Feedback and Interference Alignment in Networks

Changho Suh

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2011-107
http://www.eecs.berkeley.edu/Pubs/TechRpts/2011/EECS-2011-107.html

September 28, 2011

Feedback and Interference Alignment in Networks

by

Changho Suh

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor David Tse, Chair
Professor Kannan Ramchandran
Professor Jim Pitman
Professor Gerhard Kramer

Fall 2011

Feedback and Interference Alignment in Networks

Abstract

Feedback and Interference Alignment in Networks

by

Changho Suh

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor David Tse, Chair

The increasing complexity of communication networks in size and density provides us enormous opportunities to exploit interaction among multiple nodes, thus enabling higher date rate of data streams. On the flip side, however, this complexity comes with challenges in managing interference that multiple source-destination pairs in the network may cause to each other. In this dissertation, we make progress on how we exploit the opportunities, as well as how we overcome the challenges.

In the first part, we find that *feedback* - one of the common ways to enable interaction in networks - has a promising role in improving the capacity performance of networks. Earlier results on feedback capacity were somewhat discouraging. This is mainly due to Shannon's original result on feedback capacity where he showed that in point-to-point communication, feedback does not increase capacity. Hence, traditionally it is believed that feedback has had little impact on increasing capacity of communication links. Therefore, the use of feedback has been limited to improving the reliability of communications, usually in the form of ARQ. In this dissertation, we show that in stark contrast to the point-to-point case, feedback can improve the capacity of *interference-limited* network. In fact, the improvement can be unbounded. This result shows that feedback can have a potentially significant role to play in mitigating interference. Also in the process of deriving this conclusion, we characterize the feedback capacity of the two-user Gaussian interference channel to within 2 bits, one of the longstanding open problems in network information theory.

In the second part, we propose a new interference management technique for widely deployed cellular networks. Inspired by a recent breakthrough, the concept of interference alignment, we develop an interference alignment technique for cellular networks. Our technique promises almost interference-free communication with the increase of the number of clients in cellular networks. It shows substantial gain (around 30% to 60%) as compared to one of the interference management techniques in current cellular systems. In addition, it comes with implementation benefits: it can actually be implemented with small changes to emerging 4G cellular standards and architectures at the base-stations and clients. In par-

ticular, the required signal-processing circuitry, software control, and channel-state feedback mechanisms are extensions of existing implementations and standards.

Lastly, we extend the interference alignment principle, developed in the context of wireless networks, into other fields of network research such as storage networks. In an effort to protect information against node failures, storage networks employ coding techniques, such as maximum distance separable (MDS) erasure codes, known as optimal codes in reliability with respect to redundancy. However, these MDS codes come with prohibitive maintenance cost when it comes to repairing failed storage nodes. While only partial information stored in the failed node needs to be recovered, the conventional MDS codes focus on the complete data recovery (including unwanted data, corresponding to *interference*) by downloading too much information from survivor storage encoded nodes, thus causing the high repair cost. Building on the connection between wireless and wireline networks, we leverage the interference alignment principle to develop a new class of MDS codes that significantly reduces the repair cost over the conventional MDS codes and also achieves information-theoretic optimal bound on the repair cost for all admissible code parameters.

To my wife,

Yuni,

for her dedicated support.

Contents

List of Figures

Acknowledgments

I was very fortunate to have Prof. David Tse as my advisor during my Ph.D. study. I can definitely say that he has been (and probably will have been) the most influential mentor in my entire research life. His unique way of doing research has formed the basis of my research philosophy and style, and will entirely affect my future research career. Among plenty of valuable advices that I have received from countless one-to-one meetings with him, the following way of doing research has mostly influenced my research methodology: (1) Start with real engineering problems; (2) Focus on the simplest scenario to understand a phenomenon; (3) Challenge conventional wisdom from first-principle thinking; (4) Try to understand how the result is fitted into the broader context. Not only did this methodology form inspiration of my research philosophy as a principle, but it has been deeply absorbed into mine through actual practices that come with intimate and frequent interactions with him. In fact, his insightful advices led me to write a research diary, thus keeping those in my mind.

What I learned from him is not just the way of conducting research, but his insights, inspiration, and especially his dedication to his students. Every insight he provided me has shed lights into overcoming the difficulties that I faced when solving some challenging problems. He also dedicated himself to advising me; he spent a huge amount of time in providing detailed guidelines as well as giving encouragement, approximately more than 500 hours in total during the past four years. I will never forget his dedication and I believe this will lead me to return this devotion to my future students when I have some.

I was also very lucky to have the great opportunity to collaborate with Prof. Kannan Ramchandran who also affected my research style significantly. Due to his endless requests for simple examples capturing the essence of problems, I had to struggle in developing those examples with an insightful interpretation. In fact, this took me spend much more time rather than solving the problem itself. In addition, he has advised me on how to write a good paper. I can never forget the last day, December 31, of 2009, when he spent the whole day (around 10 hours) to provide detailed guidelines on how to write a smooth and convincing introduction. In doing this, he showed me his detailed thought process which leads to a smooth and logical flow of the paper; so from this rewarding experience, I learned a lot how to form a logical flow and make presentation clear. I also had the rewarding experience to collaborate with Minnie Ho at Intel. For the past three-year collaboration on cellular networks, I could learn from her how theory can be transferred to practice, as well as how to optimize many of the system parameters while considering many of the challenging system aspects.

I would also like to give thanks to my dissertation committee members, Prof. Gerhard Kramer and Prof. Jim Pitman, for reading my dissertation as well as providing the fruitful comments and advices. Especially Prof. Gerhard Kramer's criticism on the feedback result led me to ponder on our result seriously, eventually improving the quality of the work. I would also like to appreciate fruitful discussions and valuable comments from my collaborators,

Alireza Vahid and Prof. Salman Avestimehr at Cornell; Dimitris Papailiopoulos and Prof. Alex Dimakis at USC; Naveen Goela at UC-Berkeley; and Prof. Michael Gastpar at EPFL. Also I am grateful to my colleagues, I-Hsiang Wang, Guy Bresler, Baosen Zhang, Sudeep Kamath, Venkatesan Ekambaram, Nebojsa Milosavljevic, Se Yong Park, Hao Zhang, Kristen Woyach, Dapo Omidiran, Sahand Negahban, and many others in Wireless Foundations. Especially I want to give special thanks to I-Hsiang and Venkatesan for their criticism on my presentation as well as on my writing. I also appreciate the valuable comments and advices from many other professors, including Prof. Anant Sahai and Prof. Bernd Sturmfels at UC-Berkeley, Prof. Pramod Viswanath at UIUC, and Prof. P. Vijay Kumar at IISc.

Lastly I would like to express my most heartfelt gratitude to my wife, Yuni, for her dedicated support. During Ph.D. study, we have shared everything from joy to sorrow. Whenever I had difficulties, she was always beside me, encouraging me. Whenever I was debating between important decisions, she was giving me the best advice as a counselor. In fact, I must say that half (maybe more than half) of all of my achievements so far are due to her dedicated support. Without her devoted help, I would have never finished this long journey of Ph.D. study.

Chapter 1

Introduction

Claude Shannon's information theory has made a great impact on the design of point-to-point communication. One of the major thrusts of information theory in the past 40 years has been to extend this theory to the network setting. In the general problem setup, each node in the network wants to transmit observations from one or more sources to one or more destination nodes in the network and we would like to characterize what is the best achievable end-to-end performance. Developing such a complete theory will have significant ramifications on how we architecture tomorrow's communication networks of increasing size and complexity. Although there has been success for certain network settings such as the many-to-one multiple access channels and the one-to-many degraded broadcast channels, many fundamental questions dealing with multiple sources and multiple destination nodes have remained unanswered for decades. For example, what is the fundamental role of *interaction* between multiple nodes that may frequently occur in complex network settings to improve the network performance? How should multiple links code their information to efficiently coexist despite of the *interference* they cause to each other? We are still lacking in our understanding and far away from reaching the holy grail of network information theory.

Some of our results in this field have made progress on answering these two fundamental questions by showing the significant role of interaction and developing interference management in certain interesting network settings. Going beyond answering these intriguing theoretical questions, we have also contributed to transferring the theories into real world working systems by developing an interference management scheme for cellular networks that can be implemented while approaching the theoretical limitations. Furthermore, we have found that the principles of interference management can be extended to address some of the significant open problems in other fields of network research such as storage networks. Especially, this interdisciplinary research experience has taught us that going backward and forward across disciplines may lead us to address many of the interconnected open problems, thus shedding light on many of the promising yet challenging fields of network research.

1.1 Role of Feedback

In communication networks, interaction is enabled through the use of feedback. Traditionally, it is believed that feedback has had little impact on increasing capacity. This is mainly due to Shannon's original result on feedback capacity, where he showed that feedback cannot increase the capacity in point-to-point communication links. Hence the use of feedback has been so far limited to improving the reliability of communication, usually in the form of ARQ. However, we recently found a promising role of feedback in networks [62]. What we have shown is that when there are two interfering point-to-point links, not only can feedback increase capacity of each link, but it can in fact provide an unbounded increase in capacity. This result promises a potentially huge gain of interaction in more general networks, thus driving a new paradigm shift on the use of feedback. Also in the process of deriving this conclusion, we characterize the feedback capacity of the two-user Gaussian interference channel to within 2 bits, an open problem for more than 30 years, thus shedding light on addressing many of the related open problems in network information theory.

Furthermore, we develop two enlightening interpretations to provide qualitative insights as to where the feedback gain comes from: (1) resource hole interpretation; (2) side information interpretation. The first interpretation says that the feedback gain comes from using feedback to maximize resource utilization, thereby enabling more efficient resource sharing between the interfering links. This will lead us to clearly understand how feedback helps multiple interfering links to efficiently coexist with each other. The second interpretation is that feedback enables destination nodes to exploit their received signals as *side information* to increase the non-feedback capacity. With this interpretation, we make a connection between our feedback problem and many other interesting problems in network information theory.

1.2 Interference Alignment

Cellular Networks: A recent breakthrough in addressing how to deal with interference is the concept of interference alignment: by aligning multiple interferers in signal space, their aggregate footprint can be vastly reduced and the overall performance improved. Deep understanding on this idea has recently led us to develop a new interference alignment technique for cellular networks [60, 57]. Our technique promises almost interference-free communication with the increase of the number of clients in cellular networks. It shows substantial gain (around 30% to 60%) as compared to one of the interference management techniques in current cellular systems. In addition, it comes with implementation benefits: it can actually be implemented with small changes to emerging 4G cellular standards and architectures at the base-stations and clients. In particular, the required signal-processing circuitry, software control, and channel-state feedback mechanisms are extensions of existing implementations and standards. The proposed technique is currently being discussed for

inclusion to 4G standards such as 3GPP-LTE and WiMAX.

One observation on this result [60, 57] is that our interference alignment scheme can provide huge gain especially when interference from the dominant interferer (the nearby base-station) is much stronger than residual interference from many other base-stations. This naturally leads us to believe that our scheme has great potential to heterogeneous networks that merge a multitude of wireless networks, such as femto-cells, pico-cells, relays and Wi-Fi networks, into macro cellular networks. Notice, for example, that in macro-pico cellular networks, a user connected to a pico base-station may see significant interference from the nearby macro base-station.

Distributed Storage Networks: We extend the interference alignment principle, developed in the context of wireless networks, to other networks such as storage network, to find the interdisciplinary nature of this principle. Connecting wireless networks to distributed storage networks, we address one of the significant problems in storage networks: the storage repair problem [59, 58, 22]. In an effort to protect information against node failures, storage networks employ coding techniques, such as maximum distance separable (MDS) erasure codes, known as optimal codes in reliability with respect to redundancy. However, these MDS codes come with prohibitive maintenance cost when it comes to repairing failed storage nodes. While only partial information stored in the failed node needs to be recovered, the conventional MDS codes focus on the complete data recovery (including unwanted data, corresponding to *interference*) by downloading too much information from survivor storage encoded nodes, thus causing the high repair cost. Now a question is: how much *communication* is necessary to decode only a desired subset of the entire information? This motivates us to ponder over this problem from a communication perspective. We draw parallels between wireless and storage networks to find a striking connection centered on the interference alignment principle: specifically, we map wireless channels to the coefficients of the MDS storage codes. Building on this connection, we leverage the interference alignment principle to develop a new class of MDS codes that significantly reduces the repair cost over the conventional MDS codes and also achieves information-theoretic optimal bound on the repair cost for all admissible code parameters.

1.3 Dissertation Outline

The rest of the dissertation is divided into three major parts. In Chapter 2, we will show that feedback has a great impact on improving the capacity of the two-user Gaussian interference channel (IC). To show this, we characterize the capacity region to within 2 bits/s/Hz and the symmetric capacity to within 1 bit/s/Hz for the two-user Gaussian IC with feedback. We develop achievable schemes and derive a new outer bound to arrive at this conclusion. The result makes use of a deterministic model to provide insights into the Gaussian channel. This deterministic model is a special case of El Gamal-Costa deterministic model and as a side-generalization, we establish the exact feedback capacity region of this general class of

deterministic ICs.

In Chapter 3, we develop an interference alignment (IA) technique for a cellular system, both uplink and downlink. This scheme can provide substantial gain especially when interference from a dominant interferer is significantly stronger than the remaining interference. We also propose another IA scheme, which we call *subspace IA*, in an attempt to mitigate the interference from multiple dominant interferers. We show that under some channel conditions, our subspace IA scheme can asymptotically achieve the interference-free capacity performance with an increase in the number of clients in each cell. We also remark that our subspace IA scheme can be well exploited to address the failed storage-node repair problem.

In Chapter 4, we show the great potential to extend the interference alignment to other network research such as storage networks. Drawing parallels between wireless and wireline networks, we exploit the IA idea to develop a new class of MDS erasure codes that significantly reduce the repair cost over the conventional MDS codes. Specifically, we address (n, k, d) Exact-Repair MDS codes, which allow for any failed node to be repaired exactly with access to arbitrary d survivor nodes, where $k \leq d \leq n-1$. We construct minimum repair-bandwidth Exact-Repair MDS codes for the case of $k/n \leq 1/2$ and $d \geq 2k-1$. We also show the existence of optimal Exact-Repair codes for the entire admissible range of possible (n, k, d), i.e., $k < n$ and $k \leq d \leq n-1$. Finally we conclude the dissertation in Chapter 5.

Chapter 2

Feedback in the Gaussian Interference Channel

2.1 Introduction

Shannon showed that feedback does not increase capacity of memoryless point-to-point channels [55]. On the other hand, feedback can indeed increase capacity in channels with memory such as colored Gaussian noise. However, the gain is *bounded*: feedback can provide a capacity increase of at most one bit [18, 39, 12]. In the multiple access channel (MAC), Gaarder and Wolf [26] showed that feedback could increase capacity even when the channel is memoryless. Inspired by this result, Ozarow [47] found the feedback capacity region for the two-user Gaussian MAC. Ozarow's result reveals that feedback gain is *bounded*. The reason for the bounded gain is that in the MAC, transmitters cooperation induced by feedback can at most boost signal power via aligning signal directions. Boosting signal power provides a capacity increase of a constant number of bits.

In the MAC, the receiver decodes the messages of *all* users. A natural question is to ask whether feedback can provide more significant gain in channels where a receiver wants to decode only desired message in the presence of interference. To answer this question, we focus on the simple two-user Gaussian interference channel (IC) where each receiver wants to decode the message only from its corresponding transmitter. We first make progress on the symmetric capacity. Gaining insights from a deterministic model [7] and the Alamouti scheme [6], we develop a simple two-staged achievable scheme. We then derive a new outer bound to show that the proposed scheme achieves the symmetric capacity to within one bit for all values of the channel parameters.

