notion of distinguishability of codewords at the destination, which naturally introduces the notion of cuts in proving the achievability of the cut-set bound, and the technique of time-parameterizing cyclic networks to view them as special cases of acyclic networks. Both these techniques were used in the proofs in [7] and are used in the proof of approximate optimality of linear network codes in this dissertation.

Later, in [24] and [25], it was established that linear network codes suffice to achieve the capacity of these networks. In [24], the authors introduced algebraic tools that simplified the analysis and design of network codes. In [9], it was shown that random encoding at the source and random linear operations at the relay nodes achieves the capacity of the network in the limit of increasing block-length of the codewords. In [7], the applicability of random linear coding to linear deterministic networks was shown, where random encoding by the source and random linear encoding by the relays achieves the cut-set bound in the limit as the block-length tends to infinity.

2.3 Interference networks

Interference networks have received much attention recently, following the results in [26]. In [26], the capacity of the interference channel with two transmitters and two receivers is determined to within a constant gap of 1 bit. The near-optimal coding scheme in [26] is a specific choice among the myriad strategies proposed in [27]. A simpler proof of the result in [26] is provided in [28]. This was independently strengthened in [29], [30], and [31] where treating the interference as noise is shown to be capacity achieving for a restricted range of the parameters corresponding to a low interference regime. The capacity region of the 2×2 deterministic interference channel was determined in [32]. In [33], it is shown that the capacities of the linear deterministic interference channel and the Gaussian interference channel are within a bounded gap. A variant of the discrete superposition model was first used in [33] in a sequence of networks that reduced the 2×2 Gaussian interference channel to a linear deterministic interference channel.

Much less is known for Gaussian interference networks with more than two users. The number of degrees of freedom of the time-varying interference channel was characterized in [34] using the idea of interference alignment, and they were characterized for specific interference channels with fixed gains in [35]. The generalized degrees-of-freedom region of the fully symmetric many-user interference channel was computed in [36]. In general, the capacity region of the interference networks with three or more users is unknown, even to within a constant gap.

CHAPTER 3

MODELS FOR WIRELESS NETWORKS

In this chapter, we develop models for analyzing the wireless networks of interest. There are two aspects of a wireless network that we need to model; first, we need to model the connectivity of the network by a graph, and second, we need to model the wireless channel. We consider two types of networks depending on whether the network has one or many source-destination pairs. We describe a variety of channel models that capture the various impairments in a wireless network.

3.1 Network models

The wireless network is represented by a directed graph $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of nodes, and the directed edges in $\mathcal{E}$ correspond to wireless links. All the links are assumed to be of a unidirectional nature, where the direction of communication between two nodes is indicated by the edge connecting them. We consider two types of networks in this dissertation, relay networks and interference networks. These two networks differ in their underlying directed graphs.

3.1.1 Relay networks

Relay networks have a single source and a single destination, and a collection of relay nodes that are facilitators in the process of communication. Here the set of nodes is labeled as $\mathcal{V} = \{0, 1, \ldots, M\}$, where node 0 is the source node and node M is the destination. Nodes $1, 2, \ldots, M-1$ are the relay nodes and participate in the process of communication by cooperating with the source and with each other to allow the highest rate of communication from the source to the destination. An example of a relay network is shown in Figure 3.1.

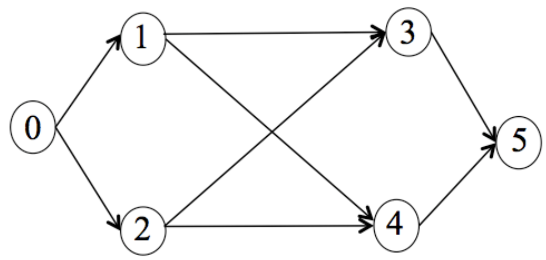


Figure 3.1: A relay network.

Multicast networks

Multicast networks have a single source node that wants to transmit the same information to a subset of the nodes in the network. The remaining nodes in the network act as relays and facilitate the communication of information. The set of intended destinations is labeled as $\mathcal{D}$ and $\mathcal{D} \subseteq \mathcal{V}$.

3.1.2 Interference networks

In an interference network, there are many sources and destinations. Specifically, we consider $K \times K$ interference networks which consist of K source-destination pairs. We divide the nodes in $\mathcal{V}$ into two sets, the first set $\{1, 2, \ldots, K\}$ consisting of the transmitters and the second set $\{1, 2, \ldots, K\}$ consisting of the receivers. Though we use the same numbers to denote the sources and the destinations, it will be clear from the context which node we are referring to. All the sources are connected to all the destinations, i.e., a particular destination receives signals from the intended source, but the received signal is corrupted by transmissions from the remaining sources in the network. In Figure 3.2, we have an example of a 3×3 interference network with 3 sources and 3 destinations.

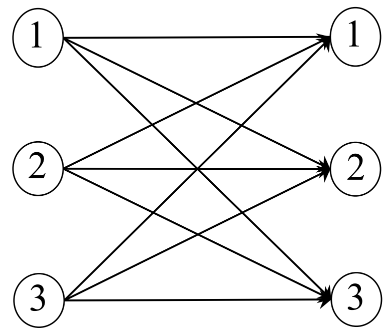


Figure 3.2: 3×3 interference network.

3.2 Channel models

Next we need a mathematical model to describe the transmitted and received signals at every node. We describe various channel models, some of which are noisy where the received signal is corrupted with noise, while others are noiseless models. All the channel models are either derived from or inspired by the Gaussian model which is described in Section 3.2.1.

3.2.1 Noisy channel models

The models described in this section are the Gaussian model, which is perhaps the most popular mathematical model for information-theoretic analysis of networks, and the discrete model, which is introduced in this dissertation as a model for a wireless network.

Gaussian model

Let the complex number x_i denote the transmission of node i and let y_j denote the received signal at the j-th node. Every node has an average power constraint, assumed to be 1 for simplicity. The channel between nodes i and j is parametrized by a complex number $h_{ij} = h_{ijR} + \imath h_{ijI}$. The channel gain is fixed for the complete duration of communication. The received signal at node j is

$$y_j = \sum_{i \in \mathcal{N}(j)} h_{ij} x_i + z_j, \tag{3.1}$$

where $\mathcal{N}(j)$ is the set of the neighbors of node j, and z_j is $\mathcal{CN}(0,1)$, complex white Gaussian noise independent of the transmit signals.

We will also allow for MIMO relay networks where the nodes have multiple transmit and receive antennas. Suppose node i has T_i transmit antennas and U_i receive antennas. In that case, the transmitted signal at each time instant at node i is a T_i-dimensional vector of complex numbers, and the received vector is U_i-dimensional. Correspondingly, the channel between two nodes i and j is then described by a collection of channel gains $\{h_{ij}^{kl}\}$ where k and l index, respectively, the transmit antennas at node i and the receive antennas at node j. The received signal at node j is

$$y_j^l = \sum_{i \in \mathcal{N}(j)} \sum_{k=1}^{T_i} h_{ij}^{kl} \, x_i^k + z_j^l, \;\; l = 1, \ldots, U_j, \tag{3.2}$$

where x_i^k is the signal transmitted from the k-th antenna of node i, and y_j^l is the received signal at the l-th antenna of node j. The term z_j^l is the complex Gaussian noise added to the received signal at the j-th node. For simplicity of exposition, we restrict a transmit signal to satisfy an individual power constraint, though our results can be extended to allow for a total power constraint across all the antennas of a node.

Discrete model

Now we describe a *quantization-based digital interface* for operating the above Gaussian network, i.e., for the purpose of defining the coding strategy. It is obtained, as the phrase above suggests, by quantizing the continuum valued signals received from the Gaussian network, and by restricting the choice of transmitted signals to lie in a discrete set. The quantization of the received signals is natural, though we will even discard sign information. We will call the overall network resulting from the discrete-inputs and discrete-outputs at each node as the *discrete network*.

In the language of automata theory, this discrete network can be *simulated* [37] from the Gaussian network, i.e., it uses less information than the original Gaussian network. The *discrete network* is obtained by quantizing the received signals and constraining the transmit signals in a Gaussian network. The received and transmit signals are allowed to take finitely many values lying in what can essentially be regarded as a quadrature amplitude modulation (QAM) constellation. Define n to be

$$n := \max_{(i,j) \in \mathcal{E}} \max\{\lfloor \log |h_{ijR}| \rfloor, \lfloor \log |h_{ijI}| \rfloor\}. \tag{3.3}$$

The channel inputs (transmit signals) in the discrete network are complex valued, with both real and imaginary parts taking values from 2^n equally spaced discrete points. The transmit symbol is

$$x = \frac{1}{\sqrt{2}}(x_R + \imath x_I), \tag{3.4}$$

where

$$x_R = \sum_{k=1}^{n} 2^{-k} x_R(k), \tag{3.5}$$

$$x_I = \sum_{k=1}^{n} 2^{-k} x_I(k), \tag{3.6}$$

with each $x_R(i)$ and $x_I(j)$ in $\mathbb{F}_2$. The symbol x can be equivalently represented by the $2n$-bit binary tuple

$$(\underline{x}_R, \underline{x}_I) = (x_R(1), x_R(2), \ldots, x_R(n), x_I(1), x_I(2), \ldots, x_I(n)). \tag{3.7}$$

Note that the above channel inputs satisfy a unit energy constraint at each discrete-time, and are therefore valid inputs even for the Gaussian network with a unit power constraint. This property will be helpful in proving the approximate optimality of the linear network code.

The channel gains are unchanged from the Gaussian model. As in the Gaussian model, the channel between two nodes i and j in the discrete network simply multiplies the input x_i by the corresponding channel gain h_{ij}. At a receiver, the received signal is defined through the composition of the following operations:

- First add all the incoming signals by the standard summation over $\mathbb{C}$.
- Then discard the signs of the real and imaginary parts.
- Further discard the fractional portions of the real and imaginary parts and retain only the integer portion.
- Then quantize the integer portion of the real and imaginary parts by truncating their binary expansions to n bits of precision, where n is as defined in (3.3).

Thus $y = y_R + \imath y_I$ is the received signal at a node in the Gaussian model, and we denote the binary expansions of the integer parts of $|y_R|$ and $|y_I|$ by $\sum_{k=1}^{\infty} 2^k y_R(k)$ and $\sum_{k=1}^{\infty} 2^k y_I(k)$, respectively. The received signal in the discrete network is then

$$y^{'} := [y] := \left(\sum_{k=1}^{n} 2^k y_R(k)\right) + \imath \left(\sum_{k=1}^{n} 2^k y_I(k)\right). \tag{3.8}$$

As with the transmit signals, $y^{'}$ can be equivalently described by the $2n$-bit binary tuple

$$(\underline{y}_R^{'}, \underline{y}_I^{'}) = (y_R(1), y_R(2), \ldots, y_R(n), y_I(1), y_I(2), \ldots, y_I(n)). \tag{3.9}$$

We will use the compact notation $[\cdot]$ to represent the overall quantization operation:

$$y_j^{'} := [y_j] := [\sum_{i \in \mathcal{N}(j)} h_{ij} x_i + z_j]. \tag{3.10}$$

It is important to note that each received signal in the discrete network can be obtained from the corresponding received signal in the Gaussian network by performing elementary quantization operations (when their transmitted signals are identical, as we intend to be the case). In fact, since the transmit signals in the discrete network are valid transmit signals for the Gaussian network, we use the same notation for the transmit signals in both models.

In a similar way, one also obtains a discrete MIMO network corresponding to a MIMO Gaussian relay network. In the MIMO discrete network, each transmit and receive antenna can be treated as a virtual node. As before, every transmit and received signal (corresponding to every transmit or receive antenna) is quantized to lie in a finite set, and the granularity of the quantization will take into account all the channel gains between various antennas in the network. The transmit signals lie in a finite set and can be described by a $2n_{MIMO}$-bit tuple. The receive signals are quantized and described by a $2n_{MIMO}$-bit tuple, where

$$n_{MIMO} := \max_{(i,j)\in\mathcal{E}} \max_{\substack{k=1,\ldots,T_i,\\ l=1,\ldots,U_j}} \{\lfloor \log |h_{ijR}^{kl}| \rfloor, \lfloor \log |h_{ijI}^{kl}| \rfloor\}. \tag{3.11}$$

3.2.2 Deterministic channel models

In a deterministic model, the received signal is a deterministic function of the transmit signals. The received signal at the j-th node is

$$y_j = f_j(\{x_i\}_{i\in\mathcal{N}(j)}), \tag{3.12}$$

where the function f_j is determined by the deterministic channel model. The superposition model and the linear deterministic model are examples of noiseless deterministic models for wireless networks. These models capture the broadcast and interference aspects of the wireless channel. The broadcast nature of the wireless medium requires a node to transmit the same signal on all its outgoing links, though the signal may be attenuated differently on various links depending on the channel gains. Due to the interference in a wireless network, the received signal at a node is the superposition of all the signals transmitted by its neighbors, with the transmit signals attenuated by the corresponding channel gains.

The deterministic models presented next incorporate these two fundamental aspects of the wireless medium.

Superposition model

The superposition model is similar in nature to the discrete model. A complex transmit symbol is given by $x = \frac{1}{\sqrt{2}}(x_R + \imath x_I)$, where

$$x_R = \sum_{k=1}^{n} 2^{-k} x_R(k), \tag{3.13}$$
$$x_I = \sum_{k=1}^{n} 2^{-k} x_I(k), \tag{3.14}$$

with each $x_R(i)$ and $x_I(j)$ in $\mathbb{F}_2$ and n defined in (3.3).

The real and imaginary parts of channel gains from the Gaussian model are *quantized* to integers by neglecting their fractional parts. The quantized channel gain for link (i, j) is given by

$$\hat{h}_{ij} := \text{sign}(h_{ijR})\lfloor|h_{ijR}|\rfloor + \imath\,\text{sign}(h_{ijI})\lfloor|h_{ijI}|\rfloor, \tag{3.15}$$

with the quantization operation $\hat{\cdot}$ as indicated above. Hence quantization of the channel gain discards the fractional portions of its real and imaginary parts. The channel between two nodes in the superposition network multiplies the input by the corresponding channel gain and quantizes the product by neglecting the fractional components of both real and imaginary parts, i.e., it forms $\widehat{\hat{h}_{ij} x_i}$. The outputs of all incoming channels at a receiver node lie in $\mathbb{Z} + \imath\mathbb{Z}$. All the quantized outputs are added up at a receiver by the standard summation over $\mathbb{Z} + \imath\mathbb{Z}$. The received signal at node j is given by

$$y_j^{''} := \sum_{i \in \mathcal{N}(j)} \widehat{\hat{h}_{ij} x_i}. \tag{3.16}$$

This model retains the essential superposition property of the wireless channel. Quantization of channel coefficients does not substantially change the channel matrix for large values of the SNR. Also, the effect of noise is captured by constraining the inputs to positive fractions that can be represented by finite bits and by quantization of the channel output. An important property of the superposition model is that transmit signals in it satisfy a unit peak power constraint, and are thus also valid for transmission in the Gaussian network. That is, encoder outputs in the superposition network can also be used in the Gaussian network.

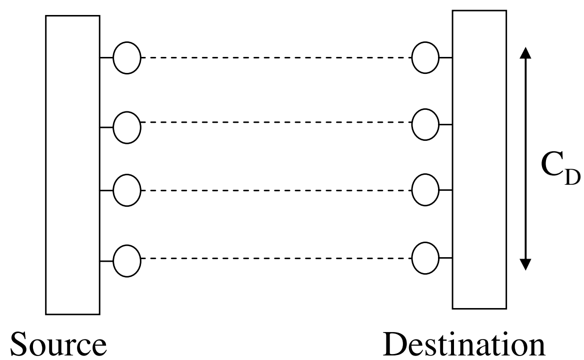
Linear deterministic model

The linear deterministic model was introduced in [7] as an approximate model to capture certain aspects of wireless networks. It should be noted that the linearity is with respect to a finite field. We introduce the model for a point-to-point Gaussian channel and later extend it to general networks.

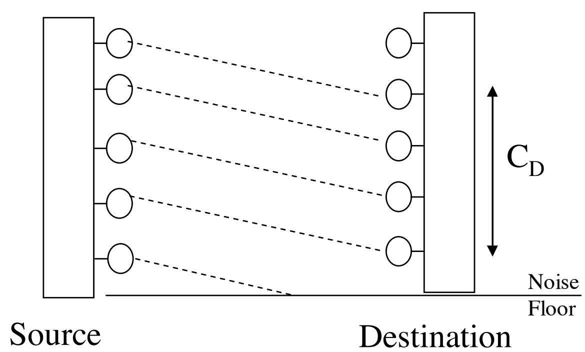
Consider a simple AWGN channel

$$y \;=\; hx + z,$$

where x satisfies an average power constraint $E[|x|^2] \;\leq\; 1$, and z is complex Gaussian noise $\mathcal{CN}(0,1)$. The capacity of this channel is $C \;=\; \log(1+|h|^2)$. Let $C_D \;:=\; \lfloor \log |h|^2 \rfloor$ approximately denote its capacity in the high SNR regime. In Figure 3.3(a), we construct a linear deterministic network of capacity C_D with a source that transmits C_D bits that are noiselessly received at the destination. For this example, C_D is equal to 4. Alternately, we can construct a linear deterministic network where the source transmits a binary vector x (of length at least C_D with bits ordered from left to right) and a channel that attenuates the signal by allowing C_D most significant bits to be received at the destination (see Figure 3.3(b)). Both the networks in Figure 3.3 are equivalent representations of the linear



(a) Noiseless channel of capacity C_D bits.



(b) Channel allows only C_D bits.

Figure 3.3: Linear deterministic model: Two viewpoints.

deterministic model for the point-to-point Gaussian channel.

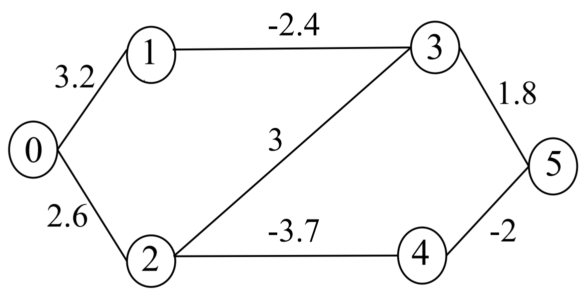
In a general Gaussian network, choose all the inputs and outputs of channels to be binary vectors of length $\max_{(i,j)\in\mathcal{E}} \lfloor \log |h_{ij}|^2 \rfloor$. Each link with channel gain h is replaced by a matrix that shifts the input vector and allows $\lfloor \log |h|^2 \rfloor$ most significant bits of the input to pass through. At a receiver, shifted vectors from multiple inputs are added bit by bit over the binary field. This models the partially destructive nature of interference in wireless networks. Modeling the broadcast feature of wireless networks, a node transmits the same vector on all outgoing links, albeit with different attenuation, and the number of significant bits arriving at a receiver depends on the channel gain. As an example, consider the Gaussian network in Figure 3.4(a) and the corresponding linear deterministic network in Figure 3.4(b). Since the magnitude of the largest channel gain is 3, all inputs and outputs of channels are vectors in $\mathbb{F}_2$ of length $\lfloor \log 3.7^2 \rfloor = 3$. The transmitted signals of nodes 1 and 2 interfere at node 3, and the received signal at node 3 is given by

$$y = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} x_1 + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x_2 \tag{3.17}$$

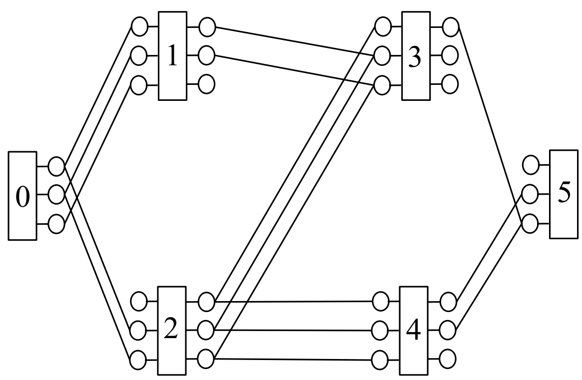
The channel is simply a linear transformation over the binary field. In the networks in Figure 3.4, nodes 1 and 2 can listen to the source's transmission, though the received signal is attenuated differently at both nodes. The corresponding received signals are

$$y_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x_0, \tag{3.18}$$

$$y_2 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} x_0. \tag{3.19}$$



(a) Example of a Gaussian network.



(b) Corresponding linear deterministic network.

Figure 3.4: Linear deterministic model of a Gaussian network.

CHAPTER 4

THE CUT-SET BOUND ON THE CAPACITY OF RELAY NETWORKS

The information-theoretic capacity of a network is determined in two steps; first by establishing a converse and proving a upper bound on the capacity, and second by constructing a coding scheme for the network which achieves any rate below the upper bound. This approach has been very successful for the point-to-point channel, but has had limited success for general networks. The rates achievable in networks via coding strategies are usually much lower than the upper bounds on the capacity. This is especially true for relay networks where the only known upper bound on the capacity is the cut-set bound and, for a long time, the rates achievable by all known coding schemes were arbitrarily lower than the cut-set bound. Recently it was shown in [7] that the cut-set bound is indeed the capacity of the relay network up to a constant gap of $O(M \log M)$ bits, with the bounded gap valid for all SNRs. In this chapter, we define the cut-set bound and show that the cut-set bounds of the Gaussian, discrete, and superposition networks are within a bounded gap of each other. All the cut-set bounds differ from each other by at most $O(M \log M)$ bits, where the bounded gap is independent of channel gains or the SNR. On the contrary, the linear deterministic network cannot provide this kind of approximation for Gaussian networks since the difference in the cut-set bounds of the Gaussian and the corresponding linear deterministic networks can be unbounded. These results have direct implications on approximating the capacity of the Gaussian relay networks with these models since, as we will see later, the cut-set bounds for certain classes of all the networks are achievable, either exactly or within an SNR-independent bounded gap.

4.1 The cut-set bound for relay networks

The cut-set bound was introduced in [14] as an upper bound on the information-theoretic capacity of a network. It is a generalization of the max-flow min-cut theorem for networks and bounds the flow of information across the network. The cut-set bound is applicable to general networks with multiple sources and destinations. Since we are mainly concerned with relay networks with a single source-destination pair, we state the bound for this specific class of networks.

The cut-set bound on the capacity C of a relay network with source 0, destination M, channel inputs $\{x_i\}$, and channel outputs $\{y_j\}$ is

$$C \leq \max_{p(x_0,x_1,\ldots,x_{M-1})} \min_{\Omega \in \Lambda} I(x_\Omega; y_{\Omega^c} | x_{\Omega^c}), \tag{4.1}$$

where Λ is the set of all partitions of $\mathcal{V}$ with $0 \in \Omega$ and $M \in \Omega^c$. Additionally, in the Gaussian model, the inputs must satisfy an average power constraint, $E[|x_i|^2] \leq 1, \forall i$. The cut-set bound is a fundamental limit on the capacity of the network. Evaluating the above bound only requires knowledge of the transition probability matrix

$$p(y_1, y_2, \ldots, y_M \mid x_0, x_1, \ldots, x_{M-1}) \tag{4.2}$$

describing the input-output relationships in the networks. This bound can be evaluated for any of the channel modes presented in Section 3.2.

In the case of multicast networks, let $\mathcal{D}$ be the set of nodes that are the intended recipients of the source's message. Then the cut-set bound for such networks is given by

$$C \leq \max_{p(x_0,x_1,\ldots,x_{M-1})} \min_{D \in \mathcal{D}} \min_{\Omega \in \Lambda_D} I(x_\Omega; y_{\Omega^c} | x_{\Omega^c}), \tag{4.3}$$

where Λ_D is the set of cuts in the network that separate the source from the destination D.