An interesting consequence of this result is that feedback can provide *multiplicative* gain in interference channels at high SNR. This can be shown from the generalized degrees-of-

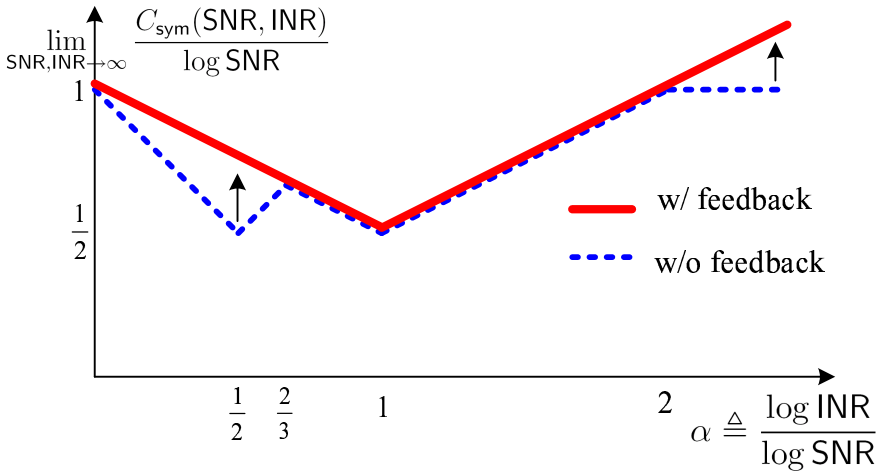


Figure 2.1: The generalized degrees-of-freedom of the Gaussian interference channel (IC) with feedback. For certain weak interference regimes ($0 \leq \alpha \leq \frac{2}{3}$) and for the very strong interference regime ($\alpha \geq 2$), the gap between the non-feedback and the feedback capacity becomes arbitrarily large as SNR and INR go to infinity. This implies that feedback can provide unbounded gain.

freedom in Fig. 2.1. The notion was defined in [25] as

$$d(\alpha) \triangleq \lim_{\mathsf{SNR},\mathsf{INR}\to\infty} \frac{C_{\mathsf{sym}}(\mathsf{SNR},\mathsf{INR})}{\log \mathsf{SNR}}, \tag{2.1}$$

where $C_{\mathsf{sym}}(\mathsf{SNR},\mathsf{INR}) = \sup\{R : (R,R) \in \mathcal{C}\}$ and $\mathcal{C}$ is the capacity region. In the figure, α (x-axis) indicates the ratio of INR to SNR in dB scale: $\alpha \triangleq \frac{\log \mathsf{INR}}{\log \mathsf{SNR}}$. Notice that in certain weak interference regimes ($0 \leq \alpha \leq \frac{2}{3}$) and in the very strong interference regime ($\alpha \geq 2$), feedback gain becomes arbitrarily large as SNR and INR go to infinity. For instance, when $\alpha = \frac{1}{2}$, the gap between the non-feedback and the feedback capacity becomes unbounded with the increase of SNR and INR, i.e.,

$$C_{\mathsf{sym}}^{\mathsf{FB}} - C_{\mathsf{sym}}^{\mathsf{NO}} \longrightarrow \frac{1}{4} \log \mathsf{SNR} \longrightarrow \infty. \tag{2.2}$$

Observing the ratio of the feedback to the non-feedback capacity in the high SNR regime, one can see that feedback provides *multiplicative* gain (50% gain for $\alpha = \frac{1}{2}$): $\frac{C_{\mathsf{sym}}^{\mathsf{FB}}}{C_{\mathsf{sym}}^{\mathsf{NO}}} \to 1.5$.

Moreover, we generalize the result to characterize the feedback capacity region to within 2 bits per user for all values of the channel parameters. Unlike the symmetric case, we develop an infinite-staged achievable scheme that employs three techniques: (i) block Markov encoding [16, 17]; (ii) backward decoding [69]; and (iii) Han-Kobayashi message splitting [32]. This result shows interesting contrast with the non-feedback capacity result. In the non-feedback case, it has been shown that the inner and outer bounds [32, 25] that guarantee a

1 bit gap to the optimality are described by five types of inequalities including the bounds for $R_1 + 2R_2$ and $2R_1 + R_2$. On the other hand, our result shows that the feedback capacity region approximated to within 2 bits requires only three types of inequalities without the $R_1 + 2R_2$ and $2R_1 + R_2$ bounds.

We also develop two interpretations to provide qualitative insights as to where feedback gain comes from. The first interpretation, which we call *resource hole interpretation*, says that the gain comes from using feedback to maximize resource utilization, thereby enabling more efficient resource sharing between the interfering users. The second interpretation is that feedback enables receivers to exploit their received signals as *side information* to increase the non-feedback capacity. With this interpretation, we make a connection between our feedback problem and many other interesting problems in network information theory.

Our results make use of a deterministic model [7] to provide insights into the Gaussian channel. This deterministic model is a special case of the El Gamal-Costa model [24]. As a side-generalization, we establish the exact feedback capacity region of this general class of deterministic ICs. From this result, one can infer an approximate feedback capacity region of two-user Gaussian MIMO ICs, as Teletar and Tse [64] did in the non-feedback case.

Interference channels with feedback have received previous attention [41, 42, 27, 63, 36]. Kramer [41, 42] developed a feedback strategy in the Gaussian IC; Kramer-Gastpar [27] and Tandon-Ulukus [63] derived outer bounds. However, the gap between the inner and outer bounds becomes arbitrarily large with the increase of SNR and INR. Although Kramer's scheme can be arbitrarily far from optimality, a careful analysis reveals that it can also provide multiplicative feedback gain. See Fig. 2.14. Jiang-Xin-Garg [36] found an achievable region in the discrete memoryless IC with feedback, based on block Markov encoding [16] and binning. However, their scheme involves three auxiliary random variables and therefore requires further optimization. Also no outer bounds are provided. We propose explicit achievable schemes and derive a new tighter outer bound to characterize the capacity region to within 2 bits and the symmetric capacity to within 1 bit universally. Subsequent to our work, Prabhakaran and Viswanath [51] have found an interesting connection between our feedback problem and the conferencing encoder problem. Making such a connection, they have independently characterized the sum feedback capacity to within 19 bits/s/Hz.

2.2 Model

Fig. 2.2 describes the two-user Gaussian IC with feedback where each transmitter gets delayed channel-output feedback only from its own receiver. Without loss of generality, we normalize signal power and noise power to 1, i.e., $P_k = 1$, $Z_k \sim \mathcal{CN}(0, 1)$, $\forall k = 1, 2$. Hence, the signal-to-noise ratio and the interference-to-noise ratio can be defined to capture the

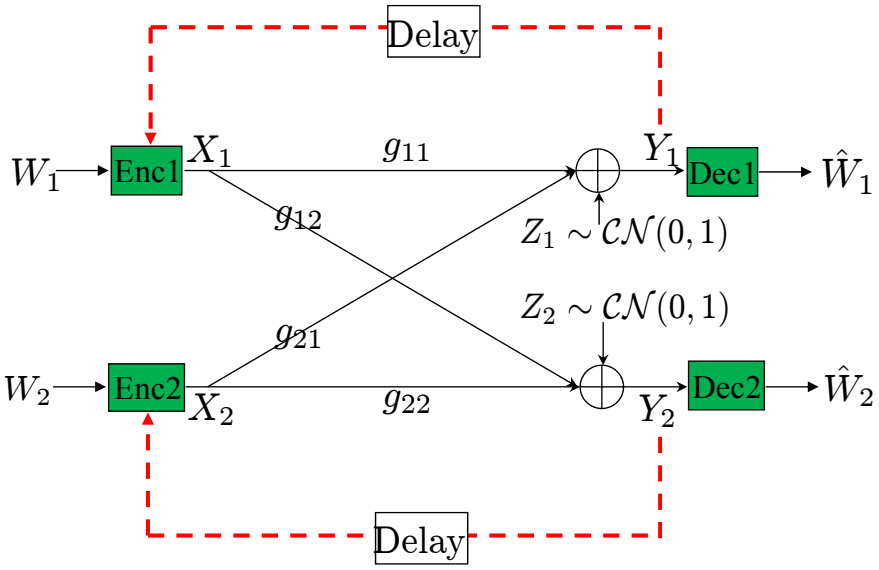


Figure 2.2: The Gaussian IC with feedback.

channel gains:

$$\begin{aligned} \mathsf{SNR}_1 &\triangleq |g_{11}|^2,\ \mathsf{SNR}_2 \triangleq |g_{22}|^2, \\ \mathsf{INR}_{12} &\triangleq |g_{12}|^2,\ \mathsf{INR}_{21} \triangleq |g_{21}|^2. \end{aligned} \tag{2.3}$$

There are two independent and uniformly distributed messages, $W_k \in \{1, 2, \cdots, m_k\}, \forall k = 1, 2$. Due to the delayed feedback, the encoded signal X_{ki} of user k at time i is a function of its own message and past output sequences:

$$X_{ki} = f_k^i\left(W_k, Y_{k1}, \cdots, Y_{k(i-1)}\right) = f_k^i\left(W_k, Y_k^{i-1}\right), \tag{2.4}$$

where we use shorthand notation Y_k^{i-1} to indicate the sequence up to $i-1$. A rate pair (R_1, R_2) is achievable if there exists a family of codebook pairs with codewords (satisfying power constraints) and decoding functions such that the average decoding error probabilities go to zero as code length N goes to infinity. The capacity region $\mathcal{C}$ is the closure of the set of the achievable rate pairs.

2.3 Symmetric Capacity to Within One Bit

We start with the symmetric channel setting where $|g_{11}| = |g_{22}| = |g_d|$ and $|g_{12}| = |g_{21}| = |g_c|$:

$$\mathsf{SNR} \triangleq \mathsf{SNR}_1 = \mathsf{SNR}_2,\ \mathsf{INR} \triangleq \mathsf{INR}_{12} = \mathsf{INR}_{21}. \tag{2.5}$$

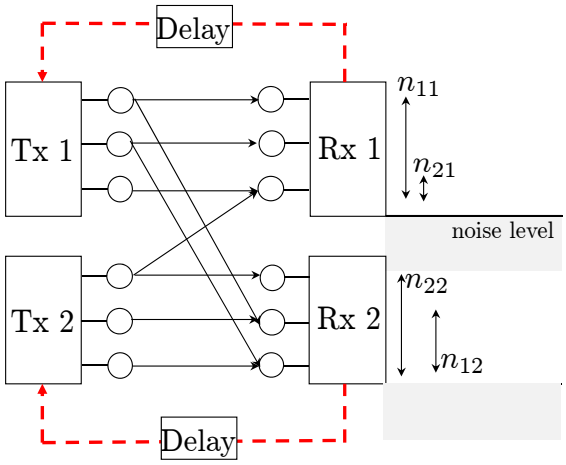


Figure 2.3: The deterministic IC with feedback.

Not only is this symmetric case simple, it also provides the key ingredients to both achievable scheme and outer bound needed for the characterization of the capacity region. Furthermore, this case provides enough qualitative insights as to where feedback gain comes from. Hence, we first focus on the symmetric channel.

Theorem 1. *We can achieve a symmetric rate of*

$$R_{\mathsf{sym}} = \max\left\{\frac{1}{2}\log\left(1+\mathsf{INR}\right), \frac{1}{2}\log\left(\frac{(1+\mathsf{SNR}+\mathsf{INR})^2 - \frac{\mathsf{SNR}}{1+\mathsf{INR}}}{1+2\mathsf{INR}}\right)\right\}. \tag{2.6}$$

The symmetric capacity is upper-bounded by

$$\overline{C}_{\mathsf{sym}} = \frac{1}{2}\sup_{0\le\rho\le 1}\left[\log\left(1+\frac{(1-\rho^2)\mathsf{SNR}}{1+(1-\rho^2)\mathsf{INR}}\right)+\log\left(1+\mathsf{SNR}+\mathsf{INR}+2\rho\sqrt{\mathsf{SNR}\cdot\mathsf{INR}}\right)\right]. \tag{2.7}$$

For all channel parameters of SNR *and* INR*,*

$$\overline{C}_{\mathsf{sym}} - R_{\mathsf{sym}} \le 1. \tag{2.8}$$

Proof. See Sections 2.3.2 and 2.3.3. □

2.3.1 Symmetric Capacity of a Deterministic Model

Deterministic Model: As a stepping stone towards the Gaussian IC, we use an intermediate model: the linear deterministic model [7], illustrated in Fig. 2.3. This model is

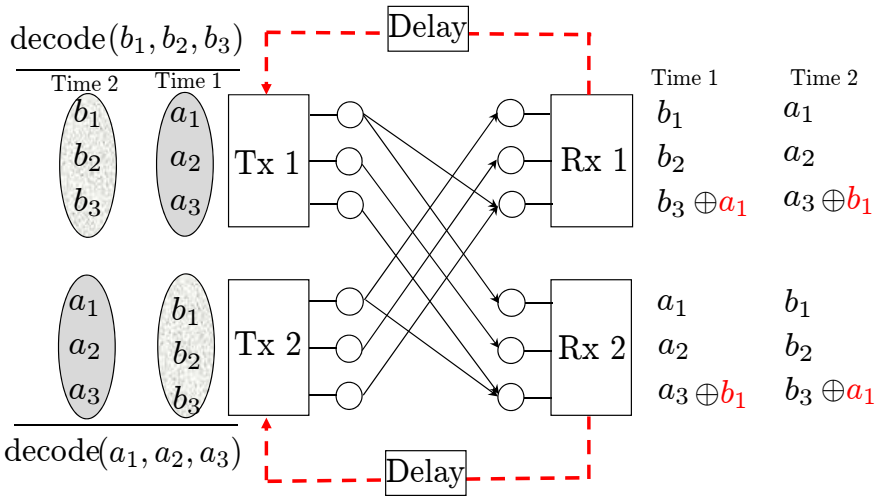


Figure 2.4: An achievable scheme for the deterministic IC: strong interference regime $\alpha := \frac{m}{n} = 3$.

useful in the non-feedback Gaussian IC: it was shown in [10] that the deterministic IC can approximate the Gaussian IC to within a constant number of bits irrespective of the channel parameter values. Our approach is to first develop insights from this model and then translate them to the Gaussian channel.

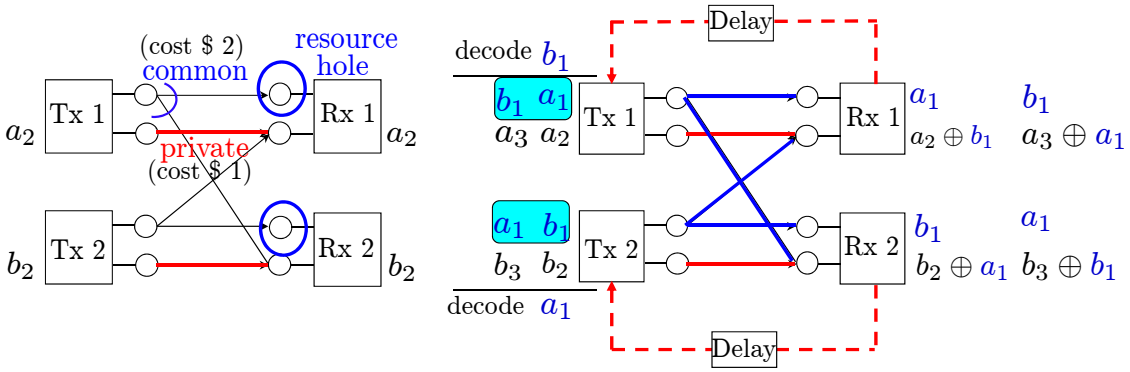
The connection with the Gaussian channel is as follows. The deterministic IC is characterized by four values: n_{11}, n_{12}, n_{21} and n_{22} where n_{ij} indicates the number of signal bit levels (or resource levels) from transmitter i to receiver j. These values correspond to the channel gains in dB scale, i.e., $\forall i \neq j$,

$$n_{ii} = \lfloor \log \mathsf{SNR}_i \rfloor, \; n_{ij} = \lfloor \log \mathsf{INR}_{ij} \rfloor. \tag{2.9}$$

In the symmetric channel, $n \triangleq n_{11} = n_{22}$ and $m \triangleq n_{12} = n_{21}$. Upper signal levels correspond to more significant bits and lower signal levels correspond to less significant bits of the received signal. A signal bit level observed by both the receivers above the noise level is broadcasted. If multiple signal levels arrive at the same signal level at a receiver, we assume a modulo-2-addition.

Achievable Scheme for the Strong Interference Regime ($m \geq n$): We explain the scheme through the simple example of $\alpha := \frac{m}{n} = 3$, illustrated in Fig. 2.4. Note that each receiver can see only one signal level from its corresponding transmitter. Therefore, in the non-feedback case, each transmitter can send only 1 bit through the top signal level. However, feedback can create a better alternative path, e.g., $[transmitter1 \rightarrow receiver2 \rightarrow feedback \rightarrow transmitter2 \rightarrow receiver1]$. This alternative path enables to increase the non-feedback rate.

The feedback scheme consists of two stages. In the first stage, transmitters 1 and 2 send independent binary symbols (a_1, a_2, a_3) and (b_1, b_2, b_3) respectively. Each receiver defers de-



(a) A non-feedback scheme (b) A feedback scheme

Figure 2.5: An achievable scheme for the weak interference regime, e.g., $\alpha = \frac{m}{n} = \frac{1}{2}$.

coding to the second stage. In the second stage, using feedback, each transmitter decodes information of the other user: transmitters 1 and 2 decode (b_1, b_2, b_3) and (a_1, a_2, a_3) respectively. Each transmitter then sends the other user's information. Each receiver gathers the received bits sent during the two stages: the six linearly independent equations containing the six unknown symbols. As a result, each receiver can solve the linear equations to decode its desired bits. Notice that the second stage was used for refining all the bits sent previously, without sending additional information. Therefore, the symmetric rate is $\frac{3}{2}$ in this example. Notice the 50% improvement from the non-feedback rate of 1. We can easily extend the scheme to arbitrary (n, m). In the first stage, each transmitter sends m bits using all the signal levels. Using two stages, these m bits can be decoded with the help of feedback. Thus, we can achieve:

$$R_{\mathsf{sym}} = \frac{m}{2}. \tag{2.10}$$

Remark 1. *The gain in the strong interference regime comes from the fact that feedback provides a better alternative path through the two cross links. The cross links relay the other user's information through feedback. We can also explain this gain using a resource hole interpretation. Notice that in the non-feedback case, each transmitter can send only 1 bit through the top level and therefore there is a resource hole (in the second level) at each receiver. However, with feedback, all of the resource levels at the two receivers can be filled up. Feedback maximizes resource utilization by providing a better alternative path. This concept coincides with correlation routing in [41].*

On the other hand, in the weak interference regime, there is no better alternative path, since the cross links are weaker than the direct links. Nevertheless, it turns out that feedback gain can also be obtained in this regime.

Achievable Scheme for the Weak Interference Regime ($m \leq n$): Let us start by examining the scheme in the non-feedback case. Unlike the strong interference regime, only

part of information is visible to the other receiver in the weak interference regime. Hence, information can be split into two parts [32]: common m bits (visible to the other receiver) and private $(n-m)$ bits (invisible to the other receiver). Notice that using common levels causes interference to the other receiver. Sending 1 bit through a common level consumes a total of 2 levels at the two receivers (say \$2), while using a private level costs only \$1. Because of this, a reasonable achievable scheme is to follow the two steps sequentially: (i) sending all of the cheap $(n-m)$ private bits on the lower levels; (ii) sending some number of common bits on the upper levels. The number of common bits is decided depending on m and n.

Consider the simple example of $\alpha = \frac{m}{n} = \frac{1}{2}$, illustrated in Fig. 2.5 (a). First transmitters 1 and 2 use the cheap private signal levels respectively. Once the bottom levels are used, however using the top levels is precluded due to a conflict with the private bits already sent, thus each transmitter can send only one bit.

Observe the two resource holes on the top levels at the two receivers. We find that feedback helps fill up all of these resource holes to improve the performance. The scheme uses two stages. As for the private levels, the same procedure is applied as that in the non-feedback case. How to use the common levels is key to the scheme. In the first stage, transmitters 1 and 2 send private bits a_2 and b_2 on the bottom levels respectively. Now transmitter 1 squeezes one more bit a_1 on its top level. While a_1 is received cleanly at receiver 1, it causes interference at receiver 2. Feedback can however resolve this conflict. In the second stage, with feedback transmitter 2 can decode the common bit a_1 of the other user. As for the bottom levels, transmitters 1 and 2 send new private bits a_3 and b_3 respectively. The idea now is that transmitter 2 sends the other user's common bit a_1 on its top level. This transmission allows receiver 2 to refine the corrupted bit b_2 from $b_2 \oplus a_1$ without causing interference to receiver 1, since receiver 1 already had the side information of a_1 from the previous broadcasting. We paid \$2 for the earlier transmission of a_1, but now we can get a *rebate* of \$1. Similarly, with feedback, transmitter 2 can squeeze one more bit b_1 on its top level without causing interference. Therefore, we can achieve the symmetric rate of $\frac{3}{2}$ in this example, i.e., the 50% improvement from the non-feedback rate of 1.

This scheme can be easily generalized to arbitrary (n, m). In the first stage, each transmitter sends m bits on the upper levels and $(n-m)$ bits on the lower levels. In the second stage, each transmitter forwards the m bits of the other user on the upper levels and sends new $(n-m)$ private bits on the lower levels. Then, each receiver can decode all of the n bits sent in the first stage and new $(n-m)$ private bits sent in the second stage. Therefore, we can achieve:

$$R_{\mathsf{sym}} = \frac{n + (n-m)}{2} = n - \frac{m}{2}. \tag{2.11}$$

Remark 2 (**Resource Hole Interpretation**). *Observe that all the resource levels are fully packed after applying the feedback scheme. Thus, feedback maximizes resource utilization to improve the performance significantly. We will discuss this interpretation in more details in*

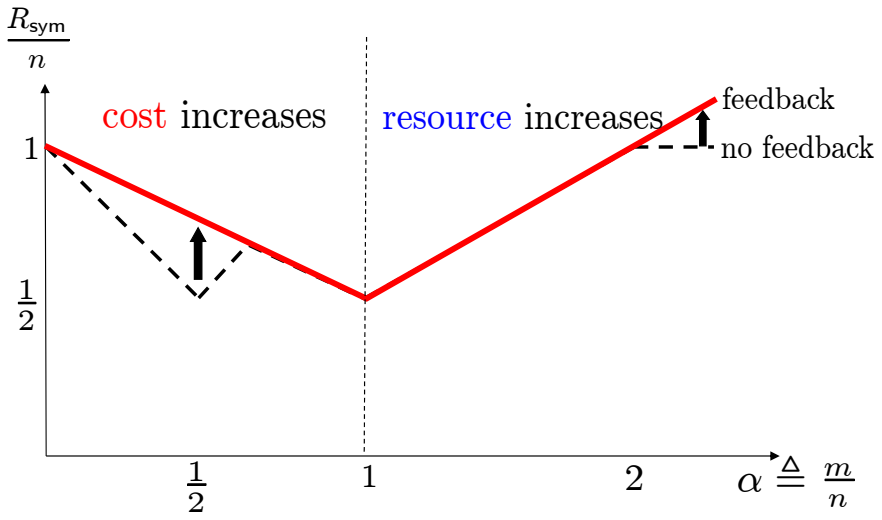


Figure 2.6: Symmetric feedback rate (2.10), (2.11) for the deterministic IC. Feedback maximizes resource utilization while it cannot reduce transmission costs. The "V" curve is obtained when all of the resource levels are fully packed with feedback. This shows the optimality of the feedback scheme.

Section 2.6.1.

We also develop another interpretation as to the role of feedback, which leads us to make an intimate connection to many other interesting problems in network information theory. We will discuss this connection later in Section 2.6.2.

Optimality of the Achievable Scheme: Now a natural question that arises is to ask the optimality of the scheme. Using the resource hole interpretation, we provide an intuitive explanation of the optimality. Later in Section 2.5, we will provide a rigorous proof.

From W to V Curve: Fig. 2.6 shows (i) the symmetric feedback rate (2.10), (2.11) of the achievable scheme (representing the "V" curve); (ii) the non-feedback capacity [10] (representing the "W" curve). Using the resource hole interpretation, we will provide intuition as to how we can go from the W curve to the V curve with feedback.

Observe that the total number of resource levels and transmission cost depend on (n, m). Specifically, suppose that the two senders employ the same transmission strategy to achieve the symmetric rate: using x private and y common levels. We then get:

$$\begin{aligned} &\#\text{ of resource levels at each receiver} = \max(n, m), \\ &\text{transmission cost} = 1 \times x + 2 \times y. \end{aligned} \tag{2.12}$$

Here notice that using a private level costs 1 level, while using a common level costs 2 levels. Now observe that for fixed n, as $\alpha = \frac{m}{n}$ grows: for $0 \leq \alpha \leq 1$, transmission cost increases; for $\alpha \geq 1$, the number of resource levels increases. Since all the resource levels are fully utilized

with feedback, this observation implies that with feedback the total number of transmission bits must decrease when $0 \leq \alpha \leq 1$ (inversely proportional to transmission cost) and must increase when $\alpha \geq 1$ (proportional to the number of resource levels). This is reflected in the V curve. In contrast, in the non-feedback case, for some range of α, resource levels are not fully utilized, as shown in the $\alpha = \frac{1}{2}$ example of Fig. 2.5 (a). This is reflected in the W curve.

Why We Cannot Go Beyond the V Curve: While feedback maximizes resource utilization to fill up all of the resource holes, *it cannot reduce transmission costs*. To see this, consider the example in Fig. 2.5 (b). Observe that even with feedback, a common bit still has to consume two levels at the two receivers. For example, the common bit a_1 needs to occupy the top level at receiver 1 in time 1; and the top level at receiver 2 in time 2. In time 1, while a_1 is received cleanly at receiver 1, it interferes with the private bit b_2. In order to refine b_2, receiver 2 needs to get a_1 cleanly and therefore needs to reserve one resource level for a_1. Thus, in order not to interfere with the private bit b_1, the common bit a_1 needs to consume a total of the two resource levels at the two receivers. As mentioned earlier, assuming that transmission cost is not reduced, a total number of transmission bits is reflected in the V curve. As a result, we cannot go beyond the "V" curve with feedback, showing the optimality of the achievable scheme. Later in Section 2.5, we will prove this rigorously.

Remark 3 (**Reminiscent of Shannon's Comment in [56]**). *The fact that feedback cannot reduce transmission costs reminds us of Shannon's closing comment in [56]: "We may have knowledge of the past and cannot control it; we may control the future but have no knowledge of it." This statement implies that feedback cannot control the past although it enables us to know the past; so this coincides with our finding that feedback cannot reduce transmission costs, as the costs already occurred in the past.*

2.3.2 An Achievable Scheme for the Gaussian IC

Let us go back to the Gaussian channel. We will translate the deterministic IC scheme to the Gaussian IC. Let us first consider the strong interference regime.

Strong Interference Regime ($\mathsf{INR} \geq \mathsf{SNR}$): The structure of the transmitted signals in Fig. 2.4 sheds some light on the Gaussian channel. Observe that in the second stage, each transmitter sends the other user's information sent in the first stage. This reminds us of the *Alamouti scheme* [6]. The beauty of the Alamouti scheme is that received signals can be designed to be orthogonal during two time slots, although the signals in the first time slot are sent without any coding. This was exploited and pointed out in distributed space-time codes [43]. With the Alamouti scheme, transmitters are able to encode their messages so that received signals are orthogonal. Orthogonality between the two different signals guarantees complete removal of the interfering signal.

In accordance with the deterministic IC example, the scheme uses two stages (or blocks). In the first stage, transmitters 1 and 2 send codewords X_1^N and X_2^N with rates R_1 and R_2

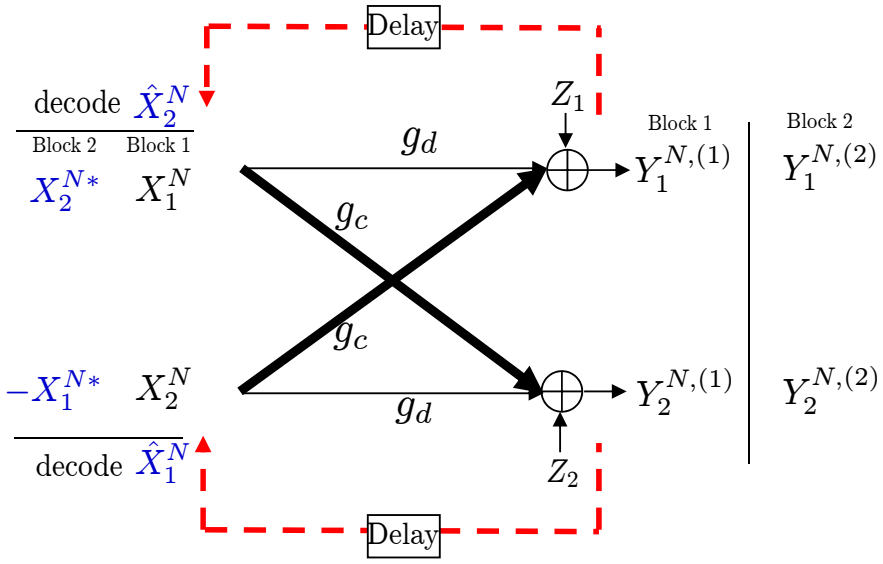


Figure 2.7: An Alamouti-based achievable scheme for the Gaussian IC: strong interference regime.

respectively. In the second stage, using feedback, transmitters 1 and 2 decode X_2^N and X_1^N respectively. This can be decoded if

$$R_1, R_2 \leq \frac{1}{2}\log\left(1+\mathsf{INR}\right) \quad \text{bits/s/Hz}. \tag{2.13}$$

We are now ready to apply the Alamouti scheme. Transmitters 1 and 2 send X_2^{N*} and $-X_1^{N*}$ respectively. Receiver 1 can then gather the two received signals: for $1 \leq i \leq N$,

$$\begin{bmatrix} Y_{1i}^{(1)} \\ Y_{1i}^{(2)*} \end{bmatrix} = \begin{bmatrix} g_d & g_c \\ -g_c^* & g_d^* \end{bmatrix} \begin{bmatrix} X_{1i} \\ X_{2i} \end{bmatrix} + \begin{bmatrix} Z_{1i}^{(1)} \\ Z_{1i}^{(2)*} \end{bmatrix}. \tag{2.14}$$

To extract X_{1i}, it multiplies the row vector orthogonal to the vector associated with X_{2i} and therefore we get:

$$\begin{bmatrix} g_d^* & -g_c \end{bmatrix} \begin{bmatrix} Y_{1i}^{(1)} \\ Y_{1i}^{(2)*} \end{bmatrix} = (|g_d|^2 + |g_c|^2)X_{1i} + g_d^* Z_{2i}^{(1)} - g_c Z_{1i}^{(2)*}. \tag{2.15}$$

The codeword X_1^N can be decoded if

$$R_1 \leq \frac{1}{2}\log\left(1+\mathsf{SNR}+\mathsf{INR}\right) \quad \text{bits/s/Hz}. \tag{2.16}$$

Similar operations are done at receiver 2. Since (2.16) is implied by (2.13), we get the desired result: the left term in (2.6).

Weak Interference Regime ($\mathsf{INR} \leq \mathsf{SNR}$)**:** Unlike the strong interference regime, in the weak interference regime, there are two types of information: common and private information. A natural idea is to apply the Alamouti scheme only for common information and newly add private information. It was shown in [61] that this scheme can approximate the symmetric capacity to within ≈ 1.7 bits/s/Hz. However, the scheme can be improved to reduce the gap further. Unlike the deterministic IC, in the Gaussian IC, private signals have some effects, i.e., these private signals cannot be completely ignored. Notice that the scheme includes *decode-and-forward* operation at the transmitters after receiving the feedback. And so when each transmitter decodes the other user's common message while treating the other user's private signals as noise, private signals can incur performance loss.