4.1.1 Cut-set bounds for Gaussian networks

In Gaussian networks, the jointly Gaussian input distribution maximizes the cut-set bound in (4.1). We simplify the computation of the bound by choosing the inputs to be independent and identically distributed Gaussian random variables. Choosing $\{x_i\}$'s to be i.i.d. $\mathcal{CN}(0,1)$ weakens the bound in (4.1) by $O(M)$ bits for any choice of channel gains and yields

$$\min_{\Omega \in \Lambda} I(x_\Omega; y_{\Omega^c} | x_{\Omega^c}) = \min_{\Omega \in \Lambda} \log |I + \mathcal{H}_\Omega \mathcal{H}_\Omega^\dagger|, \tag{4.4}$$

where $\mathcal{H}_\Omega$ is the transfer matrix of the MIMO channel corresponding to cut Ω (see [7]).

4.1.2 Cut-set bounds for deterministic networks

Since outputs are a function of inputs in a deterministic network,

$$\begin{aligned} I(x_\Omega; y_{\Omega^c} | x_{\Omega^c}) &= H(y_{\Omega^c} | x_{\Omega^c}) - H(y_{\Omega^c} | x_\Omega, x_{\Omega^c}) && (4.5) \\ &= H(y_{\Omega^c} | x_{\Omega^c}). && (4.6) \end{aligned}$$

Hence, for deterministic networks, the cut-set bound in (4.1) reduces to

$$C \leq \max_{p(x_0, x_1, \ldots, x_{M-1})} \min_{\Omega \in \Lambda} H(y_{\Omega^c} | x_{\Omega^c}). \tag{4.7}$$

The above bound is applicable for both superposition networks and linear deterministic networks. However, for the specific case of linear deterministic networks, we can further simplify the bound. For linear deterministic networks, the maximum value of the conditional entropy $H(y_{\Omega^c} | x_{\Omega^c})$ equals the rank of the transfer matrix $\mathcal{G}_{\Omega, \Omega^c}$ associated with the cut Ω [7], where the rank is determined over an appropriate finite field. An optimal input distribution is to choose input variables independent and uniformly distributed over the underlying field.

Hence (4.7) simplifies to

$$C \;\leq\; \min_{\Omega}\ \text{rank}\ \mathcal{G}_{\Omega,\Omega^c}. \tag{4.8}$$

For example, in Figure 3.4, rank of $\mathcal{G}_{\Omega,\Omega^c}$ over $\mathbb{F}_2$, with $\Omega = \{1, 2\}$, is 5 with

$$\mathcal{G}_{\Omega,\Omega^c} \;=\; \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{bmatrix}.$$

4.2 Relationships between various cut-set bounds

In this section, we prove a bounded-gap between the cut-set bounds of the Gaussian, discrete, and the superposition networks. We also prove that the gap between the cut-set bounds of the Gaussian and the linear deterministic networks can grow unbounded with SNR.

4.2.1 Cut-set bounds of Gaussian and discrete networks

Let CS_G be the cut-set bound of the Gaussian network and let CS_D be the cut-set bound of the corresponding discrete network.

Lemma 1. *The cut-set bounds of the Gaussian network and the corresponding discrete network are within a constant gap of $O(M \log M)$ bits,*

$$|CS_G - CS_D| \;=\; O(M \log M), \tag{4.9}$$

with the gap independent of channel gains or SNR.

Proof. The lemma is proved in two steps: $CS_G \ \geq \ CS_D$ in Section 4.2.1 and $CS_G - CS_D \ = \ O(M \log M)$ in Section 4.2.1. The procedure to prove both inequalities are the same. We consider a particular cut Ω in the network and choose an input distribution for this cut in one of the models. Then, in a series of steps, we transform the channel inputs and outputs of the cut to the corresponding channel inputs and outputs in the other model. We then bound the loss in the mutual information in this transformation. Repeating this procedure across all the cuts in the network completes the proof.

$CS_G \ \geq \ CS_D$

The transmit signals in the discrete network are a strict subset of the valid inputs for the Gaussian network, and the received signals in the discrete network are obtained by quantizing the corresponding received signals in the Gaussian network. Hence, as noted above, any operation in the discrete network can be simulated on the Gaussian network.

Choose any input distribution for the transmit signals in the discrete relay network. Retain the same distribution for the inputs in the Gaussian network. Since the received signals in the discrete network are obtained by quantizing the received signals in the Gaussian network, for any cut Ω, by the data processing lemma [38],

$$I(x_{\Omega}; y_{\Omega^c} | x_{\Omega^c}) \ \geq \ I(x_{\Omega}; y'_{\Omega^c} | x_{\Omega^c}). \tag{4.10}$$

This proves that $CS_G \geq CS_D$.

$CS_G - CS_D \ = \ O(M \log M)$

We show that the mutual information across a cut in the discrete network is at least as high as the mutual information across the same cut in the Gaussian network, up to $O(M \log M)$ bits.

Step 1: Cut-by-cut analysis

The cut-set bound for the Gaussian network is

$$
\begin{aligned}
CS_G &= \max_{p(x_0,x_1,\ldots,x_{M-1})} \min_{\Omega\in\Lambda} I(x_\Omega; y_{\Omega^c}|x_{\Omega^c}) && (4.11)\\
&\leq \min_{\Omega\in\Lambda} \max_{p(x_0,x_1,\ldots,x_{M-1})} I(x_\Omega; y_{\Omega^c}|x_{\Omega^c}). && (4.12)
\end{aligned}
$$

We consider a particular cut Ω/Ω^c in the network. The received signal at the j-th node is given by

$$y_j = \sum_{i\in N(j)} h_{ij}x_i + z_j. \tag{4.13}$$

The mutual information across the cut, $\mathcal{I}_1 := I(x_\Omega; y_{\Omega^c}|x_{\Omega^c})$, is maximized by the choice of jointly Gaussian inputs.

Step 2: Positive fractional inputs

Instead of the optimal joint Gaussian distribution for the inputs that maximizes $\mathcal{I}_1$, we choose a different input distribution. Consider an input symbol $x = \frac{1}{\sqrt{2}}(x_R + \imath\, x_I)$, where x_R and x_I are independent and uniformly distributed on $(0,1)$, i.e., x_R and x_I are positive fractions. Since $E[|x|^2] = 1/3$, it satisfies the average power constraint for the Gaussian network's channel inputs. Each input in the network is chosen independently and with the same distribution as x, and is denoted by $\{x_i^{(2)}\}$. The received signals are denoted by $\{y_j^{(2)}\}$. In Lemma 3 in Section 4.3, it is shown that the loss in the mutual information for this choice of inputs is $O(M)$. The mutual information of this channel is $\mathcal{I}_2 := I(x_\Omega^{(2)}; y_{\Omega^c}^{(2)}|x_{\Omega^c}^{(2)})$ and compares to the channel in Step 1 as

$$\mathcal{I}_1 - \mathcal{I}_2 = O(M). \tag{4.14}$$

Step 3: Quantization of the received signal

Next we quantize the received signal as follows:

- Retain only the integer portions of the real and imaginary parts, and discard the fractional portions, which we denote by $d_j^{(3)}$. The real and imaginary parts of $d_j^{(3)}$ have the same sign as the signal.

- Discard the signs of the real and imaginary parts, which we denote by $s_{jR}^{(3)}$ and $s_{jI}^{(3)}$, respectively.

We denote the quantized received signal by $y_j^{(3)}$. The mutual information across the channel in Step 2 can be rewritten as

$$\begin{aligned} \mathcal{I}_2 &= I(x_\Omega^{(2)}; y_{\Omega^c}^{(3)}, d_{\Omega^c}^{(3)}, s_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}) && (4.15) \\ &\leq I(x_\Omega^{(2)}; y_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}) + I(x_\Omega^{(2)}; d_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(3)}) + H(s_{\Omega^c}^{(3)}) && (4.16) \\ &\leq I(x_\Omega^{(3)}; y_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}) + I(x_\Omega^{(2)}; d_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(3)}) + 2|\Omega^c|. && (4.17) \end{aligned}$$

Now the fractional part $d_j^{(3)}$ is given by adding the fractional part of the signal with the fractional part of the noise.

$$\begin{aligned} I(x_\Omega^{(2)}; d_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(3)}) &= h(d_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(3)}) - h(d_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(3)}, x_\Omega^{(2)}) && (4.18) \\ &\leq \sum_{j \in \Omega^c} h(d_j^{(3)}) - h(\tilde{z}_j^{(3)}), && (4.19) \end{aligned}$$

where $\tilde{z}_j^{(3)}$ is the fractional part of the Gaussian noise. Since $d_j^{(3)}$ is fractional with $E[|d_j^{(3)}|^2] \leq 2$, its differential entropy is upper bounded by that of a Gaussian distribution with variance 2, and so $h(d_j^{(3)}) \leq \log(2\pi e)$. Since z_j is distributed as $\mathcal{CN}(0,1)$, $h(\tilde{z}_j^{(3)}) = O(1)$. Hence,

$$I(x_\Omega^{(2)}; d_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(3)}) \leq |\Omega^c|(\log(2\pi e) - O(1)) = O(M). \qquad (4.20)$$

So, defining $\mathcal{I}_3 := I(x_\Omega^{(2)}; y_{\Omega^c}^{(3)} | x_{\Omega^c}^{(2)})$, we have

$$\mathcal{I}_2 - \mathcal{I}_3 = O(M). \qquad (4.21)$$

Step 4: Further quantization of the received signal

Next we further quantize the received signal at the end of the previous step by restricting the binary expansions of its real and imaginary parts to n bits, and denote the result by $y_j^{(4)}$, where

$$n := \max_{(i,j)\in\mathcal{E}} \max\{\lfloor \log |h_{ijR}| \rfloor, \lfloor \log |h_{ijI}| \rfloor\}. \quad (4.22)$$

Note that we have previously quantized the real and imaginary parts to be positive integers. Denote the discarded part of the received signal by $d_j^{(4)}$. The mutual information of the channel in Step 3 can be rewritten as

$$\begin{aligned}
\mathcal{I}_3 &= I(x_\Omega^{(2)}; y_{\Omega^c}^{(4)}, d_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}) && (4.23)\\
&= I(x_\Omega^{(2)}; y_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}) + I(x_\Omega^{(2)}; d_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}, y_{\Omega^c}^{(4)}) && (4.24)\\
&\leq I(x_\Omega^{(2)}; y_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}) + I(x_\Omega^{(2)}; d_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}). && (4.25)
\end{aligned}$$

We bound $I(x_\Omega^{(2)}; d_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)})$ next. From the definition of n, $|h_{ij}| = \sqrt{(h_{ijR}^2 + h_{ijI}^2)} \leq 2^{n+1}$. Since $|x_i^{(2)}| \leq 1$, $|\sum_{i\in N(j)} h_{ij} x_i^{(2)}| \leq M2^{n+1}$. Hence, the binary expansion of the integer part of $|\sum_{i\in N(j)} h_{ij} x_i^{(2)}|$ has $(n + O(\log M))$ bits. Since $d_j^{(4)}$ is the portion of the received signal that exceeds n bits of representation, it is easy to see that at most $O(\log M)$ higher order bits in the binary representation of $\sum_{i\in N(j)} h_{ij} x_i^{(2)}$ influence $d_j^{(4)}$. Therefore, we have $I(x_\Omega^{(2)}; d_j^{(4)} | x_{\Omega^c}^{(2)}) = O(\log M)$ and subsequently $I(x_\Omega^{(2)}; d_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}) = O(M \log M)$. Now define $\mathcal{I}_4 := I(x_\Omega^{(2)}; y_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)})$ and we get

$$\mathcal{I}_3 - \mathcal{I}_4 = O(M \log M). \quad (4.26)$$

Step 5: Quantization of the transmit signals

Next, we restrict the real and imaginary parts of the scaled inputs also to n bits. Let the binary expansion of $\sqrt{2}\,x_{iR}^{(4)}$ be $0.x_{iR}(1)x_{iR}(2)\ldots$, so

$$x_{iR}^{(4)} = \frac{1}{\sqrt{2}}\sum_{p=1}^{\infty} 2^{-p} x_{iR}(p). \tag{4.27}$$

Similarly denote the binary expansion of $\sqrt{2}\,x_{iI}^{(4)}$. Define

$$\begin{aligned} x_{iR}^{'} &:= \frac{1}{\sqrt{2}}\sum_{p=1}^{n} x_{iR}(p) 2^{-p}, \\ x_{iI}^{'} &:= \frac{1}{\sqrt{2}}\sum_{p=1}^{n} x_{iI}(p) 2^{-p}. \end{aligned}$$

We will consider the new inputs $x_i^{'} := x_{iR}^{'} + \imath x_{iI}^{'}$ and let the corresponding received signals under these inputs be denoted by $y_j^{'}$. The mutual information here compares against that in Step 4 as

$$\begin{aligned} \mathcal{I}_4 &= I(x_\Omega^{(2)}, x_\Omega^{'}; y_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}, x_{\Omega^c}^{'}), \quad (\text{since } x_i^{'} \text{ is only a function of } x_i^{(2)}) && (4.28) \\ &\leq I(x_\Omega^{(2)}, x_\Omega^{'}; y_{\Omega^c}^{(4)}, y_{\Omega^c}^{'} | x_{\Omega^c}^{(2)}, x_{\Omega^c}^{'}) && (4.29) \\ &= I(x_\Omega^{(2)}, x_\Omega^{'}; y_{\Omega^c}^{'} | x_{\Omega^c}^{(2)}, x_{\Omega^c}^{'}) + I(x_\Omega^{(2)}, x_\Omega^{'}; y_{\Omega^c}^{(4)} | x_{\Omega^c}^{(2)}, x_{\Omega^c}^{'}, y_{\Omega^c}^{'}) && (4.30) \\ &= I(x_\Omega^{'}; y_{\Omega^c}^{'} | x_{\Omega^c}^{'}) + I(x_\Omega^{(2)}, x_\Omega^{'}; y_{\Omega^c}^{(4)} | x_{\Omega^c}^{(4)}, x_{\Omega^c}^{'}, y_{\Omega^c}^{'}) && (4.31) \\ &\leq I(x_\Omega^{'}; y_{\Omega^c}^{'} | x_{\Omega^c}^{'}) + H(y_{\Omega^c}^{(4)} | y_{\Omega^c}^{'}). && (4.32) \end{aligned}$$

In (4.31), the first mutual information term is obtained due the Markov chain $\{x_i^{(2)}\} \rightarrow \{x_i^{'}\} \rightarrow \{y_j^{'}\}$. Since

$$\begin{aligned} |h_{ij}(x_i^{(2)} - x_i^{'})| &= |h_{ij}||x_i^{(2)} - x_i^{'}| && (4.33) \\ &\leq (2^{n+1})(2^{-n}) \leq 2, && (4.34) \end{aligned}$$

we get

$$|\sum_{i\in N(j)} h_{ij}(x_i^{(2)} - x_i^{'})| \;=\; O(M). \tag{4.35}$$

Hence the binary representation of the integer part of $|\sum_{i\in N(j)} h_{ij}(x_i^{(2)} - x_i^{'})|$ will have $O(\log M)$ bits. This results in $H(|y_j^{(4)} - y_j^{'}|) \;=\; O(\log M)$, due to which $H(y_{\Omega^c}^{(4)}|y_{\Omega^c}^{'}) \;=\; O(M\log M)$ bits. We obtain

$$\mathcal{I}_4 - I(x_\Omega^{'}; y_{\Omega^c}^{'}|x_{\Omega^c}^{'}) \;=\; O(M\log M). \tag{4.36}$$

Now observe that $x_i^{'}$ and $y_j^{'}$ are the transmit and receive signals in the discrete network. Also, $\{x_i^{'}\}$ are i.i.d. and uniformly distributed on their alphabet. In Step 1, we started with the optimal jointly Gaussian inputs for the cut in the Gaussian network and concluded in this step with i.i.d. uniform inputs for the discrete network. The total loss in the mutual information in this transformation is $O(M\log M)$. Now define $\overline{CS}_D$ as

$$\overline{CS}_D \;:=\; \min_{\Omega\in\Lambda} \; I(x_\Omega^{'}; y_{\Omega^c}^{'}|x_{\Omega^c}^{'}). \tag{4.37}$$

$\overline{CS}_D$ evaluates the right hand side of the cut-set bound (4.1) for the discrete network for a specific choice of the input distribution. As a consequence of the above arguments, we have also proved that $CS_G - \overline{CS}_D \;=\; O(M\log M)$. □

The choice of i.i.d. uniform inputs for the discrete network will be useful in Section 5.1.3.

4.2.2 Cut-set bounds of Gaussian and superposition networks

Let CS_S denote the cut-set bound of the superposition network.

Lemma 2. *The cut-set bounds of the Gaussian network and the corresponding superposition* *network are within a constant gap of* $O(M \log M)$ *bits,*

$$|CS_G - CS_S| \;=\; O(M \log M), \tag{4.38}$$

with the gap independent of channel gains or SNR.

Proof. The proof of this lemma is similar to the proof of Lemma 1. First we prove that $\max\{CS_S - CS_G, 0\} \;=\; O(M \log M)$ and then we prove $\max\{CS_G - CS_D, 0\} \;=\; O(M \log M)$. The basic proof technique is borrowed from Lemma 1 where we compare the mutual information across a particular cut Ω in both models and bound the difference. Repeating this across all the cuts completes the proof of the lemma.

$\max\{CS_S - CS_G, 0\} \;=\; O(M \log M)$

Consider a cut Ω in the network and choose any distribution for the inputs $\{x_i\}$ in the superposition network. Since $\{x_i\}$'s satisfy an average power constraint, they are valid inputs for the Gaussian network. The received signal in the Gaussian network y_j can be rewritten as a function of the received signal $y_j^{''}$ in the superposition network, the channel gains before quantization h_{ij} and after quantization $\hat{h}_{ij}$, and the Gaussian noise z_j:

$$\begin{aligned} y_j &= y_j^{''} + \sum_{i \in \mathcal{N}(j)} \left((h_{ij} - \hat{h}_{ij})\, x_i + \text{ frac}(\widehat{\hat{h}_{ij}\; x_i}) \right) + z_j \\ &=: y_j^{''} + v_j + z_j. \end{aligned}$$

By definition $y_j^{''}$ is a discrete random variable lying in $\mathbb{Z} + \imath\mathbb{Z}$. Hence $y_j^{''}$ can be recovered from y_j, the integer parts of v_j and noise z_j, and the carry c_j obtained from adding the fractional parts of v_j and z_j. So,

$$I(x_\Omega; y''_{\Omega^c}|x_{\Omega^c}) \leq I(x_\Omega; y_{\Omega^c}, \hat{v}_{\Omega^c}, \hat{z}_{\Omega^c}, c_{\Omega^c}|x_{\Omega^c}) \quad (4.39)$$

$$\leq I(x_\Omega; y_{\Omega^c}|x_{\Omega^c}) + \sum_{j\in\Omega^c}(H(\hat{v}_j) + H(\hat{z}_j) + H(c_j)), \quad (4.40)$$

where $\hat{v}_j$ and $\hat{z}_i$ are the integer parts of the respective variables. Note that $\hat{v}_j$, $\hat{z}_j$, and c_j are actually complex numbers whose real and imaginary parts are integers. Now

$$|(h_{ij} - \hat{h}_{ij})\, x_i + \ \mathrm{frac}(\widehat{\hat{h}_{ij}\ x_i})| \ \leq \ O(1), \quad (4.41)$$

hence $|v_j| \ \leq \ O(M)$. Therefore, the integer parts of v_j can have at most $O(\log M)$ bits. We get $H(\hat{v}_j) \ \leq \ O(\log M)$. Since z_j is distributed as $\mathcal{CN}(0,1)$, $H(\hat{z}_j) \ = \ O(1)$. Since c_j is the carry, $H(c_j) \ = \ O(1)$. Substituting in (4.40), we obtain

$$I(x_\Omega; y''_{\Omega^c}|x_{\Omega^c}) - I(x_\Omega; y_{\Omega^c}|x_{\Omega^c}) \ = \ |\Omega^c|O(\log M) \quad (4.42)$$

$$= \ O(M \log M). \quad (4.43)$$

$\max\{CS_G - CS_S, 0\} \ = \ O(M \log M)$

We pick a particular cut Ω in the Gaussian network. Due to the similarities between the superposition model and the discrete model, we can directly reuse steps from Section 4.2.1 here. Steps $(1)-(2)$ from Section 4.2.1 are directly applicable here.

Step 3: Quantization of the received signal

Next we quantize the received signal by retaining only the integer portions of the real and imaginary parts and discarding the fractional portions. This step is similar to Step 3 in Section 4.2.1. The loss in the mutual information is $O(M)$ bits.

Step 4: Quantization of the transmit signal

This step is similar to Step 5 in Section 4.2.1 and the inputs are restricted to take values from the same finite alphabet. The quantization of the inputs leads to a loss in the mutual information of $O(M \log M)$ bits.

Step 5: Removal of the Gaussian noise

In this step, we remove the Gaussian noise at the receiver. The capacity of the channel can only increase if we remove the noise at the receiver, but we need to account for the carry bits in the received signal. Let the inputs and outputs at the end of Step 4 be given by $\{x_i^{(4)}\}$ and $\{y_j^{(4)}\}$, respectively, and let the mutual information across the cut in Step 4 be $\mathcal{I}_4 := I(I(x_\Omega^{(4)},;y_{\Omega^c}^{(4)}|x_{\Omega^c}^{(4)}))$. Denote the noiseless received signals in this step by $\{y_j^{(5)}\}$ where

$$y_j^{(5)} = \sum_{i \in \mathcal{N}(j)} \widehat{h_{ij} x_i^{(4)}}. \tag{4.44}$$

Let the carry bits that arise from the addition of the fractional parts of the noise and the signal be denoted by $\{c_j^{(5)}\}$. Note that each $c_j^{(5)}$ has two bits of carry corresponding to the real and imaginary parts. Also, $y_j^{(4)}$ and $y_j^{(5)}$ are discrete signals and lie in $\mathbb{Z} + \imath\mathbb{Z}$. Given $y_j^{(5)}$ and $c_j^{(5)}$, we can simulate the effect of the Gaussian noise at the receiver. Hence

$$\begin{aligned}
\mathcal{I}_4 &:= I(x_\Omega^{(4)},;y_{\Omega^c}^{(4)}|x_{\Omega^c}^{(4)}) && (4.45)\\
&\leq I(x_\Omega^{(4)},;y_{\Omega^c}^{(5)}, c_{\Omega^c}^{(5)}|x_{\Omega^c}^{(4)}) && (4.46)\\
&\leq I(x_\Omega^{(4)},;y_{\Omega^c}^{(5)}|x_{\Omega^c}^{(4)}) + H(c_{\Omega^c}^{(4)}) && (4.47)\\
&\leq I(x_\Omega^{(4)},;y_{\Omega^c}^{(5)}|x_{\Omega^c}^{(4)}) + 2|\Omega^c|. && (4.48)
\end{aligned}$$

Then if

$$\mathcal{I}_5 := I(x_\Omega^{(4)},;y_{\Omega^c}^{(5)}|x_{\Omega^c}^{(4)}), \tag{4.49}$$

we have

$$\mathcal{I}_4 - \mathcal{I}_5 = O(M). \tag{4.50}$$

Step 6: Quantization of channel gains

The channel gains are quantized by neglecting their fractional parts as mentioned in the description of the superposition model in Section 3.2.2. The analysis in this step is similar to Step 5 in Section 4.2.1, and the magnitude of the difference in the received signals before and after quantization of the channel gains is $O(M)$. Hence, the loss in the mutual information across the cut is $O(M \log M)$.

At the end of Step 6, we are left with the superposition model and the total loss in the mutual information is at most $O(M \log M)$ bits. This completes the proof of the lemma. □

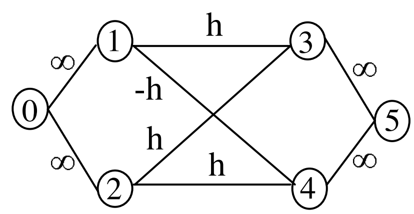
4.2.3 Cut-set bounds of Gaussian and linear deterministic networks

As mentioned in Section 4.1.1, one can compute the cut-set bound for the Gaussian network up to a bounded gap of $O(M)$ bits by choosing the i.i.d. Gaussian input distribution and, from Section 4.1.2, one can exactly evaluate the cut-set bound for the linear deterministic network. In this section, we establish the unboundedness of the difference in the cut-set bounds of the Gaussian network and the corresponding linear deterministic network. While comparing the cut-set bounds, we compare the mutual information across cuts in a Gaussian network with ranks of the corresponding cuts in the linear deterministic network.