This can be avoided by instead performing *amplify-and-forward*: with feedback, the transmitters get the interference plus noise and then forward it subject to the power constraints. This transmission allows each receiver to refine its corrupted signal sent in the previous time, without causing significant interference.[1] Importantly, notice that this scheme does not require message-splitting. Even without splitting messages, we can refine the corrupted signals (see Appendix 2.9.1 to understand this better). Therefore, there is no loss due to private signals.

Specifically, the scheme uses two stages. In the first stage, each transmitter k sends codeword X_k^N with rate R_k. In the second stage, with feedback transmitter 1 gets the interference plus noise:

$$S_2^N = g_c X_2^N + Z_1^{(1),N}. \tag{2.17}$$

Now the complex conjugate technique based on the Alamouti scheme is applied to make X_1^N and S_2^N well separable. Transmitters 1 and 2 send $\frac{S_2^{N*}}{\sqrt{1+\mathsf{INR}}}$ and $-\frac{S_1^{N*}}{\sqrt{1+\mathsf{INR}}}$ respectively, where $\sqrt{1+\mathsf{INR}}$ is a normalization factor to meet the power constraint. Under Gaussian input distribution, we can compute the rate under MMSE demodulation: $\frac{1}{2} I(X_{1i}; Y_{1i}^{(1)}, Y_{2i}^{(2)})$. Straightforward calculations give the desired result: the right term in (2.6). See Appendix 2.9.1 for detailed computations.

Remark 4. *As mentioned earlier, unlike the decode-and-forward scheme, the amplify-and-forward scheme does not require message-splitting, thereby removing the effect of private signals. This improves the performance to reduce the gap further.*

2.3.3 An Outer Bound and One-Bit Gap to the Symmetric Capacity

The symmetric rate upper bound is implied by the outer bound for the capacity region; we defer the proof to Theorem 3 in Section 2.4.2.

[1] In Appendix 2.9.1, we provide intuition behind this scheme.

Using the symmetric rate of (2.6) and the outer bound of (2.7), we get:

$$
\begin{aligned}
2(\bar{C}_{\mathsf{sym}} - R_{\mathsf{sym}}) &\overset{(a)}{\leq} \log\left(1+\frac{\mathsf{SNR}}{1+\mathsf{INR}}\right) + \log\left(1+\mathsf{SNR}+\mathsf{INR}+2\sqrt{\mathsf{SNR}\cdot\mathsf{INR}}\right) \\
&\quad - \log\left(\frac{(1+\mathsf{SNR}+\mathsf{INR})^2 - \frac{\mathsf{SNR}}{1+\mathsf{INR}}}{1+2\mathsf{INR}}\right) \\
&= \log\left(\frac{1+\mathsf{SNR}+\mathsf{INR}(1+\mathsf{SNR}+\mathsf{INR}+2\sqrt{\mathsf{SNR}\cdot\mathsf{INR}})}{1+\mathsf{INR}}\right) \\
&\quad + \log\left(\frac{(1+2\mathsf{INR})(1+\mathsf{INR})}{(1+\mathsf{SNR}+\mathsf{INR})^2(1+\mathsf{INR})-\mathsf{SNR}}\right) \\
&= \log\left(\frac{1+\mathsf{SNR}+\mathsf{INR}+2\sqrt{\mathsf{SNR}\cdot\mathsf{INR}}}{1+\mathsf{SNR}+\mathsf{INR}}\cdot\frac{1+2\mathsf{INR}}{1+\mathsf{INR}-\frac{\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2}}\right) \\
&\overset{(b)}{\leq} \log\left(2\cdot\frac{2\left(1+\mathsf{INR}-\frac{\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2}\right)-1+\frac{2\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2}}{1+\mathsf{INR}-\frac{\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2}}\right) \\
&= \log\left(2\cdot\left\{2-\left(\frac{1-\frac{2\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2}}{1+\mathsf{INR}-\frac{\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2}}\right)\right\}\right) \\
&\overset{(c)}{\leq} \log 4 = 2.
\end{aligned}
$$

Step (a) follows from choosing the trivial maximum value of the outer bound (2.7) and choosing a minimum value (the second term) of the lower bound (2.6). Note that the first and second terms in (2.7) are maximized when $\rho = 0$ and $\rho = 1$ respectively. Step (b) follows from $1+\mathsf{SNR}+\mathsf{INR}+2\sqrt{\mathsf{SNR}\cdot\mathsf{INR}} \leq 2(1+\mathsf{SNR}+\mathsf{INR})$; and (c) follows from $(1+\mathsf{SNR}+\mathsf{INR})^2 \geq 2\mathsf{SNR}$ and $\frac{\mathsf{SNR}}{(1+\mathsf{SNR}+\mathsf{INR})^2} \leq 1$.

Fig. 2.8 shows a numerical result for the gap between the inner and outer bounds. Notice that the gap is upper-bounded by exactly one bit. The worst-case gap occurs when $\mathsf{SNR} \approx \mathsf{INR}$ and these values go to infinity. Also note that in the strong interference regime, the gap approaches 0 with the increase of SNR and INR, while in the weak interference regime, the gap does not vanish. For example, when $\alpha = \frac{1}{2}$, the gap is around 0.5 bits.

Remark 5 (**Why does a 1-bit gap occur?**). *Observe in Figs. 2.7 and 2.16 that the transmitted signals of the two senders are uncorrelated in our scheme. The scheme completely loses power gain (also called beamforming gain). On the other hand, when deriving the outer bound of (2.7), we allow for arbitrary correlation between the transmitters. Thus, the 1-bit gap is based on the outer bound. In the actual system, correlation is in-between and therefore one can expect that the actual gap to the capacity is less than 1 bit.*

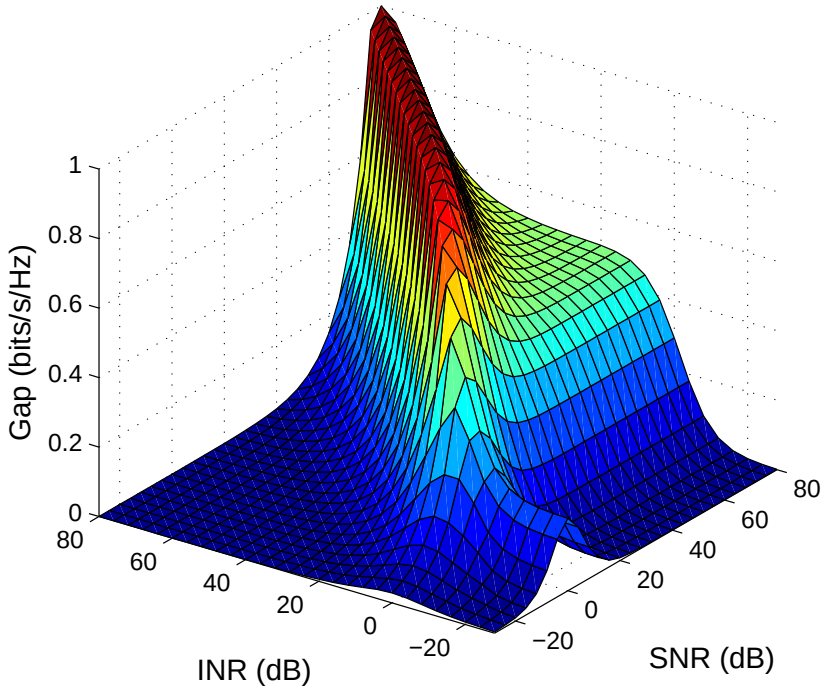


Figure 2.8: The gap between our inner and upper bounds. The gap is upper-bounded by exactly one bit. The worst-case gap occurs when $\mathsf{SNR} \approx \mathsf{INR}$ and these values go to infinity. In the strong interference regime, the gap vanishes with the increase of SNR and INR, while in the weak interference regime, the gap does not, e.g., the gap is around 0.5 bits for $\alpha = \frac{1}{2}$.

Beamforming gain is important only when SNR *and* INR *are quite close, i.e.,* $\alpha \approx 1$*. This is because when* $\alpha = 1$*, the interference channel is equivalent to the multiple access channel where the Ozarow scheme [47] and the Kramer scheme [41] (that capture beamforming gain) are optimal. In fact, the capacity theorem in [42] shows that the Kramer scheme is optimal for one specific case of* $\mathsf{INR} = \mathsf{SNR} - \sqrt{2 \cdot \mathsf{SNR}}$*, although it is arbitrarily far from optimality for the other cases. This observation implies that our proposed scheme can be improved further.*

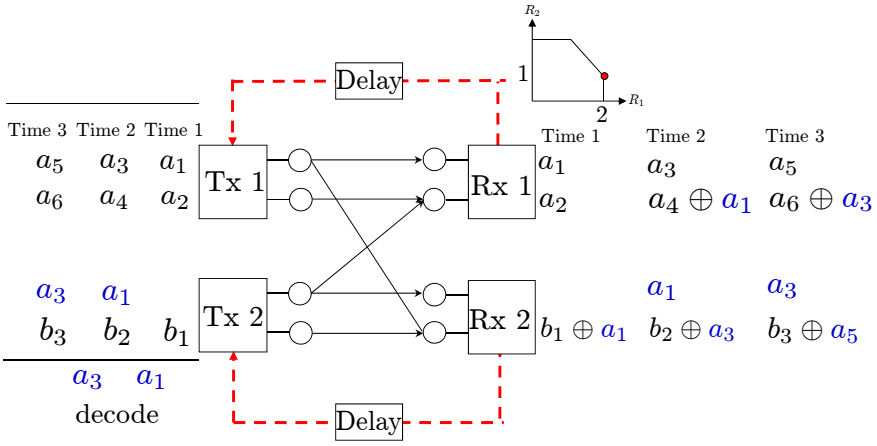


Figure 2.9: A deterministic IC example where an infinite number of stages need to be employed to achieve the rate pair of $(2, 1)$ with feedback. This example motivates us to use (1) block Markov encoding; and (2) Han-Kobayashi message splitting.

2.4 Capacity Region to Within 2 Bits

2.4.1 An Achievable Rate Region

We have developed an achievable scheme meant for the symmetric rate and provided a resource hole interpretation. For the case of the capacity region, we find that while this interpretation can also be useful, the two-staged scheme is not enough. A new achievable scheme needs to be developed for the region characterization.

To see this, let us consider a deterministic IC example in Fig. 2.9 where an infinite number of stages need to be employed to achieve a corner point of $(2, 1)$ with feedback. Observe that to guarantee $R_1 = 2$, transmitter 1 needs to send 2 bits every time slot. Once transmitter 1 sends (a_1, a_2), transmitter 2 cannot use its top level since the transmission causes interference to receiver 1. It can use only the bottom level to send information. This transmission however suffers from interference: receiver 2 gets the interfered signal $b_1 \oplus a_1$. We will show that this corrupted bit can be refined with feedback. In time 2, transmitter 2 can decode a_1 with feedback. In an effort to achieve the rate pair of $(2, 1)$, transmitter 1 sends (a_3, a_4) and transmitter 2 sends b_2 on the bottom level. Now apply the same idea used in the symmetric case: transmitter 2 sends the other user's information a_1 on the top level. This transmission allows receiver 2 to refine the corrupted signal b_1 without causing interference to receiver 1, since receiver 1 already had a_1 as *side information*. Notice that during the two time slots, receiver 1 can decode 4 bits (2 bits/time), while receiver 2 can decode 1 bits (0.5 bits/time). The point $(2, 1)$ is not achieved yet due to unavoidable loss occurred in time 1. This loss, however, can be amortized by iterating the same operation. As this example shows, the previous two-staged scheme needs to be modified so as to incorporate an infinite

number of stages.

Let us apply this idea to the Gaussian channel. The use of an infinite number of stages motivates the need for employing *block Markov encoding* [16, 17]. Similar to the symmetric case, we can now think of two possible schemes: (1) decode-and-forward (with message-splitting); and (2) amplify-and-forward (without message-splitting). As pointed out in Remark 4, in the Gaussian channel, private signals cannot be completely ignored, thereby incurring performance loss, thus the amplify-and-forward scheme without message-splitting has better performance. However, it requires heavy computations to compute the rate region, so we focus on the decode-and-forward scheme, although it induces a larger gap. As for a decoding operation, we employ backward decoding [69].

Here is the outline of our scheme. We employ block Markov encoding with a total size B of blocks. In block 1, each transmitter splits its own message into common and private parts and then sends a codeword superimposing the common and private messages. For power splitting, we adapt the idea of the simplified Han-Kobayashi scheme [25] where private power is set such that a private signal is seen below the noise level at the other receiver. In block 2, with feedback, each transmitter decodes the other user's common message (sent in block 1) while treating the other user's private signal as noise. Two common messages are then available at the transmitter: (1) its own message; and (2) the other user's message decoded with the help of feedback. Conditioned on these two common messages, each transmitter generates new common and private messages. It then sends the corresponding codeword. Each transmitter repeats this procedure until block $B-1$. In the last block B, to facilitate backward decoding, each transmitter sends the predetermined common message and a new private message. Each receiver waits until total B blocks have been received and then performs backward decoding. We will show that this scheme enables us to obtain an achievable rate region that approximates the capacity region.

Theorem 2. *The feedback capacity region includes the set $\mathcal{R}$ of (R_1, R_2) such that*

$$R_1 \le \log\left(1+\mathsf{SNR}_1+\mathsf{INR}_{21}+2\rho\sqrt{\mathsf{SNR}_1\cdot\mathsf{INR}_{21}}\right)-1 \tag{2.18}$$

$$R_1 \le \log\left(1+(1-\rho)\mathsf{INR}_{12}\right)+\log\left(2+\frac{\mathsf{SNR}_1}{\mathsf{INR}_{12}}\right)-2 \tag{2.19}$$

$$R_2 \le \log\left(1+\mathsf{SNR}_2+\mathsf{INR}_{12}+2\rho\sqrt{\mathsf{SNR}_2\cdot\mathsf{INR}_{12}}\right)-1 \tag{2.20}$$

$$R_2 \le \log\left(1+(1-\rho)\mathsf{INR}_{21}\right)+\log\left(2+\frac{\mathsf{SNR}_2}{\mathsf{INR}_{21}}\right)-2 \tag{2.21}$$

$$R_1+R_2 \le \log\left(2+\frac{\mathsf{SNR}_1}{\mathsf{INR}_{12}}\right)+\log\left(1+\mathsf{SNR}_2+\mathsf{INR}_{12}+2\rho\sqrt{\mathsf{SNR}_2\cdot\mathsf{INR}_{12}}\right)-2 \tag{2.22}$$

$$R_1+R_2 \le \log\left(2+\frac{\mathsf{SNR}_2}{\mathsf{INR}_{21}}\right)+\log\left(1+\mathsf{SNR}_1+\mathsf{INR}_{21}+2\rho\sqrt{\mathsf{SNR}_1\cdot\mathsf{INR}_{21}}\right)-2 \tag{2.23}$$

for $0 \le \rho \le 1$.

Proof. Our achievable scheme is generic, not limited to the Gaussian IC. We therefore characterize an achievable rate region for discrete memoryless ICs and then choose an appropriate joint distribution to obtain the desired result. In fact, this generic scheme can also be applied to El Gamal-Costa deterministic IC (to be described in Section 2.5).

Lemma 1. *The feedback capacity region of the two-user discrete memoryless IC includes the set of* (R_1, R_2) *such that*

$$R_1 \le I(U, U_2, X_1; Y_1) \tag{2.24}$$
$$R_1 \le I(U_1; Y_2|U, X_2) + I(X_1; Y_1|U_1, U_2, U) \tag{2.25}$$
$$R_2 \le I(U, U_1, X_2; Y_2) \tag{2.26}$$
$$R_2 \le I(U_2; Y_1|U, X_1) + I(X_2; Y_2|U_1, U_2, U) \tag{2.27}$$
$$R_1 + R_2 \le I(X_1; Y_1|U_1, U_2, U) + I(U, U_1, X_2; Y_2) \tag{2.28}$$
$$R_1 + R_2 \le I(X_2; Y_2|U_1, U_2, U) + I(U, U_2, X_1; Y_1), \tag{2.29}$$

over all joint distributions $p(u)p(u_1|u)p(u_2|u)p(x_1|u_1, u)p(x_2|u_2, u)$.