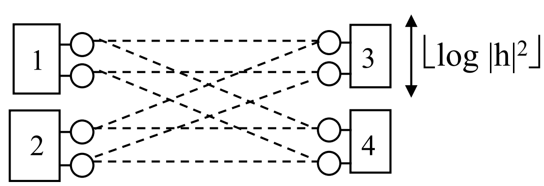
It is known that the cut-set bound is the capacity of the linear deterministic network and the cut-set bound is approximately achievable (up to $O(M \log M)$ bits) for Gaussian relay networks [7]. In fact, Theorem 6 (in Section 5.2.1) is another proof of the later result for layered Gaussian networks with a bounded gap of $O(M^2)$ bits. Hence the unboundedness result in this section implies that the linear deterministic model cannot approximate the capacity of Gaussian relay networks and subsequently the capacities of general Gaussian networks.

Inability of the linear deterministic model to capture phase

Consider the network in Figure 4.1(a) where the channels marked as ∞ have very high capacity. We can instead replace the corresponding channel gains with high values so that the cuts involving these channels do not limit the cut-set bound. So, for the network in Figure 4.1(a), the cut-set bound is determined by the mutual information across the cut $\Omega = \{0, 1, 2\}$.



(a) Gaussian network.



(b) Portion of the corresponding deterministic network.

Figure 4.1: Example of a network where the linear deterministic model of a Gaussian network does not capture the phase of the channel gain.

The mutual information across Ω is

$$\begin{aligned} I(x_\Omega; y_{\Omega^c} | x_{\Omega^c}) &= \log |I + \mathcal{H}\mathcal{H}^\dagger| \\ &= 2\log(1 + 2|h|^2) \\ &= 4\log|h| + O(1), \text{ as } |h| \to \infty, \end{aligned}$$

with

$$\mathcal{H} = \begin{bmatrix} h & -h \\ h & h \end{bmatrix}.$$

This is the minimum among all cuts and is therefore the cut-set bound of the network.

In the corresponding linear deterministic network in Figure 4.1(b), the transfer matrix of $\Omega = \{0, 1, 2\}$ is

$$\mathcal{G} = \begin{bmatrix} I & I \\ I & I \end{bmatrix}, \tag{4.51}$$

where each identity matrix has dimension $\lfloor \log |h|^2 \rfloor$. The cut-set bound of the network is the rank of $\mathcal{G}$, i.e., $\lfloor 2 \log |h| \rfloor$.

The gap between the cut-set bounds of the Gaussian network in Figure 4.1(a) and its linear deterministic counterpart is $2 \log |h| + O(1)$. Therefore the gap can grow unboundedly as a function of the channel gain h. The main reason for the unboundedness of the gap is that the linear deterministic model considers only the magnitude of a channel gain and fails to capture its phase. Constructing the deterministic model over a prime field larger than $\mathbb{F}_2$ does not circumvent this problem.

Inability of the linear deterministic model to capture power

Unfortunately, phase is not the only problem. We construct a Gaussian network with *positive* channel coefficients that still cannot be approximated by a linear deterministic network in the sense that their corresponding cut-set bounds can be very different.

Consider the Gaussian network in Figure 4.2, where $h = 2^k$ for $k \in \mathbb{Z}_+$. The linear deterministic model corresponding to this Gaussian network is the same as that in Figure 4.1(b). However, the difference in the cut-set bounds of two networks is unbounded.

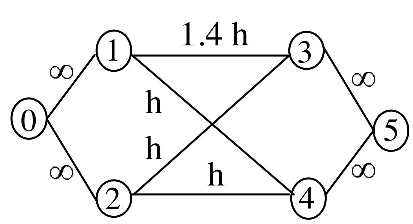


Figure 4.2: Example of a network where the linear deterministic model does not capture the power of the channel gain.

The cut-set bound of the network in Figure 4.2 is

$$\begin{aligned} CS_G &= I(x_\Omega; y_{\Omega^c} | x_{\Omega^c}) \\ &= 4 \log h + O(1) \quad (\text{as } h \to \infty), \end{aligned}$$

with $\Omega = \{0, 1, 2\}$.

Hence, once again, the difference in the cut-set bounds of the Gaussian network in Figure 4.2 and its linear deterministic counterpart will be $2 \log |h|$, which is unbounded in h. This example demonstrates the inability of the linear deterministic model to capture the received signal power. Intuitively, the linear deterministic model quantizes all channel gains to a power of 2, and hence fails to capture the received signal power in certain examples.

One may wonder if taking the channel gains into account and quantizing the gains with respect to a field larger than $\mathbb{F}_2$ will provide a bounded error approximation. However this reasoning is flawed since then the gap in the capacities would be a function of the chosen prime and thus, in turn, a function of channel gains.

4.3 Loss of capacity for uniform inputs

In Section 4.2.1, we stated that the loss in choosing uniform inputs for a cut, over Gaussian inputs, in a Gaussian network is bounded. We prove that statement in this section.

Lemma 3. Consider a cut Ω in a Gaussian network. Then the loss in the mutual information for choice of fractional inputs (as defined in Step 2 of Section 4.2.1) is $O(M)$.

Proof. Denote the i.i.d. Gaussian inputs by $\{x_i^G\}$. Also denote i.i.d Gaussian inputs distributed as $\mathcal{CN}(0, 1/3)$ by $\{\tilde{x}_i^G\}$. Now consider the fractional inputs defined in Section 4.2.1, denoted by $\{x_i^F\}$.

We observe some similarities between $\{\tilde{x}_i^G\}$ and $\{x_i^F\}$. Both the sets of inputs are i.i.d. and each input has the same variance in both cases. Hence, the covariance matrix of $\underline{\tilde{x}}^G = (\tilde{x}_i^G)$

or $\underline{x}^F = (x_i^F)$ is the scaled identity matrix $I_{M\times M}/3$. Denote the received signal in Ω^c under the Gaussian inputs and the fractional inputs by $\tilde{y}_{\Omega^c}^G$ and $y_{\Omega^c}^F$, respectively. It follows that the covariance matrices of the vectors $(\underline{\tilde{x}}^G, \tilde{y}_{\Omega^c}^G)$ and $(\underline{x}^F, y_{\Omega^c}^F)$ are the same.

Now consider the mutual information $I(x_\Omega^F; y_{\Omega^c}^F | x_{\Omega^c}^F)$,

$$\begin{aligned} I(x_\Omega^F; y_{\Omega^c}^F | x_{\Omega^c}^F) &= h(x_\Omega^F | x_{\Omega^c}^F) - h(x_\Omega^F | x_{\Omega^c}^F, y_{\Omega^c}^F) & (4.52)\\ &= \sum_{i\in\Omega} h(x_i^F) - h(\underline{x}_\Omega^F | \underline{x}_{\Omega^c}^F, \underline{y}_{\Omega^c}^F) & (4.53)\\ &= \sum_{i\in\Omega} (2\log(1/\sqrt{2})) - h(x_\Omega^F | x_{\Omega^c}^F, y_{\Omega^c}^F) & (4.54)\\ &= -|\Omega|\log 2 - h(x_\Omega^F | x_{\Omega^c}^F, y_{\Omega^c}^F). & (4.55) \end{aligned}$$

where (4.53) follows from the independence of $\{x_i\}$, and (4.54) follows from direct computation of the differential entropy. Now, for the circular Gaussian inputs

$$h(\tilde{x}_\Omega^G | \tilde{x}_{\Omega^c}^G) = \sum_{i\in\Omega} h(\tilde{x}_i^G) = \log(\pi e/3)|\Omega|. \quad (4.56)$$

Since, for a given covariance constraint, the conditional entropy is maximized by the Gaussian distribution with the same covariance [39],

$$h(x_\Omega^F | x_{\Omega^c}^F, y_{\Omega^c}^F) \leq h(\tilde{x}_\Omega^G | \tilde{x}_{\Omega^c}^G, \tilde{y}_{\Omega^c}^G). \quad (4.57)$$

Substituting (4.56) and (4.57) into (4.55),

$$\begin{aligned} I(x_\Omega^F; y_{\Omega^c}^F | x_{\Omega^c}^F) &\geq h(\tilde{x}_\Omega^G | \tilde{x}_{\Omega^c}^G) - h(\tilde{x}_\Omega^G | \tilde{x}_{\Omega^c}^G, y_{\Omega^c}^G) - |\Omega|(\log(\pi e/3) + \log 2) & (4.58)\\ &\geq I(\tilde{x}_\Omega^G; \tilde{y}_{\Omega^c}^G | \tilde{x}_{\Omega^c}^G) - O(M). & (4.59) \end{aligned}$$

Now the mutual information with i.i.d. $\mathcal{CN}(0,1)$ inputs $\{x_i^G\}$ is given by

$$I(x_\Omega^G; y_{\Omega^c}^G | x_{\Omega^c}^G) = \log |I + \mathcal{H}_\Omega \mathcal{H}_\Omega^\dagger|, \tag{4.60}$$

where $\mathcal{H}_\Omega$ is the channel transfer matrix across the cut Ω. Now consider the same cut, but increase the noise variance at all the received signals from 1 to 3. The mutual information reduces to $\log |I + \mathcal{H}_\Omega \mathcal{H}_\Omega^\dagger / 3|$. Since the effect of increasing the noise variance is the same as that of reducing the signal power to 1/3,

$$I(\tilde{x}_\Omega^G; \tilde{y}_{\Omega^c}^G | \tilde{x}_{\Omega^c}^G) = \log |I + \mathcal{H}_\Omega \mathcal{H}_\Omega^\dagger / 3|. \tag{4.61}$$

Comparing (4.60) and (4.61),

$$I(\tilde{x}_\Omega^G; \tilde{y}_{\Omega^c}^G | \tilde{x}_{\Omega^c}^G) \geq I(x_\Omega^G; y_{\Omega^c}^G | x_{\Omega^c}^G) - 3|\Omega|. \tag{4.62}$$

Hence

$$I(x_\Omega^F; y_{\Omega^c}^F | x_{\Omega^c}^F) \geq I(x_\Omega^G; y_{\Omega^c}^G | x_{\Omega^c}^G) - O(M). \tag{4.63}$$

In [7], it is shown that the loss in choosing the inputs to be i.i.d. Gaussian $\mathcal{CN}(0,1)$ instead of the joint Gaussian distribution is $O(M)$ bits. Therefore the choice of fractional inputs $\{x_i^F\}$ leads to a loss of at most $O(M)$ bits in the mutual information when compared to the joint Gaussian inputs. $\square$

CHAPTER 5

LINEAR NETWORK CODE FOR RELAY NETWORKS

In this chapter, we present a simple linear coding scheme for layered Gaussian relay networks that is approximately optimal. The linear network code is based on the quantization-based digital interface defined by the discrete model. Our key result consists of three dovetailed parts. The first part, proved in Section 4.2.1, is that there is no significant loss in the cut-set bound while modeling Gaussian networks with the discrete network, in that the bit gap between the cut-set bounds of the two networks is bounded over all SNRs. Therefore any coding strategy for the discrete network that achieves rates close to the cut-set bound can be easily converted to a nearly capacity-achieving strategy for the Gaussian relay network by simply quantizing the signals. This motivates and sets the stage for the problem of determining nearly capacity-achieving coding strategies for the discrete network, which is the topic of this chapter. The second result, described in Section 5.1, builds on this first key result. What we determine is that simple linear network coding on the bits of the en bloc finite representations of the received signals will suffice for the layered discrete network. The third part is that this coding strategy for the discrete network can be easily implemented on the Gaussian network, through simple quantization, and achieves rates within a bounded bit gap from the capacity. Thereby we obtain both a capacity-approximating discrete network, a simple linear coding strategy for the discrete network, as well as a natural mapping of the coding strategy from the layered discrete network to the layered Gaussian network. Together, the combined results establish a discrete network for both analyzing the original network, as well as operating it near-optimally through simple quantization. It is therefore appropriate to call such a discrete network with these two rigorously established properties a *quantization-based digital interface.*

5.1 Linear network code for layered discrete relay networks

The linear network code presented in the sequel achieves all rates within $O(M^2)$ bits of the cut-set bound of the layered discrete relay network, with the constant gap independent of channel gains or SNR, and only a function of the number of nodes in the network.

5.1.1 Coding scheme for relay networks

Any coding scheme for a relay network requires specification of the source's codewords, the functions applied each time to the received signals by the relay nodes, and the destination's decoding function.

A $(2^{NR}, N)$ code for a relay network is an encoding function for the source

$$\underline{x}_0 : \{1, 2, \ldots, 2^{NR}\} \to \mathcal{X}^N,$$

where $\mathcal{X}$ is the input alphabet of the channel, and a set of encoding functions for relay node k,

$$g_{k,m} : \mathcal{Y}^{m-1} \to \mathcal{X}, \text{ for } m = 1, 2, \ldots, N, \; k = 1, 2, \ldots, M-1,$$

where $\mathcal{Y}$ is the alphabet of node k's received signal. For simplicity of exposition, we assume that the input alphabets of all the relay nodes are the same, and that so are their output alphabets. The symbols transmitted by a relay can depend on all the symbols previously received by the relay. The decoding function of the destination M is given by

$$g_M : \mathcal{Y}^N \to \{1, 2, \ldots, 2^{NR}\}.$$

Let $\mathcal{W}$ be a random variable uniformly distributed on $\{1, 2, \ldots, 2^{NR}\}$ that corresponds to the message that source 0 wants to communicate. Such a $\mathcal{W}$ is mapped to the codeword $\underline{x}_0(\mathcal{W})$. The average probability of error is given by

$$P_e = Pr(g_M(\underline{y}_M) \neq \mathcal{W}).$$

The capacity of the relay network is the supremum of all rates R such that for any $\epsilon > 0$, there exists a block length N and a coding strategy for which $P_e < \epsilon$.

5.1.2 Layered networks

In the coding scheme described above, the encoding functions $\{g_{k,m}\}$ at relay nodes operate on a symbol-by-symbol basis and can vary with time. We restrict our attention to layered discrete networks, for which the encoding functions $\{g_{k,m}\}$ at the relay nodes can be simplified. In a layered network [7], as the name suggests, nodes are divided into groups or layers. The nodes of one layer can only transmit to nodes of the subsequent layer. The source node 0 is the sole node in the zeroeth layer, and the last layer, layer L, contains only the destination M. All other relays are divided among the intermediate layers. The nodes in layer k are denoted by $\mathcal{L}_k$. An example with four layers of nodes is shown in Figure 5.1 with $\mathcal{L}_0 = \{0\}, \mathcal{L}_1 = \{1, 2\}, \mathcal{L}_2 = \{3, 4\}$, $\mathcal{L}_3 = \{5\}$, and $L = 3$.

In a layered network we can restrict attention to the following simplified block-by-block coding scheme, where each block consists of N symbols. Consider a $(2^{NR}, N)$ code for the relay network, and suppose that the source transmits a codeword of length N, which we call a "block." Now let all the relays in $\mathcal{L}_1$ buffer their received signals for N time units. Subsequently each relay generates N-length transmit vectors, i.e., blocks, as a function of their received vectors. This is possible since only the source transmits to the relays in $\mathcal{L}_1$, and these relays generate their transmit symbols causally as a function of their previous receptions. Similarly, each node in $\mathcal{L}_k$ waits for the nodes in $\mathcal{L}_{k-1}$ to complete their transmissions, buffers the N received signals, and then transmits a block of length N. Finally, the

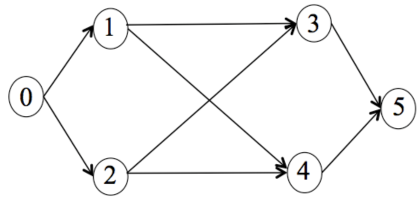


Figure 5.1: Example of a layered network.

destination receives the block of N symbols and attempts to decode the source's transmission. There is no loss in the capacity in operating on a block-by-block basis. The source can start the transmission of the next codeword once it completes the transmission of the first codeword. Also the relays in $\mathcal{L}_1$ are full-duplex, they transmit and receive at the same time. So the relays can continue receiving the next block of symbols while they transmit a block of symbols. The destination will continue to receive blocks of symbols one after another. In this way, we achieve a constant rate R of communication in the network.

5.1.3 Linear network codes for the discrete relay networks

The linear network code we propose is constructed on the discrete network. As described in Section 3.2.1, the transmit and receive signals in the discrete network are quantized to lie in a finite set of complex numbers consisting of real and complex parts in [0,1], each with only n bits of precision.

Concerning transmissions, the transmit signal $x_i = x_{i,R} + \imath x_{i,I}$ has real and imaginary parts given by

$$x_{i,R} = \sum_{k=1}^{n} 2^{-k} x_{i,R}(k), \tag{5.1}$$

$$x_{i,I} = \sum_{k=1}^{n} 2^{-k} x_{i,I}(k). \tag{5.2}$$

with each $x_{i,R}(i)$ and $x_{i,I}(j)$ in $\mathbb{F}_2$, and where

$$n \;:=\; \max_{(i,j)\,\in\,\mathcal{E}} \; \max\{\lfloor \log|h_{ijR}| \rfloor, \lfloor \log|h_{ijI}| \rfloor\}. \tag{5.3}$$

The received signals in the discrete network $\{y_j^{'}\}$ are obtained by quantizing the received signals in the Gaussian network. The Gaussian received signal is quantized by discarding the signs of the real and imaginary parts, further discarding the fractional portions of the real and imaginary parts and retaining only the integer portion, and finally quantizing the integer portion of the real and imaginary parts by truncating their binary expansions to n bits of precision. If $y_j \;=\; y_{j,R} + \imath y_{j,I}$ is the Gaussian received signal, and we denote the binary expansions of the integer parts of $|y_{j,R}|$ and $|y_{j,I}|$ by $\sum_{k=1}^{\infty} 2^k y_{j,R}(k)$ and $\sum_{k=1}^{\infty} 2^k y_{j,I}(k)$, respectively, then the received signal $y_j^{'}$ in the discrete network is

$$y_j^{'} \;=\; [y_j] \;:=\; \left(\sum_{k=1}^{n} 2^k y_{j,R}(k)\right) + \imath \left(\sum_{k=1}^{n} 2^k y_{j,I}(k)\right). \tag{5.4}$$

Hence we regard the received signal in the discrete network as a binary tuple

$$(y_{j,R}(1), y_{j,R}(2), \ldots, y_{j,R}(n), y_{j,I}(1), y_{j,I}(2), \ldots, y_{j,I}(n))$$

of length $2n$ where each entry is 0 or 1. Similarly, the transmitted signal in the discrete network can also be so regarded. In the rest of this dissertation, we reserve the phrase "binary $2n$-tuple" or "$2n$-tuple" to describe such a vector, which can be converted in a straightforward fashion to a complex symbol for transmission, or can be obtained in a straightforward fashion from a complex symbol that is received.

After N such symbols have been received, there is a $2nN$-length binary vector that is the received block, and similarly there is a $2nN$-length binary vector that is the transmitted block. We will reserve the phrase "binary $2nN$-vector" or "$2nN$-vector" to refer to such a vector which represents a block of N received symbols, either for block transmission or after block reception. We will represent a $2nN$-length received or transmitted binary vector at a node j using an underbar, as in $\underline{y}_j^{'}$ or $\underline{x}_j$, respectively.

We will employ a linear coding scheme where the transmitted block is simply obtained by multiplying the received block by a $2nN \times 2nN$ matrix of 0s and 1s.

The overall coding scheme is randomly generated, and is simple to describe.

Source's codewords

There are 2^{NR} messages. Using a uniform distribution on binary $2n$-tuples, the source randomly generates a set of 2^{NR} codewords, each of length $2nN$, where each codeword is constructed by independently picking N binary $2n$-tuples from the uniform distribution. Note that each codeword corresponds to N complex symbols, which are transmitted over N discrete time instants. The choices of the rate R and the block-length N are elaborated later in the proof of Lemma 4 in Section 5.1.4. The source transmits the codeword corresponding to the particular message that has been chosen for communication.

Relay's linear mappings

Relay j randomly chooses a $2nN \times 2nN$ binary matrix A_j, by independently picking each entry as either 0 or 1 with equal probability. This will be the matrix representing the linear code at node j. The relay buffers N received binary $2n$-tuples, and adjoins them to construct a binary $2nN$-vector that constitutes the received block $\underline{y}_j^{'}$. It then multiplies this binary $2nN$-vector by A_j to obtain a binary $2nN$-vector $\underline{x}_j$ that constitutes the transmit block. It splits this binary $2nN$-vector into N binary $2n$-tuples. Converting each of the binary $2n$-tuples back into complex numbers, with the real and imaginary parts, each in $[0, 1]$, and each of n-bit precision, gives N complex symbols. These N complex transmit symbols are transmitted by the relay node over N discrete-time instants. An example of the encoding operation at node j is shown in Figure 5.2.

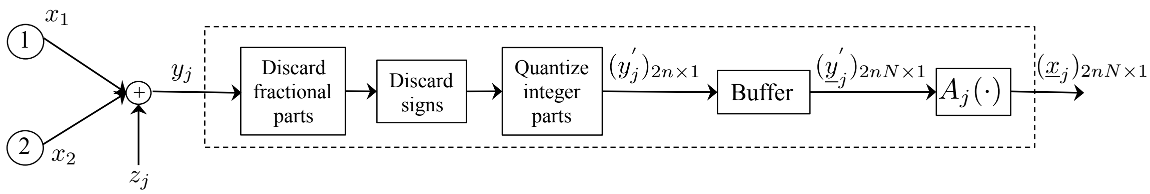


Figure 5.2: Linear encoding at node j.

All the $M-1$ relays independently construct their binary encoding matrices in a similar manner as described above.

Next, to define decoding, we define strong typicality of vectors:

Definition 1. A vector $\underline{x} \in \mathcal{X}^N$ is defined to be ϵ-*strongly typical* with respect to a distribution $p(x)$, denoted by $\underline{x} \in \mathcal{T}_{\epsilon,p}$, if

$$|\nu_x(\underline{x}) - p(x)| \;\leq\; \epsilon\, p(x),\;\forall\, x \in \mathcal{X}, \tag{5.5}$$

where $\epsilon \in \mathcal{R}^+$ and $\nu_x(\underline{x}) = \frac{1}{N}|n : x_n = x|$ is the empirical frequency.

This is extended in the standard way to include joint strong typicality of vectors (see [38]). Vectors $\underline{x} = (x_1, x_2, \ldots, x_N) \in \mathcal{X}^N$ and $\underline{y} = (y_1, y_2, \ldots, y_N) \in \mathcal{Y}^N$ are ϵ-jointly strongly typical with respect to the distribution $p(x, y)$, if $\underline{z} = (z_1, z_2, \ldots, z_N)$, where each $z_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$ is jointly typical with respect to the distribution $p(x, y)$. We employ the following definitions from [40], slightly modifying them to include the linear operations.

Define a singleton set $\chi_0(w) := \{\underline{x}_0(w)\}$ for every message w.

Now consider a node j in $\mathcal{L}_1$, the first layer. Define the *set of received vectors at node j associated with a message w* as

$$\mathcal{Y}_j^{'}(w) \;:=\; \{\underline{y}_j^{'} : (\underline{y}_j^{'}, \underline{x}_0(w)) \in \mathcal{T}_{\epsilon,p}\}, \tag{5.6}$$

with p denoting the distribution $p(\tilde{x}_0, y_j^{'})$ where $\tilde{x}_0$ is uniformly distributed and $p(\tilde{y}_j^{'}|\tilde{x}_0)$

models the channel from node 0 to node j in the discrete network.

Continuing to consider the node $j \in \mathcal{L}_1$, we define the *set of transmit vectors at node j that are associated with the message w* as

$$\chi_j(w) = \{\underline{x}_j : \underline{x}_j \;=\; A_j(\underline{y}_j^{'}), \text{ where } \underline{y}_j^{'} \in \mathcal{Y}_j^{'}(w)\}. \tag{5.7}$$

Next consider a node $j \in \mathcal{L}_k$ with $k \geq 2$. Noting that any $i \in \mathcal{N}(j)$ belongs to layer $\mathcal{L}_{k-1}$, we recursively (in k) define its set of received vectors at node j associated with message w as

$$\mathcal{Y}_j^{'}(w) \;:=\; \{\underline{y}_j^{'} : (\underline{y}_j^{'}, \{\underline{x}_i\}_{i \in \mathcal{N}(j)}) \in \mathcal{T}_{\epsilon,p}, \text{ for some } \underline{x}_i \in \chi_i(w), \text{ for each } i \in \mathcal{N}(j)\}. \tag{5.8}$$

The distribution p above is $p(\{\tilde{x}_i\}_{i \in \mathcal{N}_j}, \tilde{y}_j^{'})$, where $\{\tilde{x}_i\}_{i \in \mathcal{N}(j)}$ are independent and uniformly distributed, and $p(\tilde{y}_j^{'} | \{\tilde{x}_i\}_{i \in \mathcal{N}(j)})$ models the channel from the nodes in $\mathcal{N}(j)$ to j in the discrete network. We also define the set of transmitted vectors at node j associated with the message w as

$$\chi_j(w) := \{\underline{x}_j : \underline{x}_j \;=\; A_j(\underline{y}_j^{'}), \text{ for some } \underline{y}_j^{'} \in \mathcal{Y}_j^{'}(w)\}. \tag{5.9}$$

Definition 2. We write $(\underline{y}_j^{'}, w) \in \mathcal{T}^{\epsilon}$ if $\underline{y}_j^{'} \in \mathcal{Y}_j^{'}(w)$.

Decoding at the destination

The destination receives the $2nN$-vector $\underline{y}_M^{'}$, and decodes by searching for a message w such that $(\underline{y}_M^{'}, w) \in \mathcal{T}^{\epsilon}$. If it finds more than one such message or if it finds none, it declares an error. Else it declares the unique choice as its estimate of the source's message.

It should be noted that though the relay's encoding is a linear operation over the block of $2nN$-vectors, the end-to-end channel from the binary $2nN$-vectors transmitted by the source to the block of binary $2nN$-vectors received by the destination is not linear over the binary field, and we cannot describe decoding as just inverting a binary matrix $2nN \times 2nN$ matrix. This is because of the truncation operations that are an integral part of the very definition of the discrete network.

5.1.4 Computing the rate achieved by the linear network code

Next we compute the rate achievable by the linear network code. In (4.37), we defined $\overline{CS}_D$ as the cut-set bound's value for the discrete network for a specific choice of the input distribution.

Lemma 4. *The linear network code achieves all rates within a gap of $O(M^2)$ bits of $\overline{CS}_D$ for layered networks, where the $O(M^2)$ gap is independent of channel gains or SNR.*

Proof.

Probability of error: The probability of error for the coding scheme (see Section 5.1.1 for the notation) is given by

$$P_e = \Pr(g_M(\underline{y}_M^{'}) \neq \mathcal{W}) \tag{5.10}$$

$$= \frac{1}{2^{NR}} \sum_{w=1}^{2^{NR}} \Pr(g_M(\underline{y}_M^{'}) \neq w | \mathcal{W} = w). \tag{5.11}$$

As is standard, since P_e is symmetric in the transmitted message, we assume wlog that $\mathcal{W} = 1$ and evaluate the probability of error when the first codeword is transmitted, $P_e|_{\mathcal{W}=1}$.

Error events: The possible error events at the destination M are

- E_0: One of the transmitted or received vectors in the network is not strongly typical. That is $\underline{y}_j^{'} \notin \mathcal{T}_{\epsilon,p}$, where p denotes the distribution induced on $y_j^{'}$ by the uniform distribution on each x_i for $i \in \mathcal{N}(j)$, and by the conditional probability $p(y_j^{'}|\{x_i\}_{i\in\mathcal{N}(j)})$ describing the channel in the discrete network from the nodes in $\mathcal{N}(j)$ to $y_j^{'}$.

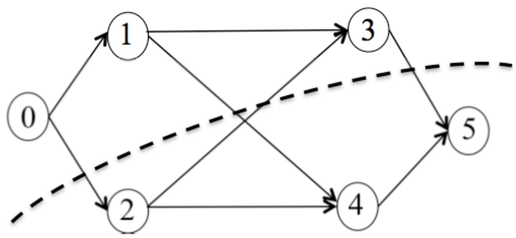


Figure 5.3: Example of a cut Ω in the network in Figure 5.1. Here $\Omega = \{0, 1, 3\}$ and $\Omega^c = \{2, 4, 5\}$.

- E_1: $(\underline{y}'_M, 1) \notin \mathcal{T}^\epsilon$.
- E_w: $(\underline{y}'_M, w) \in \mathcal{T}^\epsilon$, where $w \neq 1$.

By applying the union bound,

$$P_e|_{\mathcal{W}=1} \leq \Pr(E_0) + \Pr(E_1 \wedge E_0^c) + \Pr(\bigvee_{w \neq 1} E_w \wedge E_0^c \wedge E_1^c). \quad (5.12)$$

From Lemma 9 (see Section 5.3), for any $\epsilon > 0$,

$$\Pr(E_0) + \Pr(E_1) \leq \epsilon, \text{ for } N \text{ sufficiently large.} \quad (5.13)$$

Therefore,

$$P_e|_{\mathcal{W}=1} \leq \epsilon + \sum_{w=2}^{2^{NR}} \Pr(\tilde{E}_w), \quad (5.14)$$

where $\tilde{E}_w := E_w \wedge E_0^c \wedge E_1^c$.

Error event $\tilde{E}_w$: Let $\mathcal{L}_k$ be the set of nodes in layer k. We say that a node $j \in \mathcal{L}_k$ is *confused* [1] by w if $(\underline{y}'_j, w) \in \mathcal{T}^\epsilon$ for some $w \neq 1$. The destination is confused by w under $\tilde{E}_w$. The source is not confused by definition. Hence, under the error event $\tilde{E}_w$, the nodes in the network get separated into two sets, ones that are confused by w, and others that are not confused by w. This is a cut in the network; see Figure 5.3 for an example.

[1] This is similar to the notion of *distinguishability* used in [8] and [7].

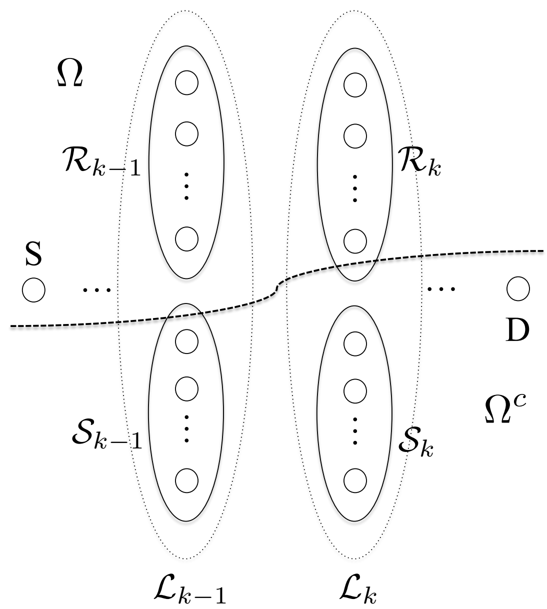


Figure 5.4: Layers $(k-1)$ and k in a network. The wireless links connecting the nodes are not shown. Each oval represents a particular set. The dashed curve in the center divides the network into Ω and Ω^c.

Fix an arbitrary cut Ω, and define the sets (see Figure 5.4)

$$F_{k,\Omega} := \mathcal{L}_k \cap \Omega \text{ and } G_{k,\Omega} := \mathcal{L}_k \cap \Omega^c. \tag{5.15}$$

Let us denote the concatenation of the received $2nN$-vectors of nodes at layer ℓ by $\underline{y}'_{\mathcal{L}_\ell}$. Or, in greater detail, if $i_1, i_2, \ldots, i_q$ are the nodes, in lexicographic order, that are present in layer ℓ, then $\underline{y}'_{\mathcal{L}_\ell} := (\underline{y}'_{i_1}, \underline{y}'_{i_2}, \ldots, \underline{y}'_{i_q})$. Correspondingly, we also define the concatenation of transmitted vectors at layer ℓ under $\underline{y}'_{\mathcal{L}_\ell}$ by $\underline{x}_{\mathcal{L}_\ell}(\underline{y}'_{\mathcal{L}_\ell}) := (A_{i_1}(\underline{y}'_{i_1}), A_{i_2}(\underline{y}'_{i_2}), \ldots, A_{i_q}(\underline{y}'_{i_q}))$. Thereby we define the concatenated received vectors $\underline{y}'_{\mathcal{L}_1}, \underline{y}'_{\mathcal{L}_2}, \ldots, \underline{y}'_{\mathcal{L}_L}$ at the L layers in the network, and also the corresponding concatenated transmitted vectors. Next we define the *network-wide received vector* $\underline{y}'_{\mathcal{V}} := (\underline{y}'_{\mathcal{L}_1}, \underline{y}'_{\mathcal{L}_2}, \ldots, \underline{y}'_{\mathcal{L}_L})$, with receptions ordered according to layers, and lexicographically within layers. The corresponding *network-wide transmitted vector* is $\underline{x}_{\mathcal{V}}(\underline{y}'_{\mathcal{V}}) := (\underline{x}_{\mathcal{L}_1}(\underline{y}'_{\mathcal{L}_1}), \underline{x}_{\mathcal{L}_2}(\underline{y}'_{\mathcal{L}_2}), \ldots, \underline{x}_{\mathcal{L}_L}(\underline{y}'_{\mathcal{L}_L}))$.

We now wish to define the subset of network-wide received vectors that are consistent in the sense of joint typicality with a source message 1. Similar to the definition of $\mathcal{Y}_j^{'}(w)$ in (5.6) and (5.8), we define *received vectors at a node j that are associated with message w,* except that we now do it at a layer and across the whole network. We can do this by exploiting the layered nature of the network where nodes at one layer only transmit to the nodes in the next layer.

We say *a network-wide received vector* $\underline{y}_{\mathcal{V}} = (\underline{y}^{'}_{\mathcal{L}_1}, \underline{y}^{'}_{\mathcal{L}_2}, \ldots, \underline{y}^{'}_{\mathcal{L}_L})$ *is associated with message w* if for each ℓ and each node $j \in \mathcal{L}_\ell$, $(\underline{y}^{'}_j, \underline{x}_{\mathcal{L}_{\ell-1}}(\underline{y}^{'}_{\mathcal{L}_{\ell-1}})) \in \mathcal{T}_{\epsilon,p}$, where p is the distribution $p(\tilde{x}_{\mathcal{L}_{\ell-1}}, \tilde{y}^{'}_j)$, where $\tilde{x}_{\mathcal{L}_{\ell-1}}$ is uniformly distributed, and $p(\tilde{y}^{'}_j|\tilde{x}_{\mathcal{L}_{\ell-1}})$ is the conditional distribution describing the channel from transmissions by nodes at layer $\ell - 1$ to node j. We denote by $\mathcal{Y}_{\mathcal{V}}(w)$ the set of such network-wide received vectors associated with message w, and denote by $\chi_{\mathcal{V}}(w)$ the set of network-wide transmitted vectors. If the source transmits message 1, then with high probability, for a sufficiently large N, by the asymptotic equipartition property (AEP) of strongly typical vectors [38], the random network-wide vectors $\underline{y}_{\mathcal{V}}$ will lie in $\mathcal{Y}_{\mathcal{V}}(w)$.

We note that the codewords corresponding to the messages are random, as are the matrices $\{A_j\}$, due to the random coding strategy. As a consequence, $\mathcal{Y}_{\mathcal{V}}$ are also random, since they are dependent on the codewords and network coding matrices. Now we proceed to conduct an analysis under the random coding strategy.

Randomly choose a codeword for each message, and a set matrices $\{A_j : 1 \leq j \leq M - 1\}$, and randomly, with a uniform distribution, pick a network-wide received vector $\underline{\tilde{y}}^{'}_{\mathcal{V}}$ from the set $\mathcal{Y}_{\mathcal{V}}(1)$. For the random coding strategy, define $\mathcal{G}^w_{k,\Omega}$ as the event that every node in $G_{k,\Omega}$ is confused by w, i.e., for a node $j \in G_{k,\Omega}$,

$$(\{\underline{x}_i(w)\}_{i\in\mathcal{L}_{k-1}}, \underline{\tilde{y}}^{'}_j) \in \mathcal{T}_{\epsilon,p}, \quad \text{for some } \underline{x}_i(w) \in \chi_i(w), \text{ for all } i \in \mathcal{L}_{k-1}, \tag{5.16}$$

with p denoting the distribution $p(\{x_i\}_{i\in\mathcal{N}(j)}, \tilde{y}^{'}_j)$, where $\{x_i\}_{i\in\mathcal{N}(j)}$ are independent and uniformly distributed and $p(\tilde{y}_j^{'}|\{x_i\}_{i\in\mathcal{N}(j)})$ models the channel in the discrete network. Define $\mathcal{F}_{k,\Omega}^{w}$ as the event that no node in $F_{k,\Omega}$ is confused by w. Note that, in (5.16), due to the random coding strategy, $\chi_i(w)$ is also random.

For a random choice of $\underline{\tilde{y}}_{\mathcal{V}}^{'}$, let $\mathcal{P}_{1,w,\Omega}$ denote the probability that no node in Ω is confused by w when 1 is the transmitted message, and all the nodes in Ω^c are confused by w. Then,

$$\begin{aligned}
\mathcal{P}_{1,w,\Omega} &= \Pr(\bigwedge_{k\geq 0} \mathcal{F}_{k,\Omega}^{w} \wedge \mathcal{G}_{k,\Omega}^{w}) && (5.17)\\
&= \prod_{k\geq 1} \Pr(\mathcal{G}_{k,\Omega}^{w}| \bigwedge_{l=0}^{k-1} \mathcal{F}_{l,\Omega}^{w} \wedge \mathcal{G}_{l,\Omega}^{w}) \Pr(\mathcal{F}_{k,\Omega}^{w}|\mathcal{G}_{k,\Omega}^{w} \wedge \bigwedge_{l=0}^{k-1} \mathcal{F}_{l,\Omega}^{w} \wedge \mathcal{G}_{l,\Omega}^{w}) && (5.18)\\
&\leq \prod_{k\geq 1} \Pr(\mathcal{G}_{k,\Omega}^{w}| \bigwedge_{l=0}^{k-1} \mathcal{F}_{l,\Omega}^{w} \wedge \mathcal{G}_{l,\Omega}^{w}). && (5.19)
\end{aligned}$$

Now we make two important observations concerning each relay node j's buffering of N of its received symbols and multiplication of such a buffered $2nN$-vector by the matrix A_j. First, since the matrix A_j is independently chosen for each relay, all the relay mappings are independent of each other.

Second, when we randomize over all the linear encodings A_j at relay node j, every transmit vector $\underline{x}_j \in \chi_j(w)$, for any w, is independently and uniformly distributed over the set of binary vectors of length $2nN$. Hence each symbol in a transmit vector $\underline{x}_j$ is independently and uniformly distributed.

Error event at layer k: Next, define $\mathcal{P}_{1,w,\Omega}^{k}$ as the contribution of the k-th layer to the probability, i.e.,

$$\mathcal{P}_{1,w,\Omega}^{k} := \Pr(\mathcal{G}_{k,\Omega}^{w}| \bigwedge_{l=0}^{k-1} \mathcal{F}_{l,\Omega}^{w} \wedge \mathcal{G}_{l,\Omega}^{w}). \qquad (5.20)$$

Consider the transmit vectors of node $i \in \mathcal{L}_{k-1}$ given by $\underline{\tilde{x}}_i = A_i\, \underline{\tilde{y}}_i^{'}$. Under the conditioning events to determine $\mathcal{P}_{1,w,\Omega}^{k}$, we know that $\underline{\tilde{x}}_i \in \chi_i(w)$ for $i \in G_{k-1,\Omega}$, and $\underline{\tilde{x}}_i \in \chi_i(1)$ for $i \in F_{k-1,\Omega}$. Denote the transmit vector of node $i \in G_{k-1,\Omega}$ by $\underline{\tilde{x}}_i(w)$ to indicate that it is the transmit block of a confused node, and denote the transmit vector of node $i \in F_{k-1,\Omega}$

by $\underline{\tilde{x}}_i(1)$, since it is not confused by any other message. Now, by our earlier random choice of $\underline{y}_{\mathcal{V}}$ from $\mathcal{Y}_{\mathcal{V}}(1)$, we know that the received vector $\underline{y}_j^{'}$ at node $j \in G_{k,\Omega}$, and $\underline{\tilde{x}}_{G_{k-1,\Omega}}(w) = \{\underline{\tilde{x}}_i(w)\}_{i \in G_{k-1,\Omega}}$ are jointly typical. Hence, in order to compute $\mathcal{P}_{1,w,\Omega}^k$, we need to compute the probability that for every node $j \in G_{k,\Omega}$,

$$(\{\underline{x}_i(w)\}_{i \in \mathcal{L}_{k-1}}, \underline{\tilde{y}}_j) \in \mathcal{T}_{\epsilon,p}, \quad \text{for some } \underline{x}_i(w) \in \chi_i(w), \text{ for all } i \in \mathcal{L}_{k-1}, \tag{5.21}$$

where p is the distribution $p(\tilde{x}_{\mathcal{L}_{k-1}}, \tilde{y}_j^{'})$, where $\{x_i\}_{i \in \mathcal{L}_{k-1}}$ is uniformly distributed, and $p(\tilde{y}_j^{'} | \tilde{x}_{\mathcal{L}_{k-1}})$ is the conditional distribution describing the channel from transmissions by nodes at layer $k-1$ to node j, given that

- $\underline{\tilde{x}}_{F_{k-1,\Omega}}(1)$ was transmitted by the nodes in $F_{k-1,\Omega}$,
- $\underline{\tilde{x}}_{G_{k-1,\Omega}}(w)$ was transmitted by the nodes in $G_{k-1,\Omega}$, and
- $(\{\underline{\tilde{x}}_i(w)\}_{i \in G_{k-1,\Omega}}, \underline{\tilde{y}}_j) \in \mathcal{T}_{\epsilon,p}$, where p is the distribution $p(\{x_i\}_{i \in G_{k-1,\Omega}}, \tilde{y}_j^{'})$.

Project $\chi_{\mathcal{V}}(w)$, the set of network-wide transmit vectors, onto the nodes in $\mathcal{L}_{k-1}$ to obtain the set $\chi_{\mathcal{L}_{k-1}}(w)$. Pick $\mathcal{L}_{k-1}$-wide transmit vectors $\underline{x}_{\mathcal{L}_{k-1}} = \{\underline{x}_i(w)\}_{i \in \mathcal{L}_{k-1}}$ from $\chi_{\mathcal{L}_{k-1}}(w)$. If node j checks to see if this choice of $x_{\mathcal{L}_{k-1}}$ will result in $(x_{\mathcal{L}_{k-1}}, \underline{\tilde{y}}_j^{'})$ lying in $\mathcal{T}_{\epsilon,p}$, and $\underline{x}_i(w) = \underline{\tilde{x}}_i(w)$, for all $i \in G_{k-1,\Omega}$, then, under the conditions listed above, it will find a positive answer with probability less than

$$2^{-NI(x_{F_{k-1,\Omega}}; y_j^{'}, x_{G_{k-1,\Omega}})} \quad = \quad 2^{-NI(x_{F_{k-1,\Omega}}; y_j^{'} | x_{G_{k-1,\Omega}})}, \tag{5.22}$$

where the uniform distribution on $\{x_i\}$ is used to evaluate the mutual information and (5.22) is due to the independence of transmit symbols $\{x_i\}$. The choice of the i.i.d. uniform distribution for $\{x_i\}$ follows from the discussion following (5.19). Instead, for the set $\underline{x}_{\mathcal{L}_{k-1}}$, if $\underline{x}_i(w) \neq \underline{\tilde{x}}_i(w)$ for all $i \in G_{k-1,\Omega}$, then the probability of (5.21) is

$$2^{-NI(x_{F_{k-1,\Omega}}, x_{G_{k-1,\Omega}}; y_j^{'},)} \quad \leq \quad 2^{-NI(x_{F_{k-1,\Omega}}; y_j^{'} | x_{G_{k-1,\Omega}})}. \tag{5.23}$$

For all other choices of $\{\underline{x}_i(w)\}_{i \in \mathcal{L}_{k-1}} \in \chi_{\mathcal{L}_{k-1}}(w)$, when if $\underline{x}_i(w) \neq \tilde{\underline{x}}_i(w)$ for node i in a subset of $G_{k-1,\Omega}$, the probability of (5.21) is similarly upper bounded by

$$2^{-NI(x_{F_{k-1,\Omega}}; y_j^{'} | x_{G_{k-1,\Omega}})}. \tag{5.24}$$

From Lemma 10 (see Section 5.4), $|\chi_{\mathcal{L}_{k-1}}(w)| = 2^{O(M)N}$. So, we apply the union bound with respect to all the vectors in $\chi_{\mathcal{L}_{k-1}}(w)$ to get

$$\begin{aligned} & \Pr(\text{node } j \text{ is confused by } w | \bigwedge_{l=0}^{k-1} \mathcal{F}_{l,\Omega}^{w} \wedge \mathcal{G}_{l,\Omega}^{w}) \\ \leq \;& 2^{O(M)N} \; 2^{-NI(x_{F_{k-1,\Omega}}; y_j^{'} | x_{G_{k-1,\Omega}})}. \end{aligned} \tag{5.25}$$

We bound $\mathcal{P}_{1,w,\Omega}^{k}$ as

$$\mathcal{P}_{1,w,\Omega}^{k} = \Pr(\text{nodes in } G_{k,\Omega} \text{ are confused by } w | \bigwedge_{l=0}^{k-1} \mathcal{F}_{l-1,\Omega}^{w} \wedge \mathcal{G}_{l-1,\Omega}^{w}) \tag{5.26}$$

$$\leq \prod_{j \in G_{k,\Omega}} \Pr(\text{node } j \text{ is confused by } w | \bigwedge_{l=0}^{k-1} \mathcal{F}_{l-1,\Omega}^{w} \wedge \mathcal{G}_{l-1,\Omega}^{w}) \tag{5.27}$$

$$\leq 2^{O(M)|G_{k,\Omega}|N} \, 2^{-N \sum_{j \in G_{k,\Omega}} I(x_{F_{k-1,\Omega}}; y_j^{'} | x_{G_{k-1,\Omega}})} \tag{5.28}$$

$$\leq 2^{O(M)|G_{k,\Omega}|N} \, 2^{-NI(x_{F_{k-1,\Omega}}; y_{G_{k,\Omega}}^{'} | x_{G_{k-1,\Omega}})}. \tag{5.29}$$

To obtain the bound in (5.27), we note that under the conditioning events, the nodes in $\mathcal{L}_{k-1}$ transmit $\underline{x}_{\mathcal{L}_{k-1}} = \{\underline{x}_i\}_{i \in \mathcal{L}_{k-1}}$. Given a set of transmissions, $\underline{x}_{\mathcal{L}_{k-1}}$, the received vectors $\underline{y}_{j_1}^{'}$ and $\underline{y}_{j_2}^{'}$ at node j_1 and j_2, respectively, in layer k are conditionally independent. We obtain the bound by noting that the conditioning events in (5.27) involve many such mutually exclusive transmissions by the nodes in $\mathcal{L}_{k-1}$. The bound in (5.28) is from (5.25). With a lexicographic ordering of the nodes in $\mathcal{L}_k$ as $\{j_1, j_2, \ldots, j_{|G_{k,\Omega}|}\}$, the bound in (5.29) is obtained as

$$\begin{aligned}
&\sum_{l=1}^{|G_{k,\Omega}|} I(x_{F_{k-1,\Omega}}; y^{'}_{j_l}|x_{G_{k-1,\Omega}}) \\
&= \sum_{l=1}^{|G_{k,\Omega}|} H(y^{'}_{j_l}|x_{G_{k-1,\Omega}}) - H(y^{'}_{j_l}|x_{\mathcal{L}_{k-1}}) && (5.30)\\
&\geq \sum_{l=1}^{|G_{k,\Omega}|} H(y^{'}_{j_l}|x_{G_{k-1,\Omega}}, \cup_{m=1}^{l-1}\underline{y}^{'}_{j_m}) - H(y^{'}_{j_l}|x_{\mathcal{L}_{k-1}}, \cup_{m=1}^{l-1}\underline{y}^{'}_{j_m}) && (5.31)\\
&= I(x_{F_{k-1,\Omega}}; y^{'}_{G_{k,\Omega}}|x_{G_{k-1,\Omega}}), && (5.32)
\end{aligned}$$

where (5.31) follows from the Markov chain $(\cup_{m=1}^{l-1}\underline{y}^{'}_{j_l}, x_{\mathcal{L}_{k-1}}) \to x_{\mathcal{L}_{k-1}} \to y^{'}_{j_l}$, and that conditioning reduces entropy.