Proof. See Appendix 2.9.2. □

Now we will choose the following Gaussian input distribution to complete the proof: $\forall k = 1, 2$,

$$U \sim \mathcal{CN}(0, \rho); U_k \sim \mathcal{CN}(0, \lambda_{ck}); X_{pk} \sim \mathcal{CN}(0, \lambda_{pk}), \tag{2.30}$$

where $X_k = U + U_k + X_{kp}$; λ_{ck} and λ_{pk} indicate the powers allocated to the common and private message of transmitter k respectively; and (U, U_k, X_{kp})'s are independent. By symmetry, it suffices to prove (2.18), (2.19) and (2.22).

To prove (2.18), consider $I(U, U_2, X_1; Y_1) = h(Y_1) - h(Y_1|U, U_2, X_1)$. Note

$$|K_{Y_1|X_1,U_2,U}| = 1 + \lambda_{p2}\mathsf{INR}_{21}. \tag{2.31}$$

As mentioned earlier, for power splitting, we adapt the idea of the simplified Han-Kobayashi scheme [25]. We set private power such that the private signal appears below the noise level at the other receiver. This idea mimics that of the deterministic IC example where the private bit is below the noise level so that it is invisible. The remaining power is assigned to the common message. Specifically, we set:

$$\lambda_{p2} = \min\left(\frac{1}{\mathsf{INR}_{21}}, 1\right), \quad \lambda_{c2} = 1 - \lambda_{p2}, \tag{2.32}$$

This choice gives

$$I(U, U_2, X_1; Y_1) = \log\left(1 + \mathsf{SNR}_1 + \mathsf{INR}_{21} + 2\rho\sqrt{\mathsf{SNR}_1 \cdot \mathsf{INR}_{21}}\right) - 1, \tag{2.33}$$

which proves (2.18). With the same power setting, we can compute:

$$I(U_1; Y_2|U, X_2) = \log\left(1 + (1-\rho)\mathsf{INR}_{12}\right) - 1, \tag{2.34}$$

$$I(X_1; Y_1|U, U_1, U_2) = \log\left(2 + \frac{\mathsf{SNR}_1}{\mathsf{INR}_{12}}\right) - 1. \tag{2.35}$$

This proves (2.19). Lastly, by (2.33) and (2.35), we prove (2.22). □

Remark 6 (**Three Types of Inequalities**)**.** *In the non-feedback case, it is shown in [25] that an approximate capacity region is characterized by five types of inequalities including the bounds for* $2R_1 + R_2$ *and* $R_1 + 2R_2$*. In contrast, in the feedback case, our achievable rate region is described by only three types of inequalities.*[2] *In Section 2.6.1, we will provide qualitative insights as to why the* $2R_1 + R_2$ *bound is missing with feedback.*

Remark 7 (Connection to Related Work [66])**.** *Our achievable scheme is essentially the same as the scheme introduced by Tuninetti [66] in a sense that the three techniques (message-splitting, block Markov encoding and backward decoding) are jointly employed. Although the author in [66] considers a different context (the conferencing encoder problem), Prabhakaran and Viswanath [51] have made an interesting connection between the feedback problem and the conferencing encoder problem. See [51] for details. Despite this close connection, however, the scheme in [66] uses five auxiliary random variables and thus requires further optimization. On the other hand, we obtain an explicit rate region by reducing those five auxiliary random variables into three and then choosing a joint input distribution appropriately.*

[2]It is still unknown whether or not the exact feedback capacity region includes only three types of inequalities.

2.4.2 An Outer Bound Region

Theorem 3. *The feedback capacity region is included by the set* $\overline{\mathcal{C}}$ *of* (R_1, R_2) *such that*

$$R_1 \leq \log\left(1 + \mathsf{SNR}_1 + \mathsf{INR}_{21} + 2\rho\sqrt{\mathsf{SNR}_1 \cdot \mathsf{INR}_{21}}\right) \tag{2.36}$$

$$R_1 \leq \log\left(1 + (1-\rho^2)\mathsf{INR}_{12}\right) + \log\left(1 + \frac{(1-\rho^2)\mathsf{SNR}_1}{1 + (1-\rho^2)\mathsf{INR}_{12}}\right) \tag{2.37}$$

$$R_2 \leq \log\left(1 + \mathsf{SNR}_2 + \mathsf{INR}_{12} + 2\rho\sqrt{\mathsf{SNR}_2 \cdot \mathsf{INR}_{12}}\right) \tag{2.38}$$

$$R_2 \leq \log\left(1 + (1-\rho^2)\mathsf{INR}_{21}\right) + \log\left(1 + \frac{(1-\rho^2)\mathsf{SNR}_2}{1 + (1-\rho^2)\mathsf{INR}_{21}}\right) \tag{2.39}$$

$$R_1 + R_2 \leq \log\left(1 + \frac{(1-\rho^2)\mathsf{SNR}_1}{1 + (1-\rho^2)\mathsf{INR}_{12}}\right) + \log\left(1 + \mathsf{SNR}_2 + \mathsf{INR}_{12} + 2\rho\sqrt{\mathsf{SNR}_2 \cdot \mathsf{INR}_{12}}\right) \tag{2.40}$$

$$R_1 + R_2 \leq \log\left(1 + \frac{(1-\rho^2)\mathsf{SNR}_2}{1 + (1-\rho^2)\mathsf{INR}_{21}}\right) + \log\left(1 + \mathsf{SNR}_1 + \mathsf{INR}_{21} + 2\rho\sqrt{\mathsf{SNR}_1 \cdot \mathsf{INR}_{21}}\right) \tag{2.41}$$

for $0 \leq \rho \leq 1$.

Proof. By symmetry, it suffices to prove the bounds of (2.36), (2.37) and (2.40). The bounds of (2.36) and (2.37) are nothing but cutset bounds. Hence, proving the non-cutset bound of (2.40) is the main focus of this proof. Also recall that this non-cutset bound is used to obtain the outer bound of (2.7) for the symmetric capacity in Theorem 1. We go through the proof of (2.36) and (2.37). We then focus on the proof of (2.40), where we will also provide insights as to the proof idea.

Proof of (2.36): Starting with Fano's inequality, we get:

$$N(R_1 - \epsilon_N) \leq I(W_1; Y_1^N) \overset{(a)}{\leq} \sum [h(Y_{1i}) - h(Z_{1i})],$$

where (a) follows from the fact that conditioning reduces entropy. Assume that X_1 and X_2 have covariance ρ, i.e., $E[X_1 X_2^*] = \rho$. Then, we get:

$$h(Y_1) \leq \log 2\pi e \left(1 + \mathsf{SNR}_1 + \mathsf{INR}_{21} + 2|\rho|\sqrt{\mathsf{SNR}_1 \cdot \mathsf{INR}_{21}}\right).$$

If (R_1, R_2) is achievable, then $\epsilon_N \to 0$ as $N \to \infty$. Therefore, we get the desired bound:

$$R_1 \leq h(Y_1) - h(Z_1) \leq \log\left(1 + \mathsf{SNR}_1 + \mathsf{INR}_{21} + 2|\rho|\sqrt{\mathsf{SNR}_1 \cdot \mathsf{INR}_{21}}\right).$$

Proof of (2.37): Starting with Fano's inequality, we get:

$$\begin{aligned}
N(R_1-\epsilon_N) &\le I(W_1;Y_1^N,Y_2^N,W_2)\\
&\overset{(a)}{=} \sum[h(Y_{1i},Y_{2i}|W_2,Y_1^{i-1},Y_2^{i-1})-h(Z_{1i})-h(Z_{2i})]\\
&\overset{(b)}{=} \sum[h(Y_{1i},Y_{2i}|W_2,Y_1^{i-1},Y_2^{i-1},X_2^i)-h(Z_{1i})-h(Z_{2i})]\\
&\overset{(c)}{=} \sum[h(Y_{2i}|W_2,Y_1^{i-1},Y_2^{i-1},X_2^i)-h(Z_{2i})]\\
&\quad+\sum[h(Y_{1i}|W_2,Y_1^{i-1},Y_2^{i-1},X_2^i,Y_{2i},S_1^i)-h(Z_{1i})]\\
&\overset{(d)}{\le} \sum\left[h(Y_{2i}|X_{2i})-h(Z_{2i})+h(Y_{1i}|X_{2i},S_{1i})-h(Z_{1i})\right]
\end{aligned}$$

where (a) follows from the fact that W_1 is independent from W_2 and $h(Y_1^N,Y_2^N|W_1,W_2)=h(Y_1^N,S_1^N|W_1,W_2)=\sum[h(Z_{1i})+h(Z_{2i})]$ (see Claim 1 below); (b) follows from the fact that X_2^i is a function of (W_2,Y_2^{i-1}); (c) follows from the fact that S_1^i is a function of (Y_2^i,X_2^i); (d) follows from the fact that conditioning reduces entropy. Hence, we get the desired result:

$$\begin{aligned}
R_1 &\le h(Y_2|X_2)-h(Z_2)+h(Y_1|X_2,S_1)-h(Z_1)\\
&\overset{(a)}{\le} \log\left(1+(1-|\rho|^2)\mathsf{INR}_{12}\right)+\log\left(1+\frac{(1-|\rho|^2)\mathsf{SNR}_1}{1+(1-|\rho|^2)\mathsf{INR}_{12}}\right)
\end{aligned}$$

where (a) follows from the fact that

$$h(Y_2|X_2)\le \log 2\pi e\left(1+(1-|\rho|^2)\mathsf{INR}_{12}\right), \tag{2.42}$$

$$h(Y_1|X_2,S_1)\le \log 2\pi e\left(1+\frac{(1-|\rho|^2)\mathsf{SNR}_1}{1+(1-|\rho|^2)\mathsf{INR}_{12}}\right). \tag{2.43}$$

The inequality of (2.43) is obtained as follows. Given (X_2,S_1), the variance of Y_1 is upper-bounded by

$$\mathrm{Var}\left[Y_1|X_2,S_1\right]\le K_{Y_1}-K_{Y_1(X_2,S_1)}K_{(X_2,S_1)}^{-1}K_{Y_1(X_2,S_1)}^*,$$

where

$$\begin{aligned}
K_{Y_1} &= E\left[|Y_1|^2\right]=1+\mathsf{SNR}_1+\mathsf{INR}_{21}+\rho g_{11}^*g_{21}+\rho^* g_{11}g_{21}^*,\\
K_{Y_1(X_2,S_1)} &= E\left[Y_1[X_2^*,S_1^*]\right]=\left[\rho g_{11}+g_{21},g_{12}^*g_{11}+\rho^*g_{21}g_{12}^*\right],\\
K_{(X_2,S_1)} &= E\left[\begin{bmatrix}|X_2|^2 & X_2S_1^*\\ X_2^*S_1 & |S_1|^2\end{bmatrix}\right]=\begin{bmatrix}1 & \rho^*g_{12}^*\\ \rho g_{12} & 1+\mathsf{INR}_{12}\end{bmatrix}.
\end{aligned}$$

By further calculation, we can get (2.43).

Claim 1. $h(Y_1^N,S_1^N|W_1,W_2)=\sum\left[h(Z_{1i})+h(Z_{2i})\right].$

Proof.

$$\begin{aligned} h(Y_1^N, S_1^N|W_1, W_2) &= \sum h(Y_{1i}, S_{1i}|W_1, W_2, Y_1^{i-1}, S_1^{i-1}) \\ &\overset{(a)}{=} \sum h(Y_{1i}, S_{1i}|W_1, W_2, Y_1^{i-1}, S_1^{i-1}, X_{1i}, X_{2i}) \\ &\overset{(b)}{=} \sum h(Z_{1i}, Z_{2i}|W_1, W_2, Y_1^{i-1}, S_1^{i-1}, X_{1i}, X_{2i}) \\ &\overset{(c)}{=} \sum \left[h(Z_{1i}) + h(Z_{2i})\right], \end{aligned}$$

where (a) follows from the fact that X_{1i} is a function of (W_1, Y_1^{i-1}) and X_{2i} is a function of (W_2, S_1^{i-1}) (by Claim 2 below); (b) follows from the fact that $Y_{1i} = g_{11}X_{1i} + g_{21}X_{2i} + Z_{1i}$ and $S_{1i} = g_{12}X_{1i} + Z_{2i}$; (c) follows from the memoryless property of the channel and the independence assumption of Z_{1i} and Z_{2i}. □

Claim 2. *For all $i \geq 1$, X_1^i is a function of (W_1, S_2^{i-1}) and X_2^i is a function of (W_2, S_1^{i-1}).*

Proof. By symmetry, it is enough to prove only one. Notice that X_2^i is a function of (W_2, Y_2^{i-1}) and Y_2^{i-1} is a function of (X_2^{i-1}, S_1^{i-1}). Hence, X_2^i is a function of $(W_2, X_2^{i-1}, S_1^{i-1})$. Iterating the same argument, we conclude that X_2^i is a function of (W_2, X_{21}, S_1^{i-1}). Since X_{21} depends only on W_2, we complete the proof. □

Proof of (2.40): The proof idea is based on the genie-aided argument [24]. However, finding an appropriate genie is not simple since there are many possible combinations of the random variables. The deterministic IC example in Fig. 2.5 (b) gives insights into this. Note that providing a_1 and (b_1, b_2, b_3) to receiver 1 does not increase the rate R_1, i.e., these are *useless gifts*. This may motivate us to choose a genie as $(g_{12}X_1, W_2)$. However, in the Gaussian channel, providing $g_{12}X_1$ is equivalent to providing X_1. This is of course too much information, inducing a loose upper bound. Inspired by the technique in [25], we instead consider a noisy version of $g_{12}X_1$:

$$S_1 = g_{12}X_1 + Z_2. \tag{2.44}$$

Intuition behind this is that we cut off $g_{12}X_1$ at the noise level. Indeed this matches intuition in the deterministic IC. This genie together with W_2 turns out to lead to the desired tight upper bound.

Starting with Fano's inequality, we get:

$$\begin{aligned}
N(R_1+R_2-\epsilon_N) &\le I(W_1;Y_1^N)+I(W_2;Y_2^N)\\
&\overset{(a)}{\le} I(W_1;Y_1^N,S_1^N,W_2)+I(W_2;Y_2^N)\\
&\overset{(b)}{=} h(Y_1^N,S_1^N|W_2)-h(Y_1^N,S_1^N|W_1,W_2)+I(W_2;Y_2^N)\\
&\overset{(c)}{=} h(Y_1^N,S_1^N|W_2)-\sum\left[h(Z_{1i})+h(Z_{2i})\right]+I(W_2;Y_2^N)\\
&\overset{(d)}{=} h(Y_1^N|S_1^N,W_2)-\sum h(Z_{1i})+h(Y_2^N)-\sum h(Z_{2i})\\
&\overset{(e)}{=} h(Y_1^N|S_1^N,W_2,X_2^N)-\sum h(Z_{1i})+h(Y_2^N)-\sum h(Z_{2i})\\
&\overset{(f)}{\le} \sum_{i=1}^{N}\left[h(Y_{1i}|S_{1i},X_{2i})-h(Z_{1i})+h(Y_{2i})-h(Z_{2i})\right]
\end{aligned}$$

where (a) follows from the fact that adding information increases mutual information (providing a *genie*); (b) follows from the independence of W_1 and W_2; (c) follows from $h(Y_1^N,S_1^N|W_1,W_2)=\sum\left[h(Z_{1i})+h(Z_{2i})\right]$ (see Claim 1); (d) follows from $h(S_1^N|W_2)=h(Y_2^N|W_2)$ (see Claim 3 below); (e) follows from the fact that X_2^N is a function of (W_2,S_1^{N-1}) (see Claim 2); (f) follows from the fact that conditioning reduces entropy.