Back substitutions: Substituting from (5.29) in (5.19), we get

$$\begin{aligned}
\mathcal{P}_{1,w,\Omega} &\leq (2^{O(M)\sum_{k\geq 1}|G_{k,\Omega}|N})\,(2^{-N\sum_{k\geq 1} I(x_{F_{k-1,\Omega}}; y^{'}_{G_{k,\Omega}}|x_{G_{k-1,\Omega}})}) && (5.33)\\
&\leq 2^{O(M)|\Omega^c|N}\, 2^{-NI(x_\Omega; y^{'}_{\Omega^c}|x_{\Omega^c})} && (5.34)\\
&= 2^{O(M^2)N}\, 2^{-NI(x_\Omega; y^{'}_{\Omega^c}|x_{\Omega^c})}. && (5.35)
\end{aligned}$$

The bound in (5.34) follows from the steps below and the chain rule for mutual information.

$$\begin{aligned}
I(x_{F_{k-1,\Omega}}; y^{'}_{G_{k,\Omega}}|x_{G_{k-1,\Omega}}) &= H(y^{'}_{G_{k,\Omega}}|x_{G_{k-1,\Omega}}) - H(y^{'}_{G_{k,\Omega}}|x_{\mathcal{L}_{k-1}}) && (5.36)\\
&= H(y^{'}_{G_{k,\Omega}}|x_{G_{k-1,\Omega}}) - H(y^{'}_{G_{k,\Omega}}|x_{\mathcal{V}}, \cup_{\ell=1}^{k-1} y^{'}_{G_{\ell,\Omega}}) && (5.37)\\
&\geq H(y^{'}_{G_{k,\Omega}}|x_{\Omega^c}, \cup_{\ell=1}^{k-1} y^{'}_{G_{\ell,\Omega}}) - H(y^{'}_{G_{k,\Omega}}|x_{\mathcal{V}}, \cup_{\ell=1}^{k-1} y^{'}_{G_{\ell,\Omega}}) && (5.38)\\
&\geq I(x_\Omega; y^{'}_{G_{k,\Omega}}|x_{\Omega^c}, \cup_{\ell=1}^{k-1} y^{'}_{G_{\ell,\Omega}}), && (5.39)
\end{aligned}$$

where the (5.37) uses the Markov structure of the layered network,

$$(x_{\mathcal{V}\setminus\mathcal{L}_{k-1}}, \cup_{\ell=1}^{k-1} y^{'}_{G_{\ell,\Omega}}) \to x_{\mathcal{L}_{k-1}} \to y^{'}_{G_{k,\Omega}}.$$

The probability of the event $\tilde{E}_w$ is bounded as

$$\begin{aligned}
\Pr(\tilde{E}_w) &\leq \sum_{\Omega\in\Lambda} \mathcal{P}_{1,w,\Omega} && (5.40)\\
&\leq \sum_{\Omega\in\Lambda} 2^{O(M^2)N}\, 2^{-NI(x_\Omega;\, y'_{\Omega^c}|x_{\Omega^c})} && (5.41)\\
&\leq 2^{M+1}\, 2^{O(M^2)N}\, 2^{-N\min_{\Omega\in\Lambda} I(x_\Omega;\, y'_{\Omega^c}|x_{\Omega^c})}. && (5.42)
\end{aligned}$$

Finally, substituting the bound for $\Pr(\tilde{E}_w)$ from (5.42) in (5.14) gives us

$$\begin{aligned}
P_e|_{\mathcal{W}=1} &\leq \epsilon + (2^{NR}-1)2^{M+1}\, 2^{O(M^2)N}\, 2^{-N\min_{\Omega\in\Lambda} I(x_\Omega;\, y'_{\Omega^c}|x_{\Omega^c})} && (5.43)\\
&\leq \epsilon + 2^{M+1} 2^{N(R+O(M^2)-\min_{\Omega\in\Lambda} I(x_\Omega;\, y'_{\Omega^c}|x_{\Omega^c}))}. && (5.44)
\end{aligned}$$

Hence, if $R < \min_{\Omega\in\Lambda} I(x_\Omega;\, y'_{\Omega^c}|x_{\Omega^c}) - O(M^2)$, then $P_e|_{\mathcal{W}=1}$ or P_e can be made arbitrarily small for a sufficiently large N. □

5.1.5 Approximate optimality of the linear network code for layered discrete networks

Next we prove that the linear network code is approximately optimal for the layered network.

Theorem 5. *The linear network coding scheme achieves the capacity of the layered discrete network up to a bounded number of bits, i.e., the rate R achieved by the coding scheme is bounded from the capacity C_D of the layered discrete relay network by $O(M^2)$ bits,*

$$C_D - R = O(M^2), \qquad (5.45)$$

where the constant gap is independent of channel gains or SNR.

Proof. In Lemma 1, we proved that the cut-set bound of the Gaussian network and its discrete counterpart are within a bounded gap of $O(M \log M)$ bits. In fact, in Section 4.2.1

we also showed that $\overline{CS}_D = CS_G - O(M \log M)$, where $\overline{CS}_D$ is the value of the cut-set bound of the discrete network for the specific choice of the uniform i.i.d. input distribution (see (4.37)) and CS_G is the cut-set bound of the Gaussian network. In Section 4.2.1, we proved that $CS_G \geq CS_D$, where CS_D is the cut-set bound of the discrete network. Therefore, $\overline{CS}_D = CS_D - O(M \log M)$. Since $CS_D \geq \overline{CS}_D$, we have

$$|CS_D - \overline{CS}_D| \; = \; O(M \log M). \tag{5.46}$$

From Lemma 4, we know that the linear network code achieves a rate R within $O(M^2)$ bits of $\overline{CS}_D$. Hence, along with the bound in (5.46), the linear network code achieves rates within $O(M^2)$ bits of the cut-set bound CS_D. Since $CS_D \geq C_D$, the theorem is proved. □

5.2 Linear network code for layered Gaussian networks

In the earlier sections we proved that the cut-set bounds of the Gaussian and the discrete relay networks are within a bounded gap of $O(M \log M)$ bits. Later we developed a simple linear coding scheme for the layered discrete network that is approximately optimal. Also, the discrete model is a digital interface for operating the Gaussian networks; the signals in the discrete model are obtained by quantizing the signals in the Gaussian model. Combining these results we obtain a nearly capacity-achieving coding strategy for layered Gaussian relay networks, which consists of (i) quantizing received signals, and (ii) collecting a block of such signals and applying linear network coding on the overall vector of bits.

In this section, we prove the optimality of the linear network code for Gaussian relay networks and later extend this to MIMO Gaussian relay networks, where the nodes can have multiple transmit and receive antennas, and to multicast networks, where the source can transmit the same information to a subset of the nodes.

5.2.1 Approximate optimality of the linear code for layered Gaussian relay networks

Theorem 6. *The linear network coding scheme achieves the capacity of the layered Gaussian relay network up to a bounded number of bits, i.e., the rate R achieved by the linear network code is bounded from the capacity C_G of the layered Gaussian relay network by $O(M^2)$ bits,*

$$C_G - R \;=\; O(M^2), \tag{5.47}$$

where the constant gap is independent of channel gains or SNR.

Proof. The proof of this theorem simply involves operating the Gaussian network on the digital interface described by the discrete model. The transmit signals are restricted to the finite alphabet of the discrete model, and the received signals in the Gaussian network are quantized to obtain the received signals for the discrete model; see Section 3.2.1 for details. Then the linear network code is constructed over the layered discrete relay network as described in Section 5.1.

From Theorem 5, we know that the linear network code achieves all rates within $O(M^2)$ bits of the cut-set bound of the layered discrete relay network CS_D. In Lemma 1, we proved that $|CS_G - CS_D| \;=\; O(M \log M)$. Combining these results, and noting that $CS_G \geq C_G$, we get the statement of the theorem. □

The theorem establishes that the discrete model can serve as a digital interface for Gaussian networks, since a coding scheme for the layered discrete relay network involving simple linear operations is approximately optimal and can be used on the layered Gaussian relay network. Since the linear network code achieves rates within a bounded bit gap of CS_G, the theorem is also a proof of the near-optimality of the cut-set bound for layered Gaussian relay networks, though this was proved earlier in [7] for a smaller bounded gap of $O(M \log M)$ bits.

5.2.2 MIMO relay networks

We can extend the linear network coding scheme to layered MIMO networks, where nodes have multiple transmit and receive antennas. In (3.2), we defined the received signal in a MIMO receiver in a Gaussian relay network. We operate the MIMO Gaussian relay network on the discrete interface as described in Section 3.2.1. The linear network code is defined on the MIMO discrete relay network. The basic ideas in the coding scheme remain the same, but with some modifications to accommodate multiple antennas. The details are:

Source's codewords

The source constructs a set of 2^{NR} codewords of length N. Every codeword is a $T_0 \times N$ matrix, where each row of the matrix is transmitted by one of the transmit antennas of the source. Each entry in the matrix is a complex number from the QAM constellation (or equivalently, a $2n_{MIMO}$-length binary tuple, see Section 3.2.1) that is independently chosen with the uniform distribution.

Relay's mappings

Relay j has U_j receive antennas, receives a vector of U_j symbols every instant, and buffers N such vectors. The received binary tuples are adjoined to construct a $2n_{MIMO}NU_j$-length binary vector. The relay constructs a $2n_{MIMO}NT_j \times 2n_{MIMO}NU_j$ binary matrix $A_{j,MIMO}$ where each entry of $A_{j,MIMO}$ is either 0 or 1. It multiplies the $2n_{MIMO}NU_j$-length binary vector with $A_{j,MIMO}$ to obtain a $2n_{MIMO}NT_j$-length binary vector. It splits this vector into T_j vectors of length $2n_{MIMO}N$. Each tuple of length $2n_{MIMO}N$ corresponds to N transmit symbols for a particular transmit antenna.

Decoding at the destination

Destination collects N received signals and finds a message that is associated with the received vector; see the decoding procedure in Section 5.1.3 for details. If it finds more than one message that satisfies this condition or if it finds none, it declares an error else it declares the unique transmitted message.

Let the maximum number of transmit or receive antennas in the network be $T_{\max}$.

Theorem 7. *The linear network coding scheme achieves the capacity of the layered MIMO Gaussian relay network up to a bounded number of bits, i.e., the linear code achieves all rates R bounded from the capacity $C_{G,MIMO}$ of the layered Gaussian relay network as*

$$C_{G,MIMO} - R \ \geq \ O(M^2 T_{\max}). \tag{5.48}$$

The constant in the bounded gap is independent of channel gains or SNR.

Proof. The steps in the proof are essentially the same as that of Theorem 6:

1. Operate the MIMO Gaussian relay network on the digital interface defined by the discrete model.

2. First we prove that the cut-set bound on the capacity of the MIMO Gaussian relay network and the MIMO discrete network are within a bounded gap of $O(MT_{\max} \log MT_{\max})$ bits. Here, the proof of Lemma 4 can be reused by a simple observation. While evaluating the cut-set bound, we can view each transmit and receive antenna as a virtual node. Hence the total number of nodes in the network is at most $MT_{\max}$ which gives us the necessary bound.

3. The coding scheme achieves all rates within $O(M^2 T_{\max})$ bits of the cut-set bound of the discrete MIMO network evaluated for a specific choice of the input distribution.

Here the input distribution is i.i.d. across all the transmit antennas in the network and is uniform over the alphabet. The arguments in the proof of Lemma 4 carry over to MIMO networks by noting that due to multiple transmit and receive antennas, the receive and transmit signals are complex vectors instead of scalars. In the proof, the value of $|\chi_i(w)|$ for MIMO networks will be different than before. For MIMO networks, the bound on $H(\underline{y}'_{\mathcal{L}_2}|\underline{x}_{\mathcal{L}_1})$ in (5.63) increases to $T_{\max}|\mathcal{L}_1|\,O(N)$. Due to this, the bound on $|\Psi(w)|$ is $2^{O(MT_{\max})N}$. With this, we get the required bounded gap of $O(M^2T_{\max})$ from the cut-set bound.

4. Then it is straightforward to prove near-optimality of the linear code for MIMO Gaussian networks; see Theorem 6.

□

5.2.3 Multicast networks

In a multicast network, the source node wants to communicate the same information to a subset of the nodes (instead of a single destination as in the previous sections). The remaining nodes which are not the intended recipients act as relays. Let $\mathcal{D}$ be the set of nodes that are the intended recipients of the source's message. Then the cut-set bound on the capacity $C_{G,Mult}$ for such networks is given by

$$C_{G,Mult} \leq \max_{p(x_0,x_1,\ldots,x_{M-1})} \min_{D\in\mathcal{D}} \min_{\Omega\,\in\,\Lambda_D} I(x_\Omega; y_{\Omega^c}|x_{\Omega^c}), \tag{5.49}$$

where Λ_D is the set of all cuts in the network that separate the source from the destination D. We can extend the linear network code from Section 5.1.3 to this class of networks, with the only difference being that *all* the intended destinations in $\mathcal{D}$ decode the source's transmission. In a layered multicast network, the destinations in $\mathcal{D}$ can be spread across the various layers in the network.

Theorem 8. *The linear network coding scheme achieves the capacity of the multicast layered Gaussian network up to a bounded number of bits, i.e., the linear network code achieves any rate R which is bounded from the capacity $C_{G,Mult}$ of the multicast layered Gaussian network as*

$$C_{G,Mult} - R \;=\; O(M^2). \tag{5.50}$$

The constant in the bounded gap is independent of channel gains or SNR.

Proof. The proof of this theorem resembles that of Theorem 6. We give a general outline of the proof and skip the details.

1. Operate the multicast Gaussian network on the digital interface defined by the discrete model.

2. First we prove that the cut-set bounds on the capacity of the multicast Gaussian network and the multicast discrete network are within a bounded gap of $O(M \log M)$ bits. Though the cut-set bound for multicast networks is slightly different from that of relay networks, the proof of Lemma 4 can be reused by individually comparing the cut-set bound between the source and each destination in $\mathcal{D}$.

3. Then we prove that the coding scheme achieves all rates within $O(M^2)$ bits of the cut-set bound of the discrete network, when the cut-set is evaluated for i.i.d. uniform inputs. We re-use the proof of Lemma 4, with the only difference being that multiple destinations want to decode the source's transmission instead of a single destination.

4. With the above arguments, we can prove the equivalent of Theorem 5 for layered multicast networks. Then, with the same arguments as in the proof of Theorem 6, we can prove the near-optimality of the linear code up to $O(M^2)$ bits for the original layered multicast Gaussian network.

□

5.3 Probability of the error event E_1

Lemma 9. The probability of the events E_0 or E_1: $(\underline{y}_M, 1) \notin \mathcal{T}$, for any $\epsilon > 0$, is bounded by

$$\Pr(E_0) + \Pr(E_1) \ \leq \ \epsilon, \text{ for } N \text{ sufficiently large.} \tag{5.51}$$

Proof. The proof of this lemma involves repeated application of the AEP for strongly typical vectors (see [38]).

Suppose the source transmits the codeword $\underline{x}_0(1)$. From the AEP of strongly typical vectors, the received signal $\underline{y}_j^{'}$ for a node in layer 1 is jointly typical with $\underline{x}_0(1)$, with probability exceeding $1-\epsilon_1$, for a sufficiently large N, for any positive ϵ_1. Hence, with high probability, $(\underline{y}_j^{'}, w) \in \mathcal{T}$ for $j \in \mathcal{L}_1$. Node i in $\mathcal{L}_1$ subsequently transmits a message vector $\underline{x}_i = A_i \underline{y}_i^{'}$, with $\underline{x}_i \in \chi_i(1)$. Then, the received vector $\underline{y}_j^{'}$ at a node in $\mathcal{L}_2$ satisfies (by AEP of strongly typical vectors)

$$(\underline{y}_j^{'}, \{\underline{x}_i\}_{i \in \mathcal{N}(j)}) \in \mathcal{T}_{\epsilon,p}, \tag{5.52}$$

with probability exceeding $1-\epsilon_2$, for a sufficiently large N, for any positive ϵ_2. In (5.52), the joint typicality of the vectors in $\mathcal{T}_\epsilon$ is with respect to the joint distribution $p(\{x_i\}_{i\in\mathcal{N}(j)}, y_j^{'})$, where $\{x_i\}_{i\in\mathcal{N}(j)}$ are independent and uniformly distributed, and $p(y_j^{'}|\{x_i\}_{i\in\mathcal{N}(j)})$ models the channel in the discrete network. Hence, with high probability, $(\underline{y}_j^{'}, 1) \in \mathcal{T}$ for $j \in \mathcal{L}_2$. Subsequently, each node in $\mathcal{L}_2$ transmits a message vector from $\chi_j(1)$.

We carry out this analysis across all the layers in the network, and obtain that the received signal in $\mathcal{L}_k$ satisfies $(\underline{y}_j^{'}, 1) \in \mathcal{T}$ with probability exceeding $1-\epsilon_k$, for a sufficiently large N, for any $\epsilon_k > 0$. The error event E_1 occurs if the received vector at the destination is not associated with message 1, and this will not occur (with high probability) if all the received vectors at the intermediate relay nodes are associated with 1. Hence, we get

$$\Pr(E_1) \ \leq \ \sum_k \epsilon_k, \text{ for a sufficiently large } N, \tag{5.53}$$

where the summation is over the number of layers.

Similarly to the above arguments, we can prove that the transmit and received message vectors in the network are strongly typical, with probability exceeding $1-\epsilon_0$, for a sufficiently large N, for any $\epsilon_0 > 0$. Hence

$$\Pr(E_0) \ \leq \ \epsilon_0. \tag{5.54}$$

Since $\{\epsilon_k\}$ are arbitrary positive numbers, for any $\epsilon > 0$,

$$\Pr(E_0) + \Pr(E_1) \ \leq \ \epsilon, \text{ for a sufficiently large } N. \tag{5.55}$$

□

5.4 Bounding the size of the typical set

Lemma 10. The size of the set $|\chi_{\mathcal{L}_{k-1}}(w)|$ (with the notations as given in the proof of Lemma 4) is $2^{O(M)\,N}$, where the size is independent of channel gains or SNR.

Proof. Consider the set $\mathcal{Y}_{\mathcal{V}}(w)$ of network-wide received vectors associated with the message w, defined in the proof of Lemma 4. For every $\underline{y}_{\mathcal{V}} \in \mathcal{Y}_{\mathcal{V}}(w)$, we can define a network-wide transmitted vector, associated with the message w, as

$$\underline{x}_{\mathcal{V}}(\underline{y}^{'}_{\mathcal{V}}) := (\underline{x}_{\mathcal{L}_1}(\underline{y}^{'}_{\mathcal{L}_1}), \underline{x}_{\mathcal{L}_2}(\underline{y}^{'}_{\mathcal{L}_2}), \ldots, \underline{x}_{\mathcal{L}_L}(\underline{y}^{'}_{\mathcal{L}_L})). \tag{5.56}$$

Define $\chi_{\mathcal{V}}(w)$ to consist of all such network-wide transmitted vectors associated with message w.

Consider the relay nodes in $\mathcal{L}_1$ and let us bound the conditional entropy $H((y_i^{'})_{i\in\mathcal{L}_1}|x_0)$:

$$
\begin{aligned}
H(y_{\mathcal{L}_1}^{'}|x_0) &\leq \sum_{i\in\mathcal{L}_1} H(y_i^{'}|\underline{x}_0) && (5.57)\\
&= \sum_{i\in\mathcal{L}_1} H([h_{1i}x_0 + z_i]|x_0) && (5.58)\\
&\leq \sum_{i\in\mathcal{L}_1} H(z_i^{'}, c_i), && (5.59)
\end{aligned}
$$

where $z_{jn}^{'}$ is the integer part of the Gaussian noise[2] and c_{jn} is the carry from adding the fractional parts of the signal and the noise. Since z_j is distributed as $\mathcal{CN}(0,1)$, $H(z_j^{'})$ is independent of channel gains or SNR and $H(\underline{z}_j^{'}) = O(1)$. Also $H(c_j) = O(1)$, hence $H(y_{\mathcal{L}_1}^{'}|x_0) = |\mathcal{L}_1|O(1)$.

Let $\mathcal{Y}_{\mathcal{L}_1}(w)$ be the projection of $\mathcal{Y}_{\mathcal{V}}(w)$ onto the nodes in $\mathcal{L}_1$. For a sufficiently large N, by the AEP of strongly typical vectors, the size of $\mathcal{Y}_{\mathcal{L}_1}(w)$ is

$$
|\mathcal{Y}_{\mathcal{L}_1}(w)| = 2^{NH(y_{\mathcal{L}_1}^{'}|x_0)} = 2^{N|\mathcal{L}_1|O(1)}. \qquad (5.60)
$$

Since every set of transmit vectors in $\chi_{\mathcal{L}_1}(w)$ is associated with a received vector in $\mathcal{Y}_{\mathcal{L}_1}(w)$, the size of $\chi_{\mathcal{L}_1}(w)$ is also bounded by

$$
|\chi_{\mathcal{L}_1}(w)| \leq 2^{N|\mathcal{L}_1|O(1)}. \qquad (5.61)
$$

Now, similarly to (5.57)–(5.59), we can bound

$$
\begin{aligned}
H(y_{\mathcal{L}_2}^{'}|x_{\mathcal{L}_1}) &\leq \sum_{j\in\mathcal{L}_2} H(y_j^{'}|x_{\mathcal{L}_1}) && (5.62)\\
&= |\mathcal{L}_2|O(1). && (5.63)
\end{aligned}
$$

Let us fix a set of transmit vectors $\underline{x}_{\mathcal{L}_1} \in \chi_{\mathcal{L}_1}(w)$; then consider a set of received vectors $\underline{y}_{\mathcal{L}_2}^{'}$

[2] z_{jn} is a complex number lying in $\mathbb{Z} + \imath\mathbb{Z}$ corresponding to the integer portion of the real and imaginary parts of the complex noise z_j.

for the nodes in $\mathcal{L}_2$ such that

$$(\underline{y}_j^{'}, \{\underline{x}_i\}_{i\in\mathcal{N}(j)}) \in \mathcal{T}_{\epsilon,p}, \text{ for all } j \in \mathcal{L}_2. \tag{5.64}$$

The distribution p is $p(\{\tilde{x}_i\}_{i\in\mathcal{N}_j}, \tilde{y}_j^{'})$, where $\{\tilde{x}_i\}_{i\in\mathcal{N}(j)}$ are independent and uniformly distributed, and $p(\tilde{y}_j^{'}|\{\tilde{x}_i\}_{i\in\mathcal{N}(j)})$ models the channel from the nodes in $\mathcal{N}(j)$ to j in the discrete network. For a fixed $\underline{x}_{\mathcal{L}_1}$, the size of the set of received vectors $y_{\mathcal{L}_2}^{'}$ that satisfy the above is given by (for a sufficiently large N, by the joint AEP of strongly typical sequences)

$$2^{NH(y_{\mathcal{L}_2}^{'}|x_{\mathcal{L}_1})} \;\leq\; 2^{N\mathcal{L}_2 O(1)}. \tag{5.65}$$

Define $\mathcal{Y}_{\mathcal{L}_1,\mathcal{L}_2}(w)$ as the projection of $\mathcal{Y}_{\mathcal{V}}(w)$ onto the nodes in $\mathcal{L}_1$ and $\mathcal{L}_2$. Now every set of vectors in $\mathcal{Y}_{\mathcal{L}_1,\mathcal{L}_2}(w)$ can be obtained by first fixing a set of received vectors $\underline{y}_{\mathcal{L}_1}^{'}$ in $\mathcal{Y}_{\mathcal{L}_1}(w)$, determining the transmit vectors generated from it as $\underline{x}_{\mathcal{L}_1}(\underline{y}_{\mathcal{L}_1}^{'})$, and finding a set of received vectors $\underline{y}_{\mathcal{L}_2}^{'}$ that are jointly typical with this set $\underline{x}_{\mathcal{L}_1}(\underline{y}_{\mathcal{L}_1}^{'})$, as in (5.64). By counting the number of vectors across the layers, the size of $\mathcal{Y}_{\mathcal{L}_1,\mathcal{L}_2}(w)$ is bounded by

$$|\mathcal{Y}_{\mathcal{L}_1,\mathcal{L}_2}(w)| \;\leq\; (2^{N\mathcal{L}_1 O(1)})(2^{N\mathcal{L}_2 O(1)}) \;=\; 2^{N(\mathcal{L}_1+\mathcal{L}_2)O(1)}. \tag{5.66}$$

With the bound, obtained similarly to (5.59),

$$H(y_{\mathcal{L}_k}^{'}|x_{\mathcal{L}_{k-1}}) \;\leq\; |\mathcal{L}_k|O(1), \tag{5.67}$$

we extend this argument across all the layers in the network. The size of $\mathcal{Y}_{\mathcal{V}}(w)$ can be bounded as

$$|\mathcal{Y}_{\mathcal{V}}(w)| \;\leq\; 2^{N\sum_{\ell=1}^{L}|\mathcal{L}_\ell|O(1)} \;=\; 2^{NO(M)}. \tag{5.68}$$

Since each network-wide transmit vector in $\chi_{\mathcal{V}}(w)$ is a function of a vector in $\mathcal{Y}_{\mathcal{V}}(w)$,

$$|\chi_{\mathcal{V}}(w)| \;=\; 2^{NO(M)}. \tag{5.69}$$

Since $\chi_{\mathcal{L}_{k-1}}(w)$ is the projection of the set $\chi_{\mathcal{V}}(w)$ onto the nodes in $\mathcal{L}_{k-1}$, the size of $\chi_{\mathcal{L}_{k-1}}(w)$ is bounded by the size of $\chi_{\mathcal{V}}(w)$. Hence, the lemma is proved. □

CHAPTER 6

LIFTING CODING SCHEMES FOR THE SUPERPOSITION NETWORK TO THE GAUSSIAN NETWORK

In Chapter 4, we proved that the cut-set bounds of the Gaussian and the superposition networks are within a bounded gap of $O(M \log M)$ bits. In this chapter, we show that the superposition network can be a surrogate for Gaussian networks. We describe a simple procedure to lift any coding scheme for the superposition network to the Gaussian network by pruning the set of codewords. This lifting procedure reduces the rate of the code by at most $O(M \log M)$ bits. Later in this chapter, the capacities of the Gaussian relay network and the superposition relay network are shown to be within a bounded gap of $O(M \log M)$ bits. Both these constant gaps are independent of channel gains or SNR. Since the superposition network well-approximates the capacity of the Gaussian network and also allows for lifting of coding strategies to the Gaussian network, near-optimal codes for the Gaussian relay network can be constructed by first constructing near-optimal codes for the superposition network and later lifting them to the Gaussian network. Hence the superposition network also serves as a digital interface for operating Gaussian networks.

6.1 Lifting codes from the superposition model

A coding strategy for either the Gaussian or superposition relay network specifies codewords transmitted by the source, a mapping from the received signal to a transmit signal for every relay node, and a decoding function for the destination (see Section 5.1.1). For the sake of simplicity of exposition, we assume that the graph describing the network is acyclic and that the relays employ time-invariant encoding functions. Later, in Section 6.1.2, we mention how to handle more general encoding functions.

We describe how to lift a coding strategy for the superposition network to a strategy for the Gaussian network. Consider a $(2^{NR}, N)$ code for the superposition network with zero probability of error, for a certain N. The probability of error can be reduced to zero due to the deterministic nature of the network (see Section 6.1.2). Denote the block of N transmissions at node j in the superposition network by an N-dimensional transmit vector $\underline{x}_j$, and similarly the received vector by $\underline{y}_j^{'}$. All signals in the superposition network are a (deterministic) function of the codeword $\underline{x}_0$ transmitted by the source.

Next, we build a $(2^{mNR}, mN)$ code, denoted by $\mathcal{C}_0$, for the superposition network, with every mN-length codeword constructed by adjoining m codewords from the old code, for a large m. This is again a rate R code since it simply uses the old code m times on the superposition network. We can visualize the construction of codewords in $\mathcal{C}_0$ by referring to Figure 6.1. In the $(2^{mNR}, mN)$ code, node j

1. breaks up its received signal, denoted by $\underline{\mathbf{y}}_j^{'}$, into m blocks of length N,

2. applies the mapping used in the $(2^{NR}, N)$ code on each of the m blocks to generate m blocks of transmit signals,

3. and adjoins m blocks of transmit signals to construct a new transmit signal, denoted by $\underline{\mathbf{x}}_j$, of length mN.

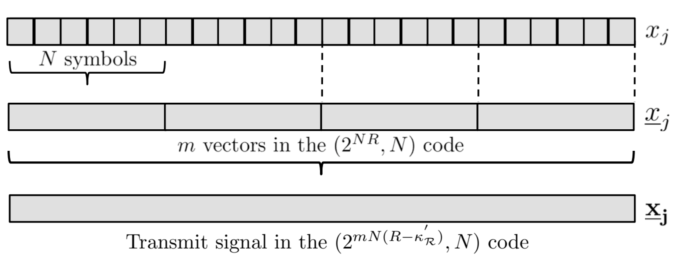


Figure 6.1: Relationship among the signals transmitted by node j.

As shown in Figure 6.1, the relationship between various signals associated with the transmission of node j is akin to packetization in computer networks.

A subset of the codewords of $\mathcal{C}_0$, defined below, forms the set of codewords of the code for the Gaussian relay network.

Pruning the set of codewords: Node j has a finite set of ϵ-strongly typical $\underline{\mathbf{y}}_j^{'}$'s (see [38]) in the code for the superposition network. We randomly, i.e., independently and uniformly, pick a $2^{-m(N\kappa+2\eta)}$ fraction of them and denote the resulting set by $\mathcal{S}_j$. Later in (6.6), $\kappa > 0$ is defined as a function only of the number of nodes in the network and not the channel gains, while $\eta > 0$ is specified later and can be made arbitrarily small. We repeat this pruning procedure for all the nodes.

Denote the intersection of the inverse images of $\mathcal{S}_j$ in $\mathcal{C}_0$, for $j = 1, 2, \cdots, M$, by $\mathcal{C}_G$. Transmission of any vector in $\mathcal{C}_G$ results in the received vector at node j belonging to $\mathcal{S}_j$ in the superposition network. The set of codewords for the Gaussian network is formed by $\mathcal{C}_G$.

Encoding and decoding procedure in the Gaussian network: The source in the Gaussian network transmits a codeword $\underline{\mathbf{x}}_\mathbf{0}$ from $\mathcal{C}_G$. Assume throughout that node 1 can listen, i.e., has a link, to the source. Node 1 receives a noisy signal and decodes to a vector in $\mathcal{S}_1$. We will specify in Section 6.1.2 how this decoding is to be done. Then, using the encoding function from $\mathcal{C}_0$, it constructs its transmit signal. All relay nodes operate in a similar way. Finally, the destination decodes its noisy reception to a signal in $\mathcal{S}_M$, and maps it to a codeword by simply using the decoding function from $\mathcal{C}_0$.

Note that we are operating the Gaussian network over the digital interface naturally defined by the signals transmitted and received in the corresponding superposition network. We summarize the main result concerning the lifting procedure in Theorem 11.

Theorem 11. *Consider a Gaussian relay network with a single source-destination pair and* $(M-1)$ *relay nodes, and consider a code for the superposition model of the network that communicates at a rate* R.

Then, the lifting procedure and the digital interface defined by the superposition model yield *a code for the original Gaussian network that communicates at a rate* $R - \kappa_{\mathcal{R}}$*, where*

$$\kappa_{\mathcal{R}} \; := \; M(\log(6M-1)+10). \tag{6.1}$$

It should be noted that $\kappa_{\mathcal{R}}$ does not depend on the channel gains. Therefore, Theorem 11 provides a lifting procedure that attains a rate in the Gaussian network within a bounded amount of R at any SNR.

6.1.1 A genie-based argument

Before delving into the details of the proof, we start with a genie-based argument explaining the ideas behind the proof of Theorem 11. The theorem is subsequently proved in detail in Section 6.1.2. The arguments presented next are not a proof of the theorem, but are the basis of our understanding of the relationship between the Gaussian and the superposition model, and motivate the search for a technique to lift a code from the superposition network to the Gaussian network.

Consider the relay network in Figure 3.1. For simplicity, assume node 1 transmits a symbol x_1 and node 2 transmits a symbol x_2 (instead of a block of symbols each) from the alphabet for the superposition network. In the superposition model of the relay network, node 3 receives

$$y_3'' \; = \; \widehat{\hat{h}_{13}x_1} + \widehat{\hat{h}_{23}x_2},$$

where $\hat{h}_{13}$ and $\hat{h}_{23}$ are the quantized channel gains (see Section 3.2.2), and it receives

$$y_3 \; = \; h_{13}x_1 + h_{23}x_2 + z_3$$

in the Gaussian relay network. Rewriting y_3, we get

$$\begin{aligned} y_3 &= y_3^{''} + (h_{13}x_1 - \hat{h}_{13}x_1) + (\hat{h}_{13}x_1 - \widehat{\hat{h}_{13}x_1}) \\ &\quad + (h_{23}x_2 - \hat{h}_{23}x_2) + (\hat{h}_{23}x_2 - \widehat{\hat{h}_{23}x_2}) + z_3 \\ &=: y_3^{''} + v_3 + z_3. \end{aligned} \tag{6.2}$$

Here we have replaced the actual values of the channel gains with appropriate variables. By definition $y_3^{''}$ lies in $\mathbb{Z} + \imath\mathbb{Z}$. Hence $y_3^{''}$ can be recovered from y_3, the quantized values $\hat{v}_3$ and $\hat{z}_3$, respectively, of v_3 and z_3, and the quantized carry c_3 obtained from adding the fractional parts of v_3 and z_3, with

$$c_3 := (v_3 - \widehat{\hat{v}_3) + (z_3} - \hat{z}_3). \tag{6.3}$$

So,

$$y_3^{''} = \hat{y}_3 - \hat{v}_3 - \hat{z}_3 - c_3,$$

and

$$H(y_3^{''}|y_3) \leq H(\hat{v}_3) + H(\hat{z}_3) + H(c_3). \tag{6.4}$$

Now, let

$$v_3 = w_{13} + w_{23},$$

where

$$w_{k3} := (h_{k3}x_k - \hat{h}_{k3}x_k) + (\hat{h}_{k3}x_k - \widehat{\hat{h}_{k3}x_k}), \ k = 1, 2.$$

Since $|h_{k3} - \hat{h}_{k3}| \leq \sqrt{2}$ and $|x_k| \leq 1$, the magnitude of v_3 is less than $2(2\sqrt{2})$. The quantized values $\hat{v}_{3R}$ and $\hat{v}_{3I}$ lie in $\{-5, -4, \ldots, 5\}$, and $H(\hat{v}_3) \leq \log(22)$. The real and imaginary parts of the carry lie in $\{0, \pm 1\}$, hence $H(c_3) \leq 3$. Since z_3 is distributed as

$\mathcal{CN}(0,1)$, from Lemma 15 in Section 6.3, $H(\hat{z}_3) \leq 6$. Adding up all the entropies and substituting in (6.4), we get the upper bound

$$H(y_3^{''}|y_3) \leq 14.$$

These computations can be repeated for all the nodes in the Gaussian network. In general, if there are M incoming signals at a relay node j, then the magnitude of v_j is less $M(2\sqrt{2})$, where v_j is similarly defined, as in (6.2), with respect to the signal received by node j. Hence, $\hat{v}_{jR}$ and $\hat{v}_{jI}$ will lie in the set $\{-3M+1, -3M+2, \ldots, 3M-1\}$;then

$$\begin{aligned} H(y_j^{''}|y_j) &\leq H(\hat{v}_j) + H(\hat{z}_j) + H(c_j) \\ &\leq \log(6M-1) + 10, \end{aligned} \tag{6.5}$$

where c_j is defined, as in (6.3), with respect to the signal received by node j. Let

$$\kappa := \log(6M-1) + 10 \tag{6.6}$$

be a function of the total number of nodes and independent of channel gains (or SNR). Now we use a code designed for the superposition network in the Gaussian network. If there were a genie providing $H(y_j^{''}|y_j)$ bits of data corresponding to the received signal to node j in every channel use, then node j can recover $\underline{\mathbf{y}}_{\mathbf{j}}^{''}$ from $\underline{\mathbf{y}}_{\mathbf{j}}$. Since the genie provides at most κ bits to every node, it provides a total of at most $M\kappa =: \kappa_{\mathcal{R}}$ bits per channel use.

Hence, with the genie's aid, a code designed for the superposition network can be used in the Gaussian network at any SNR. Our proof below prunes a fraction of the codewords representing the information that the genie would have provided, so that the decoding can work even without the genie.

6.1.2 Proof of Theorem 11

Zero probability of error

Consider the $(2^{NR}, N)$ code for the superposition network and assume that it has an average probability of error δ, where $0 \leq \delta < 1/2$. Since the superposition network is a noiseless network, each codeword is either always decoded correctly or always decoded incorrectly. Since $\delta < 1/2$, less than half of the codewords are always decoded incorrectly. Discarding them results in a code where all codewords can be successfully decoded, with a small loss in the rate. So, without loss of generality, we assume that the $(2^{NR}, N)$ code (and thus also the $(2^{mNR}, mN)$ code) for the superposition network has zero probability of error.

The random variable corresponding to the codeword, $\underline{x}_0$, has a uniform distribution with $H(\underline{x}_0) = NR$, and induces a distribution on the remaining variables in the network.

Operating over blocks of length mN

In the $(2^{mNR}, mN)$ code, we assume that every node buffers mN of its received symbols, eventually constructing a transmit signal of length mN, and transmits it over the next mN channel uses. For the layered network in Figure 3.1, this is possible since nodes can be grouped into levels such that only nodes at one level communicate with another level. For example, nodes 1 and 2 in Figure 3.1 can buffer their reception until node 0 completes its transmission, then construct their transmit signals, and transmit to nodes 3 and 4 over the next mN channel uses. For a general network, we need to differentiate between signals received by a node at various time instants to account for causality in construction of their transmit signals. This requires slightly modifying the procedure (see Section 6.1.2).

Pruning the code with respect to node 1

Each $\underline{\mathbf{y}}_{\mathbf{j}}^{''}$ (or $\underline{\mathbf{x}}_{\mathbf{j}}$) in $\mathcal{C}_0$ is generated by n independent samples from the distribution of $\underline{y}_j^{''}$ (or $\underline{x}_j$). Choose $\epsilon > 0$. For a sufficiently large m, node 1 has a collection of at most $2^{m(H(\underline{y}_1^{''})+\epsilon_2)}$

and at least $2^{m(H(\underline{y}_1^{''})-\epsilon_2)}$ ϵ-strongly typical received vectors in the superposition network corresponding to $\mathcal{C}_0$ (see [38]), where $\epsilon_2 > 0$. As $\epsilon \to 0$, $\epsilon_2 \to 0$. With η set to ϵ_2, we construct $\mathcal{S}_1$ by randomly selecting a $2^{-m(N\kappa+2\eta)}$ fraction of this collection. We do this by choosing a subset uniformly among all the subsets of the appropriate size. The set $|\mathcal{S}_1|$ can be upper bounded as follows (see (6.5)–(6.6)):

$$\begin{aligned} |\mathcal{S}_1| &\leq 2^{m(H(\underline{y}_1^{''})+\epsilon_2)}\, 2^{-m(N\kappa+2\eta)} \\ &\leq 2^{m(H(\underline{y}_1^{''})-H(\underline{y}_1^{''}|\underline{y}_1)-\epsilon_2)} = 2^{m(I(\underline{y}_1^{''};\underline{y}_1)-\epsilon_2)}. \end{aligned}$$

Similarly, we can show that $|\mathcal{S}_1| \geq 2^{m(H(\underline{y}_1^{''})-N\kappa-3\epsilon_2)}$.

For a large n, the number of codewords in $\mathcal{C}_0$ jointly ϵ-strongly typical with a particular $\underline{\mathbf{y}}_1^{''}$ can be bounded independently of the chosen $\underline{\mathbf{y}}_1^{''}$ (see [38]). The desired set has $2^{m(H(\underline{x}_0|\underline{y}_1^{''})\pm\epsilon_2)}$ codewords for a particular $\underline{\mathbf{y}}_1^{''}$, i.e., transmission of one of those codewords in the superposition network results in node 1 receiving the chosen $\underline{\mathbf{y}}_1^{''}$ (see Figure 6.2). Due to the deterministic nature of the channel, the sets of codewords in $\mathcal{C}_0$ jointly typical with two different vectors in $\mathcal{S}_1$ form disjoint sets. To construct $\mathcal{C}_{0,1}$, we pick the set of all codewords in $\mathcal{C}_0$ that are jointly ϵ-strongly typical with some vector in $\mathcal{S}_1$. We have

$$\begin{aligned} |\mathcal{C}_{0,1}| &= \sum_{\underline{\mathbf{y}}_1^{''}\in\mathcal{S}_1} \begin{array}{l}\text{(number of codewords in } \mathcal{C}_0 \text{ jointly} \\ \epsilon\text{-strongly typical with } \underline{\mathbf{y}}_1^{''})\end{array} \\ &\leq \sum_{\underline{\mathbf{y}}_1^{''}\in\mathcal{S}_1} 2^{m(H(\underline{x}_0|\underline{y}_1^{''})+\epsilon_2)} \\ &\leq 2^{m(H(\underline{y}_1^{''})-N\kappa-\epsilon_2)} \times 2^{m(H(\underline{x}_0|\underline{y}_1^{'})+\epsilon_2)} \qquad (6.7) \\ &= 2^{m(H(\underline{x}_0,\underline{y}_1^{''})-N\kappa)} \qquad (6.8) \\ &= 2^{m(H(\underline{x}_0)-N\kappa)}, \qquad (6.9) \end{aligned}$$

where (6.9) follows since $H(\underline{y}_1^{''}|\underline{x}_0) = 0$. Similarly, we can show that $|\mathcal{C}_{0,1}| \geq 2^{m(H(\underline{x}_0)-N\kappa-4\epsilon_2)}$.

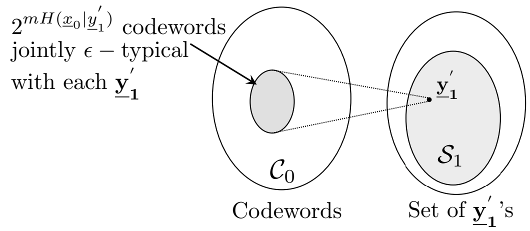


Figure 6.2: Graphic representation of pruning with respect to node 1.

If the source transmits a codeword from $\mathcal{C}_{0,1}$ in the Gaussian network, then the signal $\underline{\mathbf{y}}_1$ received by node 1 can be regarded as a noisy version of the signal $\underline{\mathbf{y}}_1^{''}$ it would have received in the superposition network, as shown in (6.2). Therefore, we define a channel with input $\underline{y}_1^{''}$ and output $\underline{y}_1$. Node 1 decodes by finding a vector in $\mathcal{S}_1$ that is jointly weakly ϵ-typical with the received vector in the Gaussian network.[1] Since $|\mathcal{S}_1| \leq 2^{m(I(\underline{y}_1^{''};\underline{y}_1)-\epsilon_2)}$, decoding is successful with block error probability less than ζ, where $\zeta \rightarrow 0$ as $n \rightarrow \infty$.

Further pruning the set of codewords with respect to node 2

There are $2^{m(H(\underline{y}_2^{''}|\underline{y}_1^{''})\pm\epsilon_2)}$ vectors in the set of $\underline{\mathbf{y}}_2^{''}$'s at node 2 that are jointly ϵ-strongly typical with a particular $\underline{\mathbf{y}}_1^{''} \in \mathcal{S}_1$. Since we constructed $\mathcal{S}_2$ by randomly choosing a subset containing a $2^{-m(N\kappa+2\epsilon_2)}$ fraction of the set of all $\underline{\mathbf{y}}_2^{''}$'s, for a large n, there are $2^{m(H(\underline{y}_2^{''}|\underline{y}_1^{''})-N\kappa\pm3\epsilon_2)}$ vectors in $\mathcal{S}_2$ jointly ϵ-strongly typical with each $\underline{\mathbf{y}}_1^{''} \in \mathcal{S}_1$. Hence, there are $2^{m(H(\underline{y}_1^{''},\underline{y}_2^{''})-2N\kappa\pm6\epsilon_2)}$ jointly ϵ-strongly typical vectors in $\mathcal{S}_1 \times \mathcal{S}_2$ with high probability (whp) as $n \rightarrow \infty$.

Now, $2^{m(H(\underline{x}_0|\underline{y}_1^{''},\underline{y}_2^{''})\pm\epsilon_2)}$ codewords in $\mathcal{C}_0$ are jointly ϵ-strongly typical with each ϵ-strongly typical tuple in $\mathcal{S}_1 \times \mathcal{S}_2$ (see Figure 6.3). We iterate the procedure in the previous subsection by collecting the codewords in $\mathcal{C}_0$ which are jointly ϵ-strongly typical with the ϵ-strongly typical tuples in $\mathcal{S}_1 \times \mathcal{S}_2$, and denote this set by $\mathcal{C}_{0,1,2}$. Naturally, $\mathcal{C}_{0,1,2}$ is a subset of $\mathcal{C}_{0,1}$. As in (6.7)–(6.9), we obtain $|\mathcal{C}_{0,1,2}|$ is about $2^{m(H(\underline{x}_0)-2N\kappa\pm7\epsilon_2)}$ whp.

[1] Since $\underline{\mathbf{y}}_1$ is a continuous signal, we use weak typicality to define the decoding operation. Note that strongly typical sequences are also weakly typical; hence sequences in $\mathcal{S}_1$ are weakly typical.

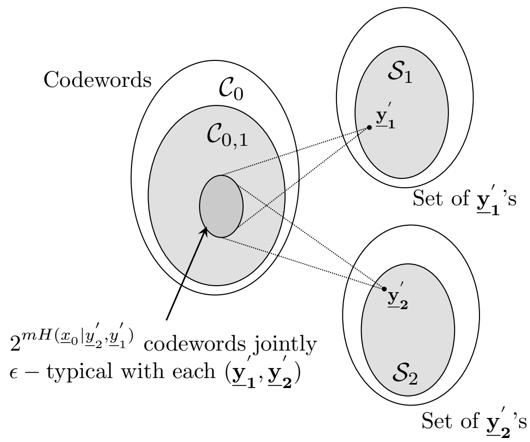


Figure 6.3: Pictorial representation of further pruning with respect to node 2.