Hence, we get

$$R_1+R_2\le h(Y_1|S_1,X_2)-h(Z_1)+h(Y_2)-h(Z_2).$$

Note that

$$h(Y_2)\le \log 2\pi e\left(1+\mathsf{SNR}_2+\mathsf{INR}_{12}+2|\rho|\sqrt{\mathsf{SNR}_2\cdot\mathsf{INR}_{12}}\right). \tag{2.45}$$

From (2.43) and (2.45), we get the desired upper bound.

Claim 3. $h(S_1^N|W_2)=h(Y_2^N|W_2)$.

Proof.

$$\begin{aligned}
h(Y_2^N|W_2) &= \sum h(Y_{2i}|Y_2^{i-1},W_2)\\
&\overset{(a)}{=} \sum h(S_{1i}|Y_2^{i-1},W_2)\\
&\overset{(b)}{=} \sum h(S_{1i}|Y_2^{i-1},W_2,X_2^i,S_1^{i-1})\\
&\overset{(c)}{=} \sum h(S_{1i}|W_2,S_1^{i-1})=h(S_1^N|W_2),
\end{aligned}$$

where (a) follows from the fact that Y_{2i} is a function of (X_{2i},S_{1i}) and X_{2i} is a function of (W_2,Y_2^{i-1}); (b) follows from the fact that X_2^i is a function of (W_2,Y_2^{i-1}) and S_1^{i-1} is a function of (Y_2^{i-1},X_2^{i-1}); (c) follows from the fact that Y_2^{i-1} is a function of (X_2^{i-1},S_1^{i-1}) and X_2^i is a function of (W_2,S_1^{i-1}) (by Claim 2). □

□

2.4.3 2-Bit Gap to the Capacity Region

Theorem 4. *The gap between the inner and outer bound regions (given in Theorems 2 and 3) is at most* 2 *bits/s/Hz/user:*

$$\mathcal{R} \subseteq \mathcal{C} \subseteq \overline{\mathcal{C}} \subseteq \mathcal{R} \oplus ([0,2] \times [0,2]) \,. \tag{2.46}$$

Proof. The proof is immediate by Theorem 2 and 3. We define δ_1 to be the difference between $\min\{(2.36),(2.37)\}$ and $\min\{(2.18),(2.19)\}$. Similarly, we define δ_2 and δ_{12}. Straightforward computation gives

$$\delta_1 \leq \max\left\{1, \log\left(1+\frac{\mathsf{SNR}_1}{1+\mathsf{INR}_{12}}\right) - \log\left(2+\frac{\mathsf{SNR}_1}{\mathsf{INR}_{12}}\right)+2\right\} \leq 2.$$

Similarly, we get $\delta_2 \leq 2$ and $\delta_{12} \leq 2$. This completes the proof. □

Remark 8 (**Why does a 2-bit gap occur?**). *The achievable scheme meant for the capacity region involves message-splitting. As mentioned in Remark 4, message-splitting incurs some loss in the process of decoding the common message while treating private signals as noise. Accounting for the effect of private signals, the effective noise power becomes double, thus incurring a 1-bit gap. The other 1-bit gap comes from a relay structure of the feedback IC. To see this, consider an extreme case where user 2's rate is completely ignored. In this case, we can view the* $[transmitter2, receiver2]$ *communication pair as a single relay which only helps the* $[transmitter1, receiver1]$ *communication pair. It has been shown in [7] that for this single relay Gaussian channel, the worst-case gap between the best known inner bound [16] and the outer bound is 1 bit/s/Hz. This incurs the other 1-bit gap. This 2-bit gap is based on the outer bound region in Theorem 3, which allows for arbitrary correlation between the transmitters. So, one can expect that the actual gap to the capacity region is less than 2 bits.*

Remark 9 (**Reducing the gap**). *As discussed, the amplify-and-forward scheme has the potential to reduce the gap. However, due to the inherent relay structure, reducing the gap into a less-than-one bit is challenging. As long as no significant progress is made on the single relay Gaussian channel, one cannot easily reduce the gap further.*

Remark 10 (**Comparison with the two-staged scheme**). *Specializing to the symmetric rate, it can be shown that the infinite-staged scheme in Theorem 2 can achieve the symmetric capacity to within 1 bit. Coincidentally, this gap matches the gap result of the two-staged scheme in Theorem 1. However, the 1-bit gap comes from different reasons. In the infinite-staged scheme, the 1-bit gap comes from message-splitting. In contrast, in the two-staged scheme, the gap is due to lack of beamforming gain. One needs to come up with a new technique that well combines these two schemes to reduce the gap into a less-than-one bit.*

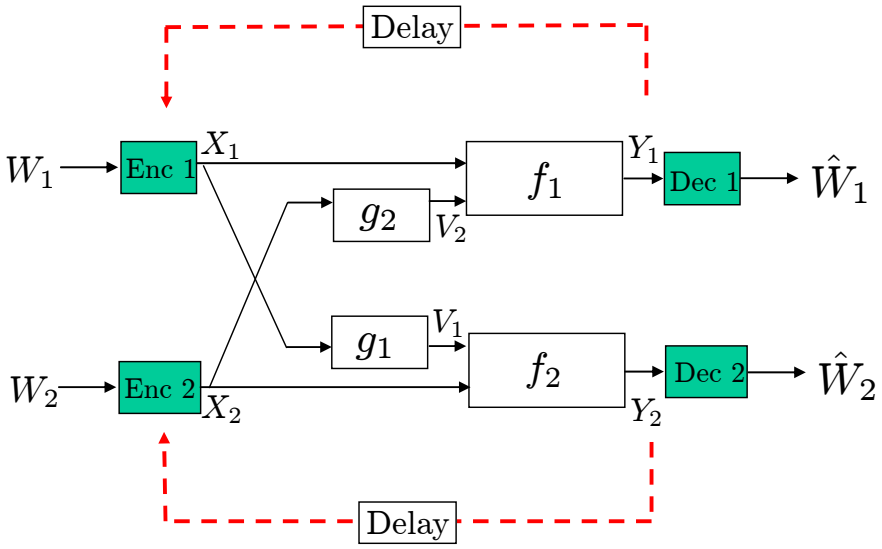


Figure 2.10: El Gamal-Costa deterministic IC with feedback

2.5 Feedback Capacity of the El Gamal-Costa Model

We have so far made use of the linear deterministic IC to provide insights into approximating the feedback capacity region of the Gaussian IC. The deterministic IC is a special case of El Gamal-Costa deterministic IC [24]. In this section, we establish the exact feedback capacity region for this general class of deterministic ICs.

Fig. 2.10 (a) illustrates El Gamal-Costa deterministic IC with feedback. The key condition of this model is given by

$$\begin{aligned} H(V_2|Y_1, X_1) &= 0, \\ H(V_1|Y_2, X_2) &= 0, \end{aligned} \tag{2.47}$$

where V_k is a part of X_k $(k = 1, 2)$, visible to the other receiver. This implies that in any working system where X_1 and X_2 are decodable at receivers 1 and 2 respectively, V_1 and V_2 are completely determined at receivers 2 and 1 respectively, i.e., these are common signals.

Theorem 5. *The feedback capacity region of El Gamal-Costa deterministic IC is the set of* (R_1, R_2) *such that*

$$\begin{aligned} R_1 &\leq \min\{H(Y_1), H(Y_2|X_2, U) + H(Y_1|V_1, V_2, U)\} \\ R_2 &\leq \min\{H(Y_2), H(Y_1|X_1, U) + H(Y_2|V_1, V_2, U)\} \\ R_1 + R_2 &\leq \min\{H(Y_1|V_1, V_2, U) + H(Y_2), H(Y_2|V_2, V_1, U) + H(Y_1)\} \end{aligned}$$

for some joint distribution $p(u, x_1, x_2) = p(u)p(x_1|u)p(x_2|u)$. *Here* U *is a discrete random variable which takes on values in the set* $\mathcal{U}$ *where* $|\mathcal{U}| \leq \min(|\mathcal{V}_1||\mathcal{V}_2|, |\mathcal{Y}_1|, |\mathcal{Y}_2|) + 3$.

Proof. Achievability proof is straightforward by Lemma 1. Set $U_k = V_k, \forall k$. Fix a joint distribution $p(u)p(u_1|u)p(u_2|u)p(x_1|u_1,u)p(x_2|u_2,u)$. We now write a joint distribution $p(u,x_1,x_2,u_1,u_2)$ in two different ways:

$$\begin{aligned}&p(u,x_1,x_2,u_1,u_2)\\&=p(u)p(x_1|u)p(x_2|u)\delta(u_1-g_1(x_1))\delta(u_2-g_2(x_2))\\&=p(u)p(u_1|u)p(u_2|u)p(x_1|u_1,u)p(x_2|u_2,u)\end{aligned}$$

where $\delta(\cdot)$ indicates the Kronecker delta function. This gives

$$\begin{aligned}p(x_1|u)&:=\frac{p(x_1|u_1,u)p(u_1|u)}{\delta(u_1-g_1(x_1))},\\p(x_2|u)&:=\frac{p(x_2|u_2,u)p(u_2|u)}{\delta(u_2-g_2(x_2))}.\end{aligned}$$

Now we can generate a joint distribution $p(u)p(x_1|u)p(x_2|u)$. Hence, we complete the achievability proof. See Appendix 2.9.3 for converse proof. □

As a by-product, we obtain the feedback capacity region of the linear deterministic IC.

Corollary 1. *The feedback capacity region of the linear deterministic IC is the set of* (R_1,R_2) *such that*

$$\begin{aligned}R_1&\le\min\left\{\max(n_{11},n_{12}),\max(n_{11},n_{21})\right\}\\R_2&\le\min\left\{\max(n_{22},n_{21}),\max(n_{22},n_{12})\right\}\\R_1+R_2&\le\min\left\{\max(n_{22},n_{12})+(n_{11}-n_{12})^+,\max(n_{11},n_{21})+(n_{22}-n_{21})^+\right\}.\end{aligned}$$

Proof. The proof is straightforward by Theorem 5. The capacity region is achieved when U is constant; and X_1 and X_2 are independent and uniformly distributed. □

2.6 Role of Feedback

Recall in Fig. 2.1 that feedback gain is *bounded* for $0\le\alpha\le\frac{2}{3}$ in terms of the symmetric rate. So a natural question that arises is to ask whether feedback gain is marginal also from a capacity-region perspective in this parameter range. With the help of Corollary 1, we show that feedback can provide *multiplicative* gain even in this regime. We next revisit the resource hole interpretation in Remark 2. With this interpretation, we address another interesting question posed in Section 2.4: why is the $2R_1+R_2$ bound missing with feedback?

Feedback Gain from a Capacity Region Perspective: Fig. 2.11 shows the feedback capacity region of the linear deterministic IC under the symmetric channel setting: $n=n_{11}=n_{22}$ and $m=n_{12}=n_{21}$. Interestingly, while for $\frac{2}{3}\le\alpha\le 2$, the symmetric capacity does not improve with feedback, the feedback capacity region is enlarged even for this regime. This implies that feedback gain could be significant in terms of the capacity region, even when there is no improvement with feedback in terms of the symmetric capacity.

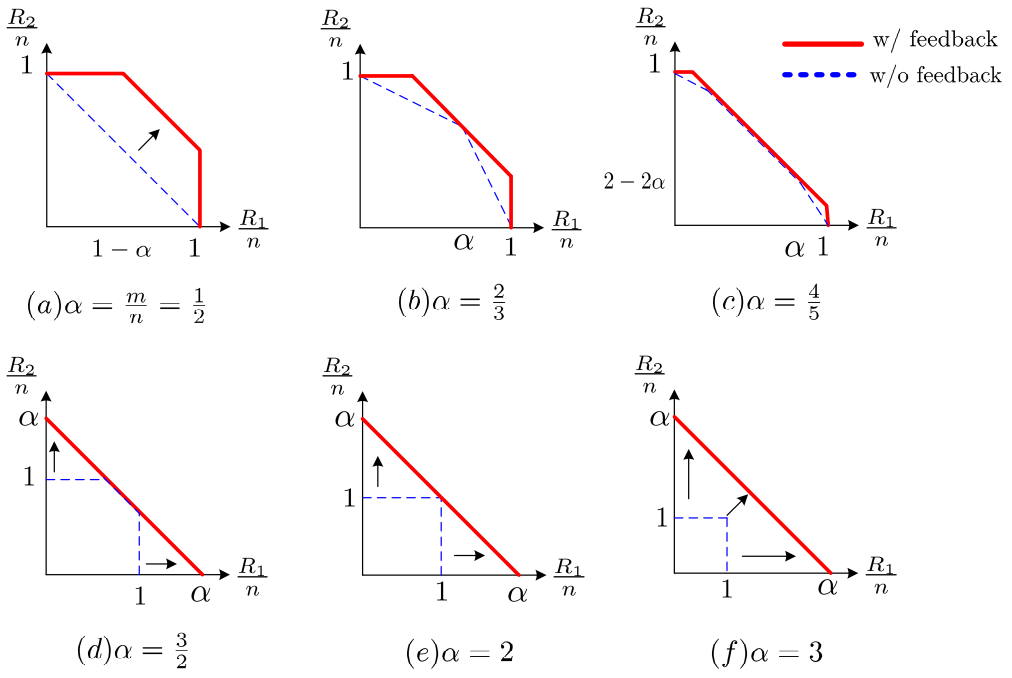


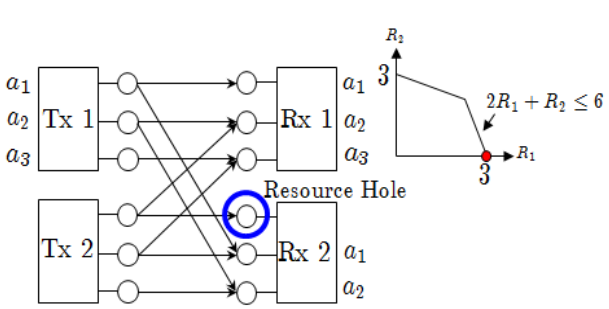
Figure 2.11: Feedback capacity region of the linear deterministic IC. This shows that feedback gain could be significant in terms of the capacity region, even when there is no improvement due to feedback in terms of the symmetric capacity.

2.6.1 Resource Hole Interpretation

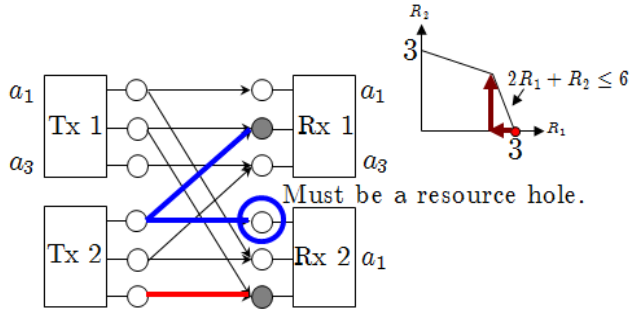
Recall the role of feedback in Remark 2: feedback maximizes resource utilization by filling up all the resource holes under-utilized in the non-feedback case. Using this interpretation, we can provide an intuitive explanation why $2R_1 + R_2$ bound is missing with feedback.

To see this, consider an example where $2R_1 + R_2$ bound is active in the non-feedback case. Fig. 2.12 (a) shows an example where a corner point of $(3, 0)$ can be achieved. Observe that at the two receivers, the five signal levels are consumed out of the six signal levels. There is one resource hole. This resource hole is closely related to the $2R_1 + R_2$ bound, which will be shown in Fig. 2.12 (b).