If the source transmits a codeword from $\mathcal{C}_{0,1,2}$, then nodes 1 and 2 can correctly decode to vectors in $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively, with high probability for a large n, since $|\mathcal{S}_j| \leq 2^{m(I(\underline{y}_j'';\underline{y}_j)-\epsilon_2)}$ for $j \in \{1,2\}$.

Further pruning with respect to the remaining nodes

The same procedure is repeated with respect to the remaining nodes in the network. In the end, we obtain a collection of at most $2^{m(H(\underline{x}_0)-MN\kappa+\epsilon_M)}$ and at least $2^{m(H(\underline{x}_0)-MN\kappa-\epsilon_M)}$ codewords whp, denoted by $\mathcal{C}_{0,1,\cdots,M} =: \mathcal{C}_G$, where $\epsilon_M > 0$. Note that $\epsilon_M \to 0$ as $\epsilon \to 0$. Transmission of a codeword in $\mathcal{C}_G$ results in the received signal at node j in the superposition network belonging to the set $\mathcal{S}_j$.

Now, if $\mathcal{C}_G$ is used on the Gaussian network with encoding and decoding procedures at all nodes as described above, then the destination can decode to the transmitted codeword whp. Thus, on the Gaussian network, $\mathcal{C}_G$ achieves the rate

$$\begin{aligned} H(\underline{x}_0)/N - M\kappa - \epsilon_M/M &= R - M\kappa - \epsilon_M/M \\ &= R - \kappa_{\mathcal{R}} - \epsilon_M/M, \end{aligned} \tag{6.10}$$

where ϵ_M can be made arbitrarily small.

Interleaving the codewords for general networks

As mentioned in Section 6.1.2, we need to slightly modify the lifting procedure for relay networks which have irregular level sets that do not permit straightforward buffering of received symbols at a node. In this case, codewords in $\mathcal{C}_0$ are constructed by adjoining N blocks of m symbols each, where the first block $\underline{x}_0(1)$ consists only of the first symbols of m codewords of the $(2^{NR}, N)$ code, the second block $\underline{x}_0(2)$ consists only of the second symbols of the same codewords, and so on. The source transmits $\underline{x}_0(t)$'s in the order of increasing t.

In the $(2^{NR}, N)$ code, let $y_j^{''}(t)$, $t = 1, \ldots, N$, denote the t-th symbol received by node j. We adjoin the t-th received symbols from m uses of the code to construct $\underline{y}_j^{''}(t)$. Since $x_j(t)$ is the t-th symbol transmitted by node j and is a function of $\{y_j^{''}(p)\}_{p=1}^{t-1}$, node j can construct $\underline{x}_j(t)$, the vector consisting of the t-th transmit symbols from m uses of the code, after receiving $\{\underline{y}_j^{''}(p)\}_{p=1}^{t-1}$. Essentially, we interleave the symbols from m uses of the same code to ensure that the nodes can buffer their receptions.

In order to lift the coding scheme to the Gaussian network, we prune $\mathcal{C}_0$ by randomly picking a $2^{-m(\kappa+2\eta)}$-fraction of the set of ϵ-strongly jointly typical $(\underline{y}_j^{''}(t))_t$, for all j, and collecting the codewords jointly ϵ-strongly typical with them to form $\mathcal{C}_G$. In the Gaussian network, each node buffers its reception for m time units, decodes to the appropriate $\underline{y}_j^{''}(t)$, constructs $\underline{x}_j(t+1)$, and transmits it over the next m time units. The destination decodes individual m-length blocks to get $\underline{y}_M^{''}(t)$, $t = 1, 2, \ldots, N$, and decodes to a codeword in $\mathcal{C}_G$ after de-interleaving $\{\underline{y}_M^{''}(t)\}$.

This completes the proof of Theorem 11.

6.2 Discrete superposition model as an interface for Gaussian networks

Next, we prove that the superposition model serves as a near-optimal interface for operating Gaussian networks.

Theorem 12. *The capacity of a Gaussian relay network and the corresponding superposition network are within a bounded gap of $O(M \log M)$ bits. Also, the cut-set bound is the capacity of the superposition relay network up to a bounded gap of $O(M \log M)$ bits. Furthermore, the bounded gaps are independent of channel gains or SNR.*

Proof. We know that the expression in the cut-set bound when evaluated for the restricted choice of product input distributions is achievable for a general deterministic network [7]. In Lemma 2, we proved that the cut-set bounds of the Gaussian and the superposition network are within a gap of at most $O(M \log M)$ bits. This also proves that the cut-set bound of the superposition network (evaluated for a product distribution) is approximately its capacity. To make this rigorous:

- Start with the optimal input distribution for the superposition network.
- Then, as in Section 4.2.2, consider a particular cut and convert the optimal distribution to an input distribution for the Gaussian network with a loss of at most $O(M \log M)$ bits.
- Next replace this with the optimal joint Gaussian distribution for the inputs in the Gaussian network, with no loss in the mutual information.
- Then, choose the i.i.d. Gaussian distribution for the inputs with a loss of at most $O(M)$ bits in the mutual information across the cut [7].
- Finally, as in Section 4.2.2, reduce this to a product distribution for the superposition network with a loss of at most $O(M \log M)$ bits.

Hence, when the cut-set bound expression is evaluated for the choice of the product distribution for the superposition network, it is at most $O(M \log M)$ bits lower than the cut-set bound with the optimal distribution.

Now, the cut-set bounds of the Gaussian and superposition networks are within a bounded gap of $O(M \log M)$ bits and are also achievable up to a bounded gap of $O(M \log M)$ bits. This proves the theorem. □

In this chapter, we described a simple lifting procedure to translate any code for the superposition network into a code for the Gaussian relay network. The lifting procedure applies to any coding scheme for the superposition network and, in particular, to a near-optimal scheme. Since the capacities of the Gaussian and superposition networks are within a bounded gap, a near-optimal scheme for the superposition network can be lifted to obtain a near-optimal coding scheme for the Gaussian network, thereby proving that the superposition model can be used as a digital interface for Gaussian relay networks. We can construct an approximately-optimal linear network coding scheme for the superposition network along lines similar to the linear network code for the discrete Gaussian network presented earlier; but we skip this to avoid repetition.

6.2.1 MIMO relay networks

MIMO relay networks can be handled by treating each transmitted/received signal as a collection of vectors, where the size of the collection depends on the number of transmit/receive antennas (see Section 3.2.1 for the details). We state the counterpart of Lemma 11 and Theorem 12 for MIMO relay networks.

Theorem 13. *Consider a relay network where every node has a maximum of $T_{\max}$ transmit or receive antennas. The capacity of the Gaussian relay network and the capacity of the superposition relay network is within a bounded gap of $O(T_{\max} M \log(T_{\max} M))$ bits.*

Furthermore, a coding scheme for the superposition MIMO relay network can be lifted to the Gaussian MIMO relay network with a loss of $O(T_{\max} M \log(T_{\max} M))$ bits in the rate. Both the bounded gaps are independent of channel gains or SNR.

Proof. The arguments in the proofs of Lemma 11 and Theorem 12 can be reused here. A simple way to derive the above results is to replace each antenna at a transmitter or receiver by a virtual node. Now there are a total of at most $O(T_{\max}M)$ virtual nodes in the network. The bounds in the theorem follow by replacing M with $T_{\max}M$ in Theorem 12 and Lemma 11. In order to lift the code from the superposition MIMO network to the Gaussian relay network, we need to prune the source's codebook with respect to all the virtual nodes. □

6.2.2 Multicast networks

We defined multicast networks in Section 3.1.1. We can also operate multicast Gaussian networks on the digital interface defined by the superposition model. This is summarized in the theorem below.

Theorem 14. *The capacities of the multicast Gaussian network and the multicast superposition network are within a bounded gap of $O(M \log M)$ bits. Furthermore, a coding scheme for the multicast superposition network can be lifted to the Gaussian multicast relay network with a loss of at most $O(M \log M)$ bits in the rate.*

Proof. In order to prove the bounded gap in the capacities of the two networks, we first prove that the cut-set bounds are within a bounded gap. Here we can reuse the proof of Lemma 2. Now the cut-set bound is approximately achievable for the multicast Gaussian network, and it is achievable within a bounded gap for the multicast superposition network if we evaluate the expression in the cut-set bound for the choice of product distributions [7]. With arguments similar to those in the proof of Theorem 12, we can complete the proof of the bounded gap of $O(M \log M)$ bits between the capacities of the two networks.

The key to lifting a coding scheme from the superposition to the Gaussian network is that all the intended destinations in a multicast network are decoding the same data. Hence if we prune the source's codebook with respect to all the nodes, as in the proof of Theorem 11, then the pruned code can be decoded on the Gaussian network. We skip the details.

We can extend the above theorem to the case when the nodes have multiple transmit and receive antennas. □

6.3 Maximum entropy

Lemma 15. Let x be a random variable whose domain is $\mathbb{Z} + \imath\mathbb{Z}$, with $E[|x|^2] \leq 1$. Then the entropy of x is bounded as $H(x) \leq 6$.

Proof. The entropy of x can be bounded as

$$\begin{aligned} H(x) &= H(x_R + \imath x_I) \\ &\leq H(x_R) + H(x_I). \end{aligned}$$

Now

$$\begin{aligned} H(x_R) &= H(\text{sign}(x_R), |x_R|^2) \\ &\leq H(\text{sign}(x_R)) + H(|x_R|^2) \\ &\leq 1 + H(|x_R|^2). \end{aligned}$$

Let $z = |x_R|^2$. The domain of z is the set of non-negative integers. Since $E[|x|^2] \leq 1$, we have $E[z] = E[|x_R|^2] \leq 1$. If $E[z] < 1$, then $p(z = 0) > 0$. Since the alphabet of z is countably infinite, there is always a $k \in \mathbb{Z}^+$ with $p(z = k) < p(z = 0)$, with $p(z = k)$ possibly zero. Now mixing the probability distribution of z by replacing each of $p(z = 0)$ and $p(z = k)$ by the average of the two probabilities will increase the entropy (see [38]), and will also increase the mean. Hence the entropy of z, subject to $E[z] \leq 1$, is maximized when $E[z] = 1$. For a given mean, the geometric distribution maximizes the entropy among all discrete distributions (see [38]). Since the entropy of a geometric random variable over the non-negative integers with unit mean is 2, $H(|x_R|^2) = H(z) \leq 2$.

Hence $H(x_R) \leq 3$. Similarly, we can prove that $H(x_I) \leq 3$. Combining the two bounds, we get the statement of the lemma. $\square$

CHAPTER 7

A DIGITAL INTERFACE FOR GAUSSIAN INTERFERENCE NETWORKS

In this chapter, we prove that the capacity regions of the Gaussian interference channel and the superposition interference channel are within a bounded number of bits, with the bound independent of channel gains or SNR. This result was proved for the case of 2×2 interference channel (and a slightly different deterministic model) in [33]. Also, similarly to the case of the relay network, we develop a systematic way to "lift" any code for the superposition interference network to the Gaussian interference network, and establish that it does so with no more than a bounded loss in the rate. We could present similar results relating the capacity region and codes for the discrete interference network with those of the Gaussian interference network, but we skip them.

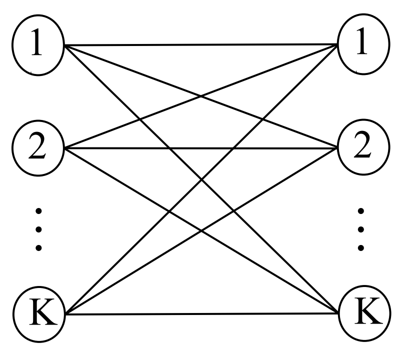


Figure 7.1: $K \times K$ interference network.

7.1 General code for interference networks

Consider the $K \times K$ interference network shown in Figure 7.1. Consider a block code for the interference network, either the Gaussian version or the superposition version. Such a $(2^{NR_1}, 2^{NR_2}, \ldots, 2^{NR_K}, N)$ code for the interference channel is defined by an encoding function for each source

$$\underline{x}_k : \{1, 2, \ldots, 2^{NR_k}\} \rightarrow \mathcal{X}^N, \text{ for } k = 1, 2, \ldots, K,$$

where $\mathcal{X}$ is the input alphabet of the channel, and a decoding function for each destination

$$g_k : \mathcal{Y}_k^N \rightarrow \{1, 2, \ldots, 2^{NR_k}\}, \text{ for } k = 1, 2, \ldots, K,$$

where $\mathcal{Y}_k$ is the output alphabet of the channel at destination node k. Let $\mathcal{M}_j$ be a random variable uniformly distributed on $\{1, 2, \ldots, 2^{NR_j}\}$, for each j, corresponding to the message that source j wants to communicate to destination j. $\mathcal{M}_j$ is mapped to the codeword $\underline{x}_j(\mathcal{M}_j)$. The average probability of error is given by

$$P_e = Pr(g_k(\underline{y}_k) \neq \mathcal{M}_k, \text{ for some } k),$$

where $\underline{y}_k$ is the signal received by destination k. The capacity region is the collection of all rate tuples $\underline{R}$ such that for any $\epsilon > 0$, there exists a blocklength N for which $P_e < \epsilon$.

All rate vectors in the capacity region are referred to as achievable rate vectors. The next lemma states the equivalence between achievable rate vectors and a collection of multi-letter mutual information terms.

Lemma 16. *A rate vector $\underline{R}$ lies in the capacity region of the $K \times K$ interference channel if and only if there exists a blocklength N, and a collection of independent random variables $\{\underline{x}_k, k = 1, 2, \ldots, K\}$ such that*

$$R_k < \frac{1}{N} I(\underline{x}_k; \underline{y}_k), \text{ for } k = 1, 2, \ldots, K, \tag{7.1}$$

where $\underline{y}_k$ is the received signal at the k-th receiver.

Proof. Suppose we are given a set of distributions $\{p(\underline{x}_k)\}$ on input vectors of length N. Consider the k-th source-destination pair and construct a collection of 2^{mNR} codewords for the k-th source where each codeword is constructed by independently picking m vectors, each of length N, with the distribution $p(\underline{x}_k)$, and appending them to get codewords of length mN. All the sources similarly construct their set of codewords. The k-th destination decodes by looking for a codeword potentially transmitted by the k-th source jointly typical with its reception. Using standard random coding arguments (see [38]), the probability of error can be made arbitrarily small by considering large enough m. This proves the achievability of the rate.

For proving the converse, fix a $(2^{NR_1}, 2^{NR_2}, \ldots, 2^{NR_K}, N)$ code and observe

$$\begin{aligned} NR_k &= H(\mathcal{M}_k) \\ &= I(\mathcal{M}_k; \underline{y}_k) + H(\underline{y}_k | \mathcal{M}_k) \\ &\leq I(\mathcal{M}_k; \underline{y}_k) + 1 + P_e N R_k \qquad (7.2) \\ &\leq I(\underline{x}_k; \underline{y}_k) + 1 + P_e N R_k, \qquad (7.3) \end{aligned}$$

where we used Fano's lemma in (7.2) and the data processing inequality in (7.3). Since the rate R_k is achievable, P_e can be made arbitrarily small for sufficiently large N. On dividing both sides of (7.3) by N, we get the converse. □

7.2 A digital interface for Gaussian interference networks

The main result of this section is the following theorem.

Theorem 17. *Consider the $K \times K$ interference channel described above. The capacity region of the Gaussian interference channel and the superposition interference channel are within a bounded gap, where the gap is independent of channel gains or SNR.*

If $\underline{R}_G$ is a rate vector in the capacity region of the Gaussian interference channel, then there is a rate vector $\underline{R}_{DS}$ in the capacity region of the superposition interference channel such that

$$|R_{k,G} - R_{k,DS}| \ \leq \ \kappa_{\mathcal{I}}, \ \textit{for } k = 1, 2, \ldots, K, \tag{7.4}$$

where

$$\kappa_{\mathcal{I}} \ := \ 6K + \log(144K + 1). \tag{7.5}$$

Conversely, a coding scheme for the superposition interference channel corresponding to a rate vector $\underline{R}_{DS}$ can be lifted to the Gaussian interference channel to obtain an achievable rate vector $\underline{R}_G$ with a loss of at most $\kappa^{'}_{\mathcal{I}}$ bits, where

$$\kappa^{'}_{\mathcal{I}} \ := \ \log(6K - 1) + 10. \tag{7.6}$$

Proof. We prove the theorem in a series of steps. First, using Lemma 16, we convert any achievable rate tuple for either the Gaussian or superposition network into a set of mutual information terms. Then, by analyzing these mutual information terms, we prove that any coding scheme for the Gaussian channel can be transformed into a coding scheme for the superposition channel and vice versa, with at most a bounded loss in the rate. This will contain the proof of the lifting procedure to translate any coding scheme for the superposition interference channel to the Gaussian interference channel, again with at most a bounded loss in the rate. The proof of the theorem is split over Sections 7.2.1, 7.2.2, and 7.2.3.

7.2.1 Capacity(Gaussian interference channel) $\subseteq$ Capacity(superposition), to within $O(\log K)$ bits

We convert a coding scheme for the superposition interference channel to a mutual information expression using Lemma 16, and bound the loss in the mutual information when transforming the superposition model to the Gaussian model. The proof below contains all the ideas on lifting a code from the superposition model to the Gaussian model, and we explicitly mention them in Section 7.2.3.

Pick any coding scheme for the superposition interference channel which achieves the rate tuple $(R_{k,S})_k$. From Lemma 16, we know that this corresponds to a collection of mutual information terms $\{I(\underline{x}_k;\underline{y}_k^{''}),\ k = 1, 2, \ldots, K\}$. Note that here $\underline{x}_k$ corresponds to the input of the k-th source in the superposition interference network, and $\underline{y}_k^{''}$ corresponds to the output of the k-th destination in the superposition network. Now, we use the same input distribution on the Gaussian interference network and show that the mutual information corresponding to the rate of the k-th source does not decrease by more than a bounded amount. The same arguments will be applicable to the remaining mutual information terms.

Since every element of the vector $\underline{x}_k$ satisfies a peak power constraint, it also satisfies an average power constraint. Hence it can be used as an input to the Gaussian channel to get

$$\underline{y}_k = \sum_{j=1}^{K} h_{jk}\,\underline{x}_j + \underline{z}_k. \tag{7.7}$$

We know that

$$\underline{y}_k^{''} = \sum_{j=1}^{K} \widehat{\hat{h}_{jk}\,\underline{x}_j}, \tag{7.8}$$

where $\hat{h}_{jk}$ is the quantized channel gain. Hence, we have

$$\begin{aligned} \underline{y}_k &=: \underline{y}_k^{''} + \sum_{j=1}^{K} \underline{w}_{jk} + \underline{z}_k \\ &=: \underline{y}_k^{''} + \underline{v}_k + \underline{z}_k, \end{aligned} \tag{7.9}$$

where $\underline{w}_{jk}$ is defined as

$$\underline{w}_{jk} := (h_{jk} - \hat{h}_{jk})\,\underline{x}_j + (\hat{h}_{jk}\,\underline{x}_j - \widehat{\hat{h}_{jk}\,\underline{x}_j}).$$

By definition $\underline{y}_k^{''}$ is a vector of entries from $\mathbb{Z} + \imath\mathbb{Z}$. Hence $\underline{y}_k^{''}$ can be recovered from $\underline{y}_k$, the quantized values of $\underline{v}_k$ and $\underline{z}_k$, and the vector of carries $\underline{c}_k$ obtained from adding the fractional parts of $\underline{v}_k$ and $\underline{z}_k$. So,

$$\begin{aligned} I(\underline{x}_k; \underline{y}_k^{''}) &\leq I(\underline{x}_k; \underline{y}_k, \hat{\underline{v}}_k, \hat{\underline{z}}_k, \underline{c}_k) \\ &\leq I(\underline{x}_k; \underline{y}_k) + H(\hat{\underline{v}}_k) + H(\hat{\underline{z}}_k) + H(\underline{c}_k), \end{aligned} \tag{7.10}$$

where $\hat{\underline{v}}_k$ and $\hat{\underline{z}}_k$ are defined as quantized integer portions of $\underline{v}_k$ and $\underline{z}_k$. Similarly to the arguments preceding (6.5), we get

$$\begin{aligned} H(\hat{\underline{v}}_k) &\leq N \log(12K - 2), \\ H(\underline{c}_k) &\leq 3N, \\ H(\hat{\underline{z}}_k) &\leq 6N. \end{aligned}$$

In (7.10), $I(\underline{x}_k; \underline{y}_k)$ corresponds to N times the rate $R_{k,G}$ achieved on the Gaussian interference channel by the k-th source. Therefore we get

$$\begin{aligned} R_{k,G} &\geq R_{k,S} - \log(6K - 1) - 10 \\ &\geq R_{k,S} - O(\log K). \end{aligned}$$

7.2.2 Capacity(superposition interference channel) $\subseteq$ Capacity(Gaussian interference channel), to within an additive constant

Next we prove that the capacity region of the superposition interference network is at least as large as that of the Gaussian interference network, minus a constant number of bits. From Lemma 16, we know that we can replace any achievable rate vector $(R_{k,G})_k$ for the Gaussian interference channel by a collection of mutual information terms $\{I(\underline{x}_k; \underline{y}_k),\ k = 1, 2, \ldots, K\}$. We perform a series of approximations to convert the mutual information $I(\underline{x}_k; \underline{y}_k)$ corresponding to the rate of the k-th source to a mutual information term for the superposition interference channel, incurring a loss no greater than a bounded amount.

The received signal at the k-th receiver in the Gaussian interference channel is

$$\underline{y}_k = \sum_{j=1}^{K} h_{jk} \, \underline{x}_j + \underline{z}_k, \tag{7.11}$$

with the transmitted signals satisfying an average unit power constraint.

Note that $\underline{x}_k = (x_{k1}, x_{k2}, \ldots, x_{kN})$ and each x_{km} can be split into its quantized part $\hat{x}_{km}$ (see (3.15) for the definition of the quantization operation) and fractional part $\tilde{x}_{km}$, where

$$\tilde{x}_{km} \quad := \quad x_{km} - \hat{x}_{km}.$$

We discard $\hat{x}_{km}$ and retain $\tilde{x}_{km}$. Since x_{km} satisfies a unit average power constraint, $\tilde{x}_{km}$ also satisfies a unit average power constraint. Define

$$\underline{\tilde{y}}_j \; := \; \sum_{k=1}^{K} h_{jk} \, \underline{\tilde{x}}_k + \underline{z}_j. \tag{7.12}$$

Denote the discarded portion of the received signal by

$$\underline{\breve{y}}_j \; := \; \sum_{k=1}^{K} h_{jk} \, \underline{\hat{x}}_k. \tag{7.13}$$

Comparing the mutual information corresponding to the j-th source-destination pair for channels (7.11) and (7.12), we get

$$
\begin{aligned}
NR_{Gj} &= I(\underline{x}_j;\underline{y}_j) \\
&\le I(\underline{x}_j;\underline{\tilde{y}}_j,\underline{\breve{y}}_j) \\
&\le I(\underline{\tilde{x}}_j,\underline{\hat{x}}_j;\underline{\tilde{y}}_j,\underline{\breve{y}}_j) \\
&= I(\underline{\tilde{x}}_j,\underline{\hat{x}}_j;\underline{\tilde{y}}_j) + I(\underline{\tilde{x}}_j,\underline{\hat{x}}_j;\underline{\breve{y}}_j|\underline{\tilde{y}}_j) \\
&= I(\underline{\tilde{x}}_j;\underline{\tilde{y}}_j) + I(\underline{\tilde{x}}_j,\underline{\hat{x}}_j;\underline{\breve{y}}_j|\underline{\tilde{y}}_j) \qquad (7.14) \\
&\le I(\underline{\tilde{x}}_j;\underline{\tilde{y}}_j) + H(\underline{\breve{y}}_j) \\
&\le I(\underline{\tilde{x}}_j;\underline{\tilde{y}}_j) + \sum_{k=1}^{K} H(\underline{\hat{x}}_k), \qquad (7.15)
\end{aligned}
$$

where (7.14) follows because $\underline{\hat{x}}_j \to \underline{\tilde{x}}_j \to \underline{\tilde{y}}_j$ form a Markov chain, and (7.15) holds because $\breve{y}_{jn}$ is a function of $\{\hat{x}_{kn}\}$ from (7.13). It is proved in Lemma 15 in Section 6.3 that $H(\hat{x}_{km}) \le 6$; hence we get

$$I(\underline{\tilde{x}}_j;\underline{\tilde{y}}_j) \;\ge\; I(\underline{x}_j;\underline{y}_j) - 6KN. \qquad (7.16)$$

Since $\tilde{x}_{kmR}$ and $\tilde{x}_{kmI}$ lie in $(-1,1)$, we obtain positive inputs by adding 1 to each. This is equivalent to adding $\sum_k h_{jk}(1+\imath)$ to $\tilde{y}_{jm}$. Denoting by $\underline{\nu}$ the vector of $(1+\imath)$'s, we also divide by $2\sqrt{2}$ throughout to get

$$
\begin{aligned}
&(\underline{\tilde{y}}_j + \sum_k h_{jk}\underline{\nu})/2\sqrt{2} \\
&\quad = \sum_k h_{jk}(\underline{\tilde{x}}_k + \underline{\nu})/2\sqrt{2} + \underline{z}_j/2\sqrt{2}. \qquad (7.17)
\end{aligned}
$$

Note that

$$\begin{aligned} I(\underline{\tilde{x}}_j;\underline{\tilde{y}}_j) &= I\left(\underline{\tilde{x}}_j, \frac{\underline{\tilde{x}}_j+\underline{\nu}}{2\sqrt{2}}; \underline{\tilde{y}}_j\right) \\ &= I\left(\underline{\tilde{x}}_j, \frac{\underline{\tilde{x}}_j+\underline{\nu}}{2\sqrt{2}}; \underline{\tilde{y}}_j, \frac{\underline{\tilde{y}}_j+\sum_k h_{jk}\underline{\nu}}{2\sqrt{2}}\right) \\ &= I\left(\frac{\underline{\tilde{x}}_j+\underline{\nu}}{2\sqrt{2}}; \frac{\underline{\tilde{y}}_j+\sum_k h_{jk}\underline{\nu}}{2\sqrt{2}}\right). \end{aligned}$$

To avoid introducing new notation, for the rest of the proof we abuse notation and denote the left hand side of (7.17) by $\underline{y}_j$, $(\underline{\tilde{x}}_k+\underline{\nu})/2\sqrt{2}$ by $\underline{x}_k$, and $\underline{z}_j/2\sqrt{2}$ by $\underline{z}_j$, for all j.

With the new notation, $|x_{km}| \leq 1$, with positive real and imaginary parts, and z_{jm} is distributed as $\mathcal{CN}(0,1/8)$.

The features of the model that we next address are

1. channel gains are quantized to lie in $\mathbb{Z}+\imath\mathbb{Z}$,
2. real and imaginary parts of the scaled inputs are restricted to

$$n := \max_{(i,j)\in\mathcal{E}} \max\{\lfloor \log|h_{ijR}|\rfloor, \lfloor \log|h_{ijI}|\rfloor\}$$

   bits,

3. there is no AWGN, and
4. outputs are quantized to lie in $\mathbb{Z}+\imath\mathbb{Z}$.

Let the binary expansion of $\sqrt{2}\,x_{kmR}$ be $0.x_{kmR}(1)x_{kmR}(2)\ldots$, i.e.,

$$x_{kmR} =: \frac{1}{\sqrt{2}}\sum_{p=1}^{\infty} 2^{-p}x_{kmR}(p).$$

The received signal in the superposition channel only retains the following relevant portion of the input signals:

$$\underline{y}_k^{''} = \sum_j \widehat{\hat{h}_{jk}\,\underline{x}_j^{'}}, \tag{7.18}$$

where $\hat{h}_{jk}$ is the quantized channel gain and

$$\begin{aligned} x_{kmR}^{'} &:= \frac{1}{\sqrt{2}} \sum_{p=1}^{n} x_{kmR}(p) 2^{-p}, \\ x_{kmI}^{'} &:= \frac{1}{\sqrt{2}} \sum_{p=1}^{n} x_{kmI}(p) 2^{-p}. \end{aligned}$$

To obtain (7.18) we subtracted $\underline{\delta}_k$ from $\underline{y}_k$, where

$$\begin{aligned} \underline{\delta}_k &:= \sum_{j=1}^{K} \left(h_{jk}(\underline{x}_j - \underline{x}_j^{'}) + (h_{jk} - \hat{h}_{jk})\,\underline{x}_j^{'} + (\hat{h}_{jk}\,\underline{x}_j^{'} - \widehat{\hat{h}_{jk}\,\underline{x}_j^{'}}) \right) + \underline{z}_k \qquad (7.19) \\ &=: \sum_j \underline{w}_{jk} + \underline{z}_k \\ &=: \underline{v}_k + \underline{z}_k. \end{aligned}$$

To bound the loss in the mutual information in the superposition network from the original Gaussian interference network, we have

$$\begin{aligned} I(\underline{x}_k; \underline{y}_k) &\leq I(\underline{x}_k; \underline{y}_k, \underline{y}_k^{''}, \underline{\delta}_k) \\ &= I(\underline{x}_k; \underline{y}_k^{''}, \underline{\delta}_k) \\ &= I(\underline{x}_k; \underline{y}_k^{''}) + I(\underline{x}_k; \underline{\delta}_k | \underline{y}_k^{''}) \\ &= I(\underline{x}_k^{'}; \underline{y}_k^{''}) + h(\underline{\delta}_k | \underline{y}_k^{''}) - h(\underline{\delta}_k | \underline{y}_k^{''}, \underline{x}_k) \\ &\leq I(\underline{x}_k^{'}; \underline{y}_k^{''}) + h(\underline{\delta}_k) - h(\underline{\delta}_k | \underline{y}_k^{''}, \underline{x}_k, \underline{v}_k) \\ &= I(\underline{x}_k^{'}; \underline{y}_k^{''}) + h(\underline{\delta}_k) - h(\underline{z}_k) \\ &= I(\underline{x}_k^{'}; \underline{y}_k^{''}) + I(\underline{v}_k; \underline{\delta}_k). \end{aligned}$$

By bounding the magnitudes of the terms in (7.19), we get $|w_{jkm}| \leq 3\sqrt{2}$. So, $I(\underline{v}_k; \underline{\delta}_k)$ is the mutual information of N uses of a Gaussian MISO channel with average input power constraint less than $(3\sqrt{2})^2 K \leq 18K$ and

$$\begin{aligned} I(\underline{v}_k; \underline{\delta}_k) &\leq N \log(1 + 18K/(1/8)) \\ &< N \log(1 + 144K). \end{aligned} \tag{7.20}$$

Note that $I(\underline{x}_k^{'}; \underline{y}_k^{''})$ is the mutual information between the input and output of the j-th source-destination pair in the superposition interference channel. By Lemma 16, this mutual information translates into an achievable rate $R_{k,S} = I(\underline{x}_k^{'}; \underline{y}_k^{''})/N$. By accumulating the losses in transforming the inputs for the Gaussian channel into the corresponding inputs for the deterministic channel, we obtain

$$R_{k,S} \geq R_{k,G} - (6K + \log(1 + 144K)) \tag{7.21}$$

$$\geq R_{k,G} - O(K). \tag{7.22}$$

7.2.3 Lifting codewords to the Gaussian interference network

Since the construction of a digital interface and the procedure of lifting codewords from the superposition interference channel to the Gaussian interference channel was implicit in the proof of the above lemma, we summarize the procedure below:

- Consider any coding scheme for the superposition interference channel that achieves a rate tuple $(R_{k,S})_k$ with probability of error ϵ. Using Lemma 16, it can be converted to a collection of mutual information terms $\{I(\underline{x}_k; \underline{y}_k^{''})\}$, where the k-th term corresponds to the rate of the k-th source-destination pair, and a collection of independent input distributions $\{p(\underline{x}_k)\}$, where the k-th distribution corresponds to the input distribution for the k-th source. The input distribution $p(\underline{x}_k)$ is over N-length vectors, where N is the length of the codewords for the superposition interference channel.

- We construct an mN-length codeword for the k-th source by picking m vectors independently with the distribution $p(\underline{x}_k)$ and adjoining them. We construct $2^{mN(R_{k,S}-O(K))}$ codewords this way. This set forms the $(2^{mN(R_{k,S}-O(K))}, mN)$ code for the j-th source in the Gaussian interference network. Similarly to the lifting procedure for the relay network, we can visualize the construction of codewords in the $(2^{mN(R_{k,S}-O(K))}, mN)$ code by referring to Figure 6.1.

- In the Gaussian interference channel, with joint typical decoding, the k-th decoder can recover the codeword transmitted by the k-th source with probability of error less than ϵ, as we allow m to tend to ∞. This is essentially proved in Section 7.2.1.

Therefore, we can operate the Gaussian interference channel on the digital interface defined by the signals transmitted and received in the superposition interference channel. Now we know that the capacities of both the networks are within a bounded gap. Hence, if we choose a near-optimal coding scheme for the superposition interference network, we can transform it by following the above procedure for lifting codewords and obtain a near-optimal digital interface for operating the Gaussian interference network. □

7.2.4 MIMO interference networks

The results in Theorem 17 can be extended to $K \times K$ interference channels where each transmitter and destination have multiple antennas. Once again, the constant determining the bounded gap is a function of the number of nodes in the network, as well as the number of transmit and receive antennas at the various nodes.

Theorem 18. *Consider the $K \times K$ MIMO interference channel, where every node has a maximum of $T_{\max}$ antennas. The capacity region of the Gaussian MIMO interference channel and the superposition MIMO interference channel are within a bounded gap of $\kappa_{\mathcal{I},T}$*

bits, where

$$\kappa_{\mathcal{I},T} \;:=\; 6T_{\max}K + T_{\max}\log(144T_{\max}K+1). \tag{7.23}$$

Furthermore, a coding scheme for the superposition MIMO interference channel can be lifted to the Gaussian MIMO interference channel with a loss of $\kappa^{'}_{\mathcal{I},T}$ *bits in the rate, where*

$$\kappa^{'}_{\mathcal{I},T} \;:=\; T_{\max}(\log(6T_{\max}K-1)+10). \tag{7.24}$$

Proof. The proof of this theorem is similar to that of Theorem 17. We can treat each transmitting antenna and receiving antenna as a virtual node. For proving (7.23), we note that in this case there are at most $KT_{\max}$ virtual transmitters that contribute $6T_{\max}K$ to the bound. Since there are $T_{\max}$ virtual receivers at every node and each one contributes $\log(144T_{\max}K+1)$ to the bound, the total contribution of the virtual receivers is $T_{\max}\log(144T_{\max}K+1)$. Adding up the two contributions, we get the bound in (7.23) (see Section 7.2.2 for the details). The bound in (7.24) can be similarly proved. Each of the $T_{\max}$ virtual receivers at a particular receiver contributes $\log(6T_{\max}K-1)+10$ to the bound, with a total contribution of $T_{\max}(\log(6T_{\max}K-1)+10)$ due to all the virtual receivers at a node (see Section 7.2.1 for the details). □

CHAPTER 8

CONCLUSIONS

In this dissertation, we have presented the discrete and superposition networks that approximate the capacities of certain Gaussian networks and provide a digital interface for operating them. Our models have two important properties. First, the capacities of the Gaussian network and the corresponding discrete or superposition network are within a bounded gap, where the bound is independent of channel gains or SNR. Second, near-optimal coding schemes for the discrete or the superposition network can be naturally lifted to obtain near-optimal coding schemes for the Gaussian network. Both these properties together prove that the discrete and superposition models are near-optimal digital interfaces for a class of Gaussian networks.

8.1 Approximating Gaussian networks with the discrete model

The discrete network is a quantization-based digital interface for the class of layered Gaussian relay networks. These are networks where the nodes are arranged into groups or layers such that only nodes of one layer can transmit to nodes of another layer. We proved that a simple linear coding scheme for such layered discrete networks achieves rates within a bounded gap of $O(M^2)$ bits from the cut-set bound, where the bound is independent of channel gains or SNR. The transmit signals in the discrete network satisfy a unit power constraint, and hence are valid transmit signals for the Gaussian network. Also, the received signals in the Gaussian network are obtained by quantizing the received signals in the Gaussian network. Hence, the Gaussian network can be easily operated on the digital interface defined by the discrete network. The same linear coding scheme can be used on the layered Gaussian network, and is near-optimal for the Gaussian network.

Our overall near-optimal strategy for a Gaussian relay network can be summarized as follows. The number n of bits of precision is chosen as the logarithm of the largest real or imaginary part of any channel gain. Then by simple quantization and truncation, we create a purely discrete network. This discrete network is operated in an en bloc fashion. At each node, received signals are buffered for a block, on the basis of which the transmit signals in the next block are generated. The coding strategy is particularly simple. Linear network coding is performed by simply multiplying the buffered vector by a square random binary matrix. The resulting long vector is broken into symbols for use in the next block. A similar strategy can be used for MIMO nodes, and also for multicast relay networks.

The above strategy has the advantage that it employs a simple coding strategy requiring minimal signal processing at the relays. We note that most codes for relay networks do require considerable signal processing by the relay nodes consisting of non-linear operations such as compression, decoding, etc.

Each relay performs scalar quantization followed by a simple matrix multiplication. Non-linearity is thus introduced into the code due to quantization. Introducing some non-linearity is unavoidable due to the inability of linear codes to achieve the capacity within a constant gap in the Gaussian relay network.

The random matrix interleaves all the bits and perhaps increases the complexity of the decoding algorithm at the destination. It is an interesting problem to simplify this by constructing explicit encoding matrices for the relays that preserve the properties of a random matrix. This might help us construct a graphical model for describing the channel from the source to the destination and lead to a low-complexity code for relay networks that is decodable with iterative message passing algorithms.

The linear network code is a robust scheme in the sense that the relay need not know the channel gains on either the incoming or the outgoing links. Since the transmit and receive signals are quantized to binary tuples of length $2n$, all the nodes only need to know the global parameter n.

The quantization requirements of the linear network code are completely defined by the parameter n, which therefore also determines the resolution of the analog-to-digital convertor (ADC) and the digital-to-analog convertor (DAC) for operating the network within a bounded gap from the network capacity.

For a general Gaussian network with many sources, many destinations, relays, and any data transmission requirements, one can similarly construct the corresponding discrete counterpart and prove that the capacity region of the corresponding discrete network is contained in the capacity region of the Gaussian network. The converse, however, remains to be proved: that the capacity region of the original Gaussian network is contained in that of the discrete network, up to a constant gap.

8.2 Approximating Gaussian networks with the superposition model

We have proved that the superposition network serves as a near-optimal digital interface for designing codes for the Gaussian relay network and the Gaussian interference network. Hence, we have transformed the problem of designing near-optimal codes for the Gaussian network into one of designing near-optimal codes for the discrete counterpart. Also, the problem of computing the capacity of the Gaussian network is reduced to a combinatorial problem of computing the capacity of a discrete network. In the case of the relay network, even though we already know near-optimal coding schemes for the network, it may still be helpful to construct simple codes for its superposition counterpart. Such simple schemes can be directly translated, via the lifting procedure proposed earlier, to construct simple coding schemes for the original Gaussian network. In the case of the Gaussian interference network, computing the capacity region of the 3×3 superposition interference channel will yield the capacity region of the original Gaussian network to within a constant, and will improve our understanding of larger practical interference networks. We have not been able to prove the near-optimality of the superposition model in approximating the capacity of a general Gaussian network.

One of the main problems in network information theory is computing the capacity region of a large network with many sources, many destinations, and arbitrary data transmission requirements. Currently, this appears to be no more than a distant possibility. As suggested by [7], a possibly simpler aim is to approximate a general network with a deterministic model, perhaps with the superposition model.

Determining the Shannon capacity of relay networks and constructing codes and protocols that achieve them is a long-standing problem in network information theory. Recently progress has been made by relaxing the stringent goals of information theory, and instead solving a possibly tractable problem of determining the capacity region within a bounded gap. The models and the codes presented here are steps towards constructing simple near-optimal schemes for wireless networks. A better understanding of the limits of approximating Gaussian networks with such models will help us in computing the fundamental limits of wireless networks with many users, and may also help in designing coding schemes for them.

REFERENCES

[1] R. Ahlswede, "Multi-way communication channels," in *Proceedings of the 2nd International Symposium on Information Theory*, Budapest, Hungary, 1973, pp. 23–51.

[2] H. Liao, "Multiple access channels," Ph.D. dissertation, Department of Electrical Engineering, University of Hawaii, Honolulu, HA, 1972.

[3] T. Cover, "Broadcast channels," *IEEE Transactions on Information Theory*, vol. 18, pp. 2–6, 1972.

[4] P. Bergmans, "A simple converse for broadcast channels with additive white Gaussian noise," *IEEE Transactions on Information Theory*, vol. 20, pp. 279–280, 1974.

[5] E. C. van der Meulen, "Three-terminal communication channels," *Advances in Applied Probability*, vol. 3, pp. 120–154, 1971.

[6] T. M. Cover and A. El Gamal, "Capacity theorems for the relay channel," *IEEE Transactions on Information Theory*, vol. IT 25, pp. 572–584, Sept. 1979.

[7] A. S. Avestimehr, S. N. Diggavi, and D. N. C. Tse, "Wireless network information flow: A deterministic approach," *IEEE Transactions on Information Theory*, submitted for publication.

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

[9] T. Ho, M. Médard, R. Koetter, D. Karger, M. Effros, 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, Oct. 2006.

[10] M. Anand and P. R. Kumar, "A digital interface for Gaussian relay and interference networks: Lifting codes from the discrete superposition model to Gaussian relay networks," *IEEE Transactions on Information Theory: Special Issue on Interference Networks*, vol. 57, no. 5, pp. 2548–2564, May 2011.

[11] M. Anand and P. R. Kumar, "On approximating Gaussian relay networks with deterministic networks," in *Proc. IEEE ITW 09*, Taormina, Sicily, Oct. 2009, pp. 625–629.

[12] M. Anand and P. R. Kumar, "A digital interface for Gaussian relay networks: lifting codes from the discrete superposition model to Gaussian relay networks," in *Proc. IEEE ITW 10*, Cairo, Egypt, Jan. 2010, pp. 465–469.

[13] M. Anand and P. R. Kumar, "Linear codes for relay networks," *IEEE Transactions on Information Theory*, to be submitted for publication.

[14] A. El Gamal, "On information flow in relay networks," *IEEE National Telecommunications Conference*, vol. 2, pp. D4.1.1–D4.1.4, Nov. 1981.

[15] L.-L. Xie and P. R. Kumar, "A network information theory for wireless communication: Scaling laws and optimal operation," *IEEE Transactions on Information Theory*, vol. 50, no. 5, pp. 748–767, May 2004.

[16] L.-L. Xie and P. R. Kumar, "An achievable rate for the multiple level relay channel," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1348–1358, April 2005.

[17] M. Gastpar, G. Kramer, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Transactions on Information Theory*, vol. 51, no. 9, pp. 3037–3063, Sept. 2005.

[18] S. H. Lim, Y.-H. Kim, A. El Gamal, and S.-Y. Chung, "Noisy Network Coding," *IEEE Transactions on Information Theory: Special Issue on Interference Networks*, vol. 57, no. 5, pp. 3132–3152, May 2011.

[19] A. Raja and P. Viswanath, "Compress-and-forward scheme for a relay network: approximate optimality and connection to algebraic flows," submitted to *IEEE Transactions on Information Theory*, Dec. 2010. arXiv: 1012.0416

[20] V. Nagpal, I. Wang, M. Joragovanovic, D. Tse, and B. Nikolić "Quantize-map-and-forward relaying: Coding and system design," in *Proceedings of Allerton Conference on Communications, Control, and Computing*, Monticello, IL, Sept. 2010, pp. 443–450.

[21] T. K. Dikaliotis, H. Yao, S. Avestimehr, S. Jaggi, and T. Ho, "Low-complexity near-optimal codes for Gaussian relay networks," Feb. 2011. [Online]. arXiv: 1102.5138.

[22] A. Özgür and S. Diggavi, "Approximately achieving Gaussian relay network capacity with lattice codes," in *Proceedings of International Symposium on Information Theory (ISIT)*, June 2010, pp 669–673.

[23] S. Kannan, A. Raja, and P. Viswanath, "Approximately optimal wireless broadcasting," *IEEE Transactions on Information Theory*, submitted for publication.

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

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

[26] R. Etkin, D. N. C. Tse, and H. Wang, "Gaussian interference channel capacity to within one bit," *IEEE Transactions on Information Theory*, vol. 54, no. 12, pp. 5534–5562, Dec. 2008.

[27] T. S. Han and K. Kobayashi, "A new achievable rate region for the interference channel," *IEEE Transactions on Information Theory*, vol. 27, no. 1, pp. 49–61, Jan. 1981.

[28] E. Telatar and D. Tse, "Bounds on the capacity region of a class of interference channels," in *Proc. of the 2007 International Symposium on Information Theory*, Nice, France, June 2007, pp. 2871–2874.

[29] X. Shang, G. Kramer, and B. Chen, "A new outer bound and the noisy-interference sum-rate capacity for the Gaussian interference channels," *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 689–699, Feb. 2009.

[30] V. Annapureddy and V. Veeravalli, "Gaussian interference networks: Sum capacity in the low-interference regime and new outer bounds on the capacity region," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3032–3050, July 2009.

[31] A. S. Motahari and A. K. Khandani, "Capacity bounds for the Gaussian interference channel," *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 620–643, Feb. 2009.

[32] A. El Gamal and M. H. M. Costa, "The capacity region of a class of deterministic interference channels," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 343–346, Mar. 1982.

[33] G. Bresler and D. N. C. Tse, "The two-user Gaussian interference channel: a deterministic view," *European Transactions on Telecommunications*, vol. 19, no. 4, pp. 333–354, June 2008.

[34] V. Cadambe and S. Jafar, "Interference alignment and degrees of freedom of the K-user interference channel," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3425–3441, Aug. 2008.

[35] V. Cadambe, S. Jafar, and S. Shamai, "Interference alignment on the deterministic channel and application to fully connected Gaussian interference networks," *IEEE Transactions on Information Theory*, vol. 55, no. 1, pp 269–274, Jan. 2009.

[36] S. A. Jafar and S. Vishwanath, "Generalized degrees of freedom of the symmetric Gaussian K-user interference channel," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3297–3303, July 2010.

[37] R. Milner, "An algebraic definition of simulation between programs," in *Proceedings of the IJCAI*, Sept. 1971, pp. 481–489.

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

[39] J. A. Thomas, "Feedback can at most double Gaussian multiple access channel capacity," *IEEE Transactions on Information Theory*, vol. 33, no. 5, pp. 711–716, Sept. 1987.

[40] A. S. Avestimehr, S. N. Diggavi, and D. N. C. Tse, "Approximate capacity of Gaussian relay networks," in *Proc. of IEEE ISIT 2008*, Toronto, Canada, July 2008, pp. 474–478.

AUTHOR'S BIOGRAPHY

Anand Muralidhar obtained his B.E. in Electronics & Communication Engineering from Visvesvaraya Technological University, Belgaum, in 2003, and M.Sc. (Engg.) in Electrical Communication Engineering at Indian Institute of Science, Bangalore, in 2007. He is currently a Ph.D. candidate in the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign.

Previously, he was an engineer in the voiceband modem team at Ittiam Systems, Bangalore, from June 2003 - May 2004. He was an intern in the systems team at Qualcomm Flarion Tech in summer 2010, where he worked on design of peer-to-peer systems over licensed spectrum. He is a student member of the IEEE.

His interests are broadly in wireless systems, networks, and stochastic control.