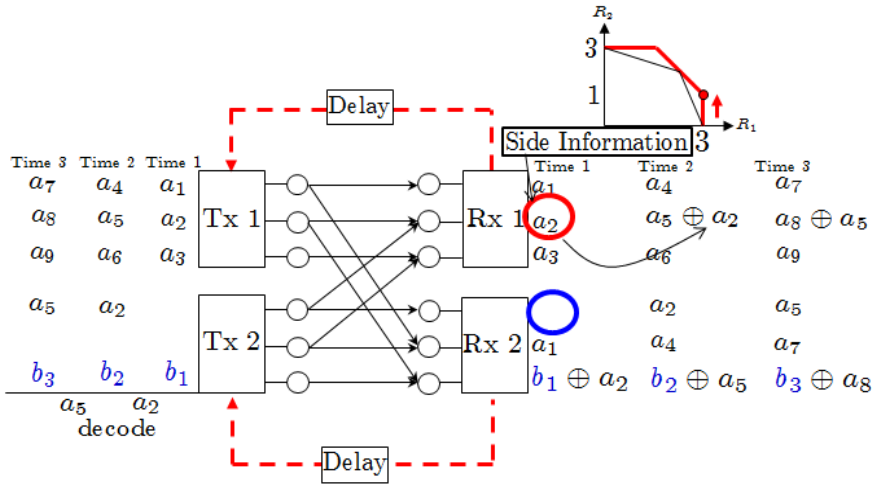
Suppose the $2R_1 + R_2$ bound is active. This implies that if R_1 is reduced by 1 bit, then R_2 should be increased by 2 bits. Suppose that in order to decrease R_1 by 1 bit, transmitter 1 sends no information on the second signal level. We then see the two empty signal levels at the two receivers (marked as the gray balls): one at the second level at receiver 1; the other at the bottom level at receiver 2. Transmitter 2 can now send 1 bit on the bottom level to increase R_2 by 1 bit (marked as the thick red line). Also it allows transmitter 2 to send one more bit on the top level. This implies that the top level at receiver 2 must be a resource hole in the previous case. This observation combined with the following observation can give



(a) A resource hole vs. $2R_1 + R_2$ bound



(b) Whenever $2R_1 + R_2$ is active, there is a resource hole.



(c) Feedback fills up all resource holes to maximize resource utilization

Figure 2.12: Relationship between a resource hole and $2R_1 + R_2$ bound. The $2R_1 + R_2$ bound is missing with feedback.

an answer to the question.

Fig. 2.12 (c) shows the feedback role that it fills up all the resource holes to maximize resource utilization. We employ the same feedback strategy used in Fig. 2.9 to obtain the result in Fig. 2.12 (c). Notice that with feedback, all of the resource holes are filled up except a hole in the first stage, which can be amortized by employing an infinite number of stages. Therefore, we can now see why the $2R_1 + R_2$ bound is missing with feedback.

2.6.2 Side Information Interpretation

By carefully looking at the feedback scheme in Fig. 2.12(c), we develop another interpretation as to the role of feedback. Recall that in the non-feedback case that achieves the $(3, 0)$ corner point, the broadcast nature of the wireless medium precludes transmitter 2 from using any levels, as transmitter 1 is already using all of the levels. In contrast, if feedback is allowed, transmitter 2 can now use some levels to improve the non-feedback rate. Suppose that transmitters 1 and 2 send (a_1, a_2, a_3) and b_1 through their signal levels respectively. Receivers 1 and 2 then get the bits (a_1, a_2, a_3) and $(a_2, b_1 \oplus a_2)$ respectively. With feedback, in the second stage, the bit a_2 - received cleanly at the desired receiver while interfering with b_1 at the other receiver - can be exploited as *side information* to increase the non-feedback capacity. For example, with feedback transmitter 2 decodes the other user's bit a_2 and forwards it through the top level. This transmission allows receiver 2 to refine the corrupted bit b_1 from $b_1 \oplus a_2$. This seems to cause interference to receiver 1. But this does not cause interference since receiver 1 already had the *side information* of a_2 from the previous broadcasting. We exploited the side information with the help of feedback to refine the corrupted bit without causing interference. With this interpretation, we can now make a connection between our feedback problem and a variety of other problems in network information theory [5, 71, 38, 8, 28, 45].

Connection to Other Problems (Fig. 2.13): In 2000, Alshwede-Cai-Li-Yeung [5] invented the breakthrough concept of network coding and came up with the butterfly example where the network coding combined with the idea of exploiting side information can significantly improve the routing performance. This result shows that exploiting side information plays an important role in decoding the desired signals from the network-coded signals (equations). This network coding idea combined with the idea of exploiting side information was shown to be powerful in wireless networks as well [71, 38]. Specifically, in the context of two-way relay channels, it was shown that the broadcast nature of wireless medium can be exploited to generate side information, and this generated side information plays a crucial role in increasing capacity. Subsequently, the index coding problem was introduced by Bar-Yossef, *et.al.* [8] where the significant impact of side information was directly addressed.

In our work, as a consequence of addressing the two-user Gaussian IC with feedback, we develop an interpretation as to the role of feedback: feedback enables receivers to exploit their received signals as *side information*, thus improving the non-feedback capacity significantly.

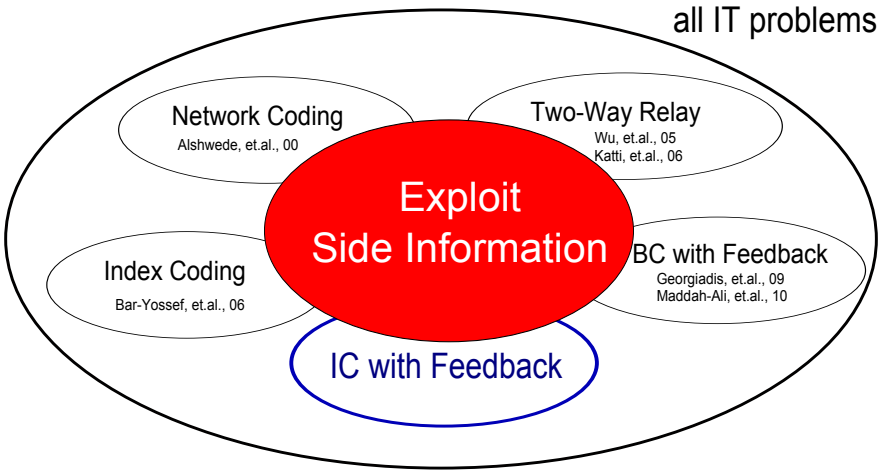


Figure 2.13: Connection to other problems in network information theory: network coding problems [5]; two-way relay channels [71]; general wireless networks [38]; index coding problems [8]; broadcast erasure channels with feedback [28]; and MIMO Gaussian broadcast channels with outdated channel state feedback [45].

With the help of this interpretation, we find that all of the above problems can be intimately linked through the common idea of *exploiting side information*.

Very recently, the authors in [28, 45] came up with interesting results on feedback capacity. Georgiadis and Tassiulas [28] showed that feedback can significantly increase the capacity of the broadcast erasure channel. Maddah-Ali and Tse [45] showed that *channel state feedback*, although it is *outdated*, can increase the non-feedback MIMO broadcast channel capacity. We find that interestingly the role of feedback in these channels is the same as that in our problem: feedback enables receivers to exploit their received signals as side information to increase capacity. This reveals a connection to the above problems.

2.7 Discussion

Comparison to Related Work [41, 42, 27]: For the symmetric Gaussian IC, Kramer [41, 42] developed a feedback strategy based on the Schalkwijk-Kailath scheme [53] and the Ozarow scheme [47]. Due to lack of closed-form rate-formula for the scheme, we cannot see how the Kramer scheme is close to our symmetric rate in Theorem 1. To see this, we compute the generalized degrees-of-freedom of the Kramer scheme.

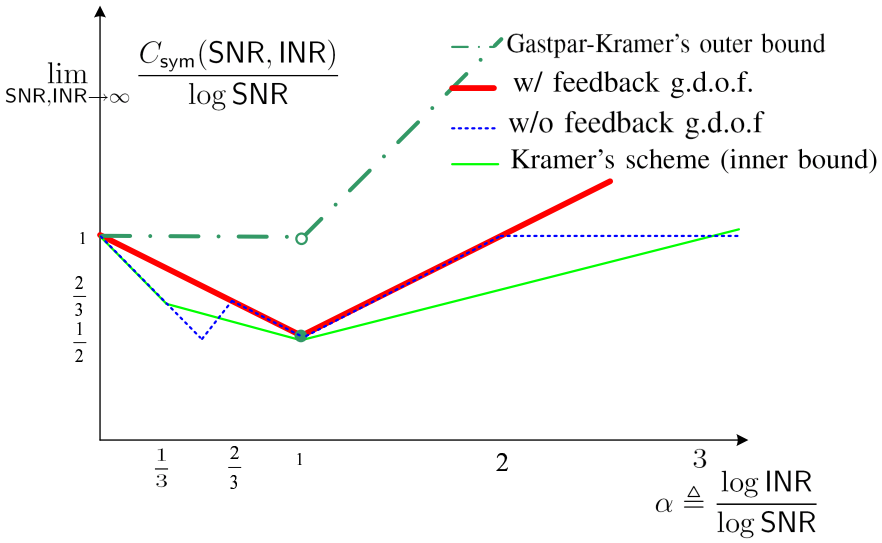


Figure 2.14: Generalized degrees-of-freedom comparison.

Lemma 2. *The generalized degrees-of-freedom of the Kramer scheme is given by*

$$\underline{d}(\alpha) = \begin{cases} 1-\alpha, & 0 \le \alpha < \frac{1}{3}; \\ \frac{3-\alpha}{4}, & \frac{1}{3} \le \alpha < 1; \\ \frac{1+\alpha}{4}, & \alpha \ge 1. \end{cases} \tag{2.48}$$

Proof. See Appendix 2.9.4. □

Note in Fig. 2.14 that the Kramer scheme can be arbitrarily far from optimality, i.e., it has an unbounded gap to the symmetric capacity for all values of α except $\alpha = 1$. We also plot the symmetric rate for finite channel parameters as shown in Fig. 2.15. Notice that the Kramer scheme is very close to the outer bounds only when INR is similar to SNR. In fact, the capacity theorem in [42] says that they match each other at $\mathsf{INR} = \mathsf{SNR} - \sqrt{2\mathsf{SNR}}$. However, if INR is quite different from SNR, it becomes far away from the outer bounds. Also note that our new bound is much tighter than Gastpar-Kramer's outer bounds in [41, 27].

Closing the Gap: *Less than 1-bit gap to the symmetric capacity*: Fig. 2.15 implies that our achievable scheme can be improved especially when $\alpha \approx 1$ where beamforming gain plays a significant role. As mentioned earlier, our two-staged scheme completely loses beamforming gain. In contrast, the Kramer scheme captures the beamforming gain. As discussed in Remark 10, one may develop a unified scheme that beats both the schemes for all channel parameters to reduce the worst-case gap.

Less than 2-bit gap to the capacity region: As mentioned in Remark 8, a 2-bit gap to the feedback capacity region can be improved up to a 1-bit gap. The idea is to remove message splitting. Recall that the Alamouti-based amplify-and-forward scheme in Theorem 1

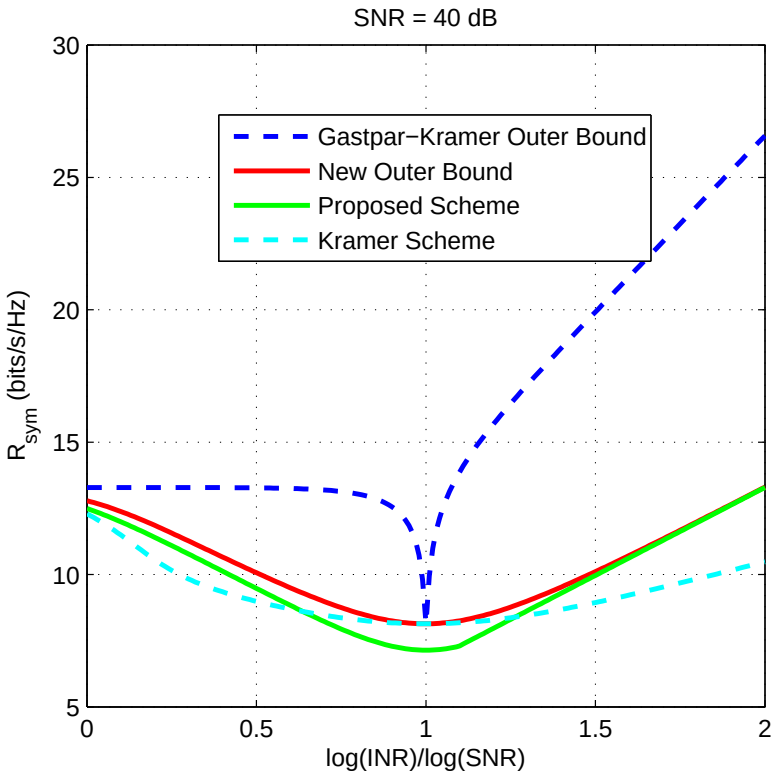


Figure 2.15: Symmetric rate comparison.

improves the performance by removing message splitting. Translating the same idea to the characterization of the capacity region is needed for the improvement. A noisy binary expansion model in Fig. 2.17 may give insights into this.

Extension to Gaussian MIMO ICs with Feedback: The feedback capacity result for El Gamal-Costa model can be extended to Teletar-Tse IC [64] where in Fig. 2.10, f_k's are deterministic functions satisfying El Gamal-Costa condition (2.47) while g_k's follow arbitrary probability distributions. Once extended, one can infer an approximate feedback capacity region of the two-user Gaussian MIMO IC, as [64] did in the non-feedback case.

2.8 Summary

We have established the feedback capacity region to within 2 bits/s/Hz/user and the symmetric capacity to within 1 bit/s/Hz/user universally for the two-user Gaussian IC with feedback. The Alamouti scheme inspires our two-staged achievable scheme meant for the symmetric rate. For an achievable rate region, we have employed block Markov encoding

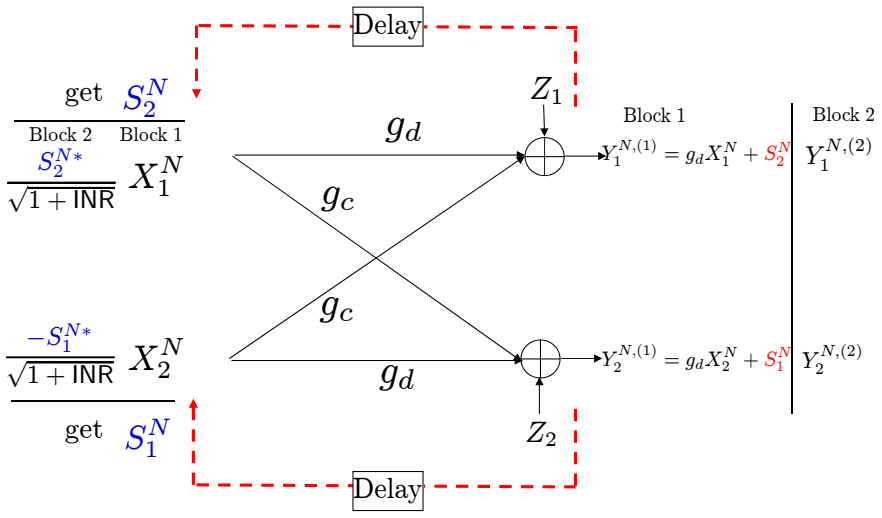


Figure 2.16: An achievable scheme in the symmetric Gaussian IC: Alamouti-based amplify-and-forward scheme

to incorporate an infinite number of stages. A new outer bound was derived to provide an approximate characterization of the capacity region. As a side-generalization, we have characterized the exact feedback capacity region of El Gamal-Costa deterministic IC.

An interesting consequence of our result is that feedback could provide *multiplicative* gain in many-to-many channels unlike point-to-point, many-to-one, or one-to-many channels. We develop two interpretations as to how feedback can provide significant gain. One interpretation is that feedback maximizes resource utilization by filling up all the resource holes under-utilized in the non-feedback case. The other interpretation is that feedback can exploit received signals as side information to increase capacity. The latter interpretation leads us to make a connection to other problems.

2.9 Appendices

2.9.1 Achievable Scheme for the Symmetric Rate of (2.6)

The scheme uses two stages (blocks). In the first stage, each transmitter k sends codeword X_k^N with rate R_k. In the second stage, with feedback transmitter 1 gets the interference plus noise: $S_2^N = g_c X_2^N + Z_1^{(1),N}$. Now the complex conjugate technique based on Alamouti's scheme is applied to make X_1^N and S_2^N well separable. Transmitters 1 and 2 send $\frac{S_2^{N*}}{\sqrt{1+\mathsf{INR}}}$ and $-\frac{S_1^{N*}}{\sqrt{1+\mathsf{INR}}}$ respectively, where $\sqrt{1+\mathsf{INR}}$ is a normalization factor to meet the power constraint.

Receiver 1 can then gather the two received signals: for $1 \le i \le N$,

$$\mathbf{Y}_i \triangleq \begin{bmatrix} Y_{1i}^{(1)} \\ Y_{1i}^{(2)*} \end{bmatrix} = \begin{bmatrix} g_d & 1 \\ -\frac{\mathsf{INR}}{\sqrt{1+\mathsf{INR}}} & \frac{g_d^*}{\sqrt{1+\mathsf{INR}}} \end{bmatrix} \begin{bmatrix} X_{1i} \\ S_{2i} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{g_c^*}{\sqrt{1+\mathsf{INR}}} Z_{2i}^{(1)} + Z_{1i}^{(2)*} \end{bmatrix}.$$

Under Gaussian input distribution, we can compute the rate under MMSE demodulation:

$$\frac{1}{2} I(X_{1i}; \mathbf{Y}_i) = \frac{1}{2} h(\mathbf{Y}_i) - \frac{1}{2} h(\mathbf{Y}_i | X_{1i}) = \frac{1}{2} \log \frac{|K_{\mathbf{Y}_i}|}{|K_{\mathbf{Y}_i|X_{1i}}|}.$$

Straightforward calculations give

$$|K_{\mathbf{Y}_i}| = \left| \begin{bmatrix} 1+\mathsf{SNR}+\mathsf{INR} & \frac{g_d}{\sqrt{1+\mathsf{INR}}} \\ \frac{g_d^*}{\sqrt{1+\mathsf{INR}}} & 1+\mathsf{SNR}+\mathsf{INR} \end{bmatrix} \right| = (1+\mathsf{SNR}+\mathsf{INR})^2 - \frac{\mathsf{SNR}}{1+\mathsf{INR}}$$

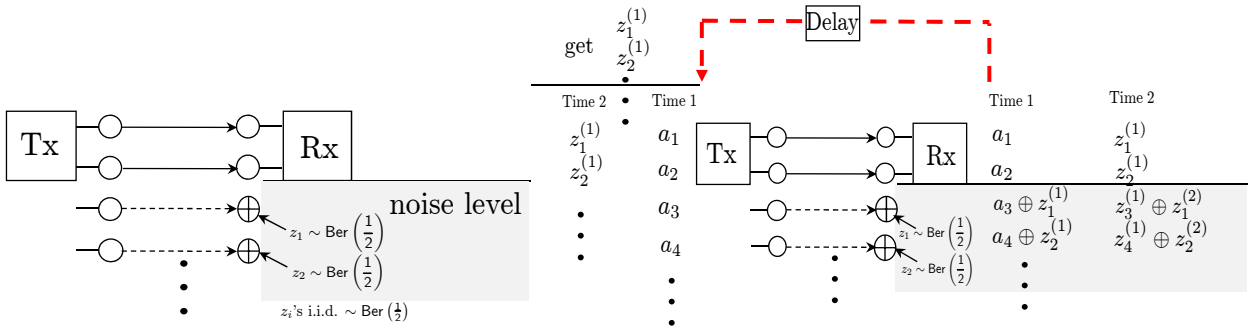
$$|K_{\mathbf{Y}_i|X_{1i}}| = \left| \begin{bmatrix} 1+\mathsf{INR} & g_d\sqrt{1+\mathsf{INR}} \\ g_d^*\sqrt{1+\mathsf{INR}} & \mathsf{SNR}+\frac{2\mathsf{INR}+1}{\mathsf{INR}+1} \end{bmatrix} \right| = 1+2\mathsf{INR}.$$

Therefore, we get the desired result: the right term in (2.6).

$$R_{\mathsf{sym}} = \frac{1}{2} \log \left(\frac{(1+\mathsf{SNR}+\mathsf{INR})^2 - \frac{\mathsf{SNR}}{1+\mathsf{INR}}}{1+2\mathsf{INR}} \right). \tag{2.49}$$

Intuition Behind the Proposed Scheme: To provide intuition behind our proposed scheme, we introduce a new model that we call a noisy binary expansion model, illustrated in Fig. 2.17 (a). In the non-feedback Gaussian channel, due to the absence of noise information at transmitter, transmitter has no chance to refine the corrupted received signal. On the other hand, if feedback is allowed, noise can be learned. Sending noise information (innovation) enables to refine the corrupted signal: the Schalkwijk-Kailath scheme [53]. However, the linear deterministic model cannot capture interplay between noise and signal. To capture this issue, we slightly modify the deterministic model so as to reflect the effect of noise. In this model, we assume that noise is a $\mathsf{Ber}(\frac{1}{2})$ random variable i.i.d. across time slots (memoryless) and levels. This induces the same capacity as that of the deterministic channel, so it matches the Gaussian channel capacity in the high SNR regime.

As a stepping stone towards the interpretation of the proposed scheme, let us first understand Schalkwijk-Kailath scheme [53] using this model. Fig. 2.17 (b) illustrates an example where 2 bits/time can be sent with feedback. In time 1, transmitter sends independent bit streams $(a_1, a_2, a_3, a_4, \cdots)$. Receiver then gets $(a_1, a_2, a_3 \oplus z_1^{(1)}, a_4 \oplus z_2^{(1)}, \cdots)$ where $z_i^{(j)}$ indicates an i.i.d. $\mathsf{Ber}\left(\frac{1}{2}\right)$ random variable of noise level i at time j. With feedback, transmitter can get noise information $(0, 0, z_1^{(1)}, z_2^{(1)}, \cdots)$ by subtracting the transmitted signals (sent previously) from the received feedback. This process corresponds to an MMSE operation in Schalkwijk-Kailath scheme: computing innovation. Transmitter scales the noise



(a) A noisy binary-expansion model (b) Interpretation of Schalkwijk-Kailath scheme

Figure 2.17: A noisy binary expansion model. Noise is assumed to be a $\mathsf{Ber}(\frac{1}{2})$ random variable i.i.d. across time slots (memoryless) and levels. This induces the same capacity as that of the deterministic channel, so it matches the Gaussian channel capacity in the high SNR regime.

information to shift it by 2 levels and then sends the shifted version. The shifting operation corresponds to a scaling operation in Schalkwijk-Kailath scheme. Receiver can now recover (a_3, a_4) corrupted by $(z_1^{(1)}, z_2^{(1)})$ in the previous slot. We repeat this procedure.

The viewpoint based on the binary expansion model can provide intuition behind our proposed scheme. See Fig. 2.18. In the first stage, each transmitter sends three independent bits: two bits above the noise level; one bit below the noise level. Transmitters 1 and 2 send (a_1, a_2, a_3) and (b_1, b_2, b_3) respectively. Receiver 1 then gets: (1) the clean signal a_1; (2) the interfered signal $a_2 \oplus b_1$; and (3) the interfered-and-noised signal $a_3 \oplus b_2 \oplus z_1^{(1)}$. Similarly for receiver 2. In the second stage, with feedback, each transmitter can get interference plus noise by subtracting the transmitted signals from the feedback. Transmitters 1 and 2 get $(0, b_1, b_2 \oplus z_1^{(1)})$ and $(0, a_1, a_2 \oplus z_2^{(1)})$ respectively. Next, each transmitter scales the subtracted signal subject to the power constraint and then forwards the scaled signal. Transmitters 1 and 2 send $(b_1, b_2 \oplus z_1^{(1)})$ and $(a_1, a_2 \oplus z_2^{(1)})$ respectively. Each receiver can then gather the two received signals to decode 3 bits. From this figure, one can see that it is not needed to send additional information on top of innovation in the second stage. Therefore, this scheme matches Alamouti-based amplify-and-forward scheme in the Gaussian channel.

2.9.2 Proof of Lemma 1

Codebook Generation: Fix a joint distribution $p(u)p(u_1|u)p(u_2|u)p(x_1|u_1, u)p(x_2|u_2, u)$. First generate $2^{N(R_{1c}+R_{2c})}$ independent codewords $u^N(i, j)$, $i \in \{1, \cdots, 2^{NR_{1c}}\}$, $j \in \{1, \cdots, 2^{NR_{2c}}\}$, according to $\prod_{i=1}^N p(u_i)$. For each codeword $u^N(i, j)$, encoder 1 generates $2^{NR_{1c}}$ independent codewords $u_1^N((i, j), k)$, $k \in \{1, \cdots, 2^{NR_{1c}}\}$, according to $\prod_{i=1}^N p(u_{1i}|u_i)$. Subsequently, for each pair of codewords $\left(u^N(i, j), u_1^N((i, j), k)\right)$, generate $2^{NR_{1p}}$ independent codewords $x_1^N((i, j), k, l)$, $l \in \{1, \cdots, 2^{NR_{1p}}\}$, according to $\prod_{i=1}^N p(x_{1i}|u_{1i}, u_i)$.

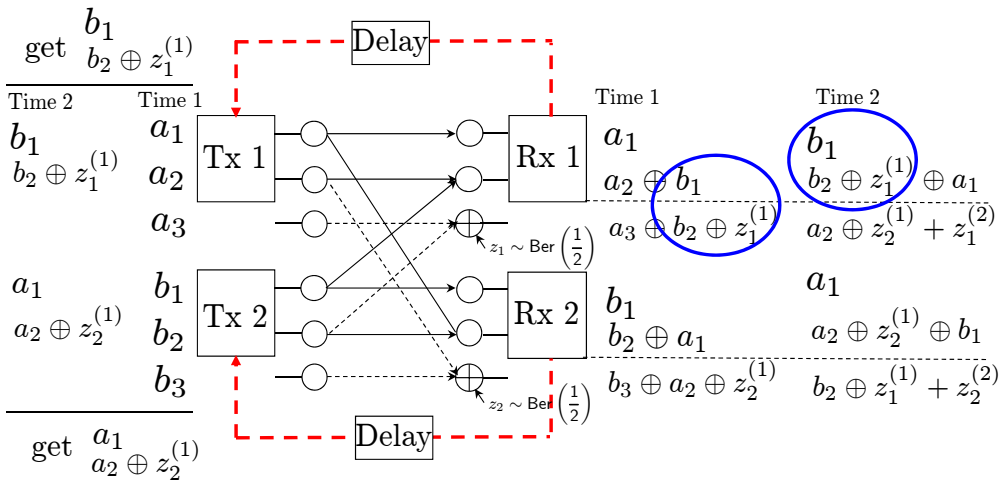


Figure 2.18: Intuition behind the Alamouti-based amplify-and-forward scheme.

Similarly, for each codeword $u^N(i,j)$, encoder 2 generates $2^{NR_{2c}}$ independent codewords $u_2^N((i,j),r)$, $r \in \{1,\cdots,2^{NR_{2c}}\}$, according to $\prod_{i=1}^N p(u_{2i}|u_i)$. For $\left(u^N(i,j), u_2^N((i,j),r)\right)$, generate $2^{NR_{2p}}$ independent codewords $x_2^N((i,j),r,s)$, $s \in \{1,\cdots,2^{NR_{2p}}\}$, according to $\prod_{i=1}^N p(x_{2i}|u_{2i},u_i)$.

Notation: Notations are independently used only for this section. The index k indicates the common message of user 1 instead of user index. The index i is used for both purposes: (1) indicating the previous common message of user 1; (2) indicating time index. It could be easily differentiated from contexts.

Encoding and Decoding: We employ block Markov encoding with a total size B of blocks. Focus on the bth block transmission. With feedback $y_1^{N,(b-1)}$, transmitter 1 tries to decode the message $\hat{w}_{2c}^{(b-1)} = \hat{k}$ (sent from transmitter 2 in the $(b-1)$th block). In other words, we find the unique $\hat{k}$ such that

$$\left(u^N\left(w_{1c}^{(b-2)}, \hat{w}_{2c}^{(b-2)}\right), u_1^N\left((w_{1c}^{(b-2)}, \hat{w}_{2c}^{(b-2)}), w_{1c}^{(b-1)}\right),\right.$$
$$\left. x_1^N\left((w_{1c}^{(b-2)}, \hat{w}_{2c}^{(b-2)}), w_{1c}^{(b-1)}, w_{1p}^{(b-1)}\right), u_2^N\left((w_{1c}^{(b-2)}, \hat{w}_{2c}^{(b-2)}), \hat{k}\right), y_1^{N,(b-1)}\right) \in A_\epsilon^{(N)},$$

where $A_\epsilon^{(N)}$ indicates the set of jointly typical sequences. Note that transmitter 1 already knows its own messages $(w_{1c}^{(b-2)}, w_{1c}^{(b-1)}, w_{1p}^{(b-1)})$. We assume that $\hat{w}_{2c}^{(b-2)}$ is correctly decoded from the previous block $(b-1)$. The decoding error occurs if one of two events happens: (1) there is no typical sequence; (2) there is another $\hat{w}_{2c}^{(b-1)}$ such that it is a typical sequence. By AEP, the first error probability becomes negligible as N goes to infinity. By [19], the second error probability becomes arbitrarily small (as N goes to infinity) if

$$R_{2c} \le I(U_2; Y_1|X_1, U). \tag{2.50}$$

Based on $(w_{1c}^{(b-1)}, \hat{w}_{2c}^{(b-1)})$, transmitter 1 generates a new common message $w_{1c}^{(b)}$ and a private message $w_{1p}^{(b)}$. It then sends $x_1^N\left((w_{1c}^{(b-1)}, \hat{w}_{2c}^{(b-1)}), w_{1c}^{(b)}, w_{1p}^{(b)}\right)$. Similarly transmitter 2 decodes $\hat{w}_{1c}^{(b-1)}$, generates $(w_{2c}^{(b)}, w_{2p}^{(b)})$ and then sends $x_2^N\left((\hat{w}_{1c}^{(b-1)}, w_{2c}^{(b-1)}), w_{2c}^{(b)}, w_{2p}^{(b)}\right)$.

Each receiver waits until total B blocks have been received and then does *backward decoding*. Notice that a block index b starts from the last B and ends to 1. For block b, receiver 1 finds the unique triple $(\hat{i}, \hat{j}, \hat{k})$ such that

$$\left(u^N\left(\hat{i}, \hat{j}\right), u_1^N\left((\hat{i}, \hat{j}), \hat{w}_{1c}^{(b)}\right), x_1^N\left((\hat{i}, \hat{j}), \hat{w}_{1c}^{(b)}, \hat{k}\right), u_2^N\left((\hat{i}, \hat{j}), \hat{w}_{2c}^{(b)}\right), y_1^{N,(b)}\right) \in A_\epsilon^{(N)},$$

where we assumed that a pair of messages $(\hat{w}_{1c}^{(b)}, \hat{w}_{2c}^{(b)})$ was successively decoded from block $(b+1)$. Similarly receiver 2 decodes $(\hat{w}_{1c}^{(b-1)}, \hat{w}_{2c}^{(b-1)}, \hat{w}_{2p}^{(b)})$.

Error Probability: By symmetry, we consider the probability of error only for block b and for a pair of transmitter 1 and receiver 1. We assume that $(w_{1c}^{(b-1)}, w_{2c}^{(b-1)}, w_{1p}^{(b)}) = (1,1,1)$ was sent through block $(b-1)$ and block b; and there was no backward decoding error from block B to $(b+1)$, i.e., $(\hat{w}_{1c}^{(b)}, \hat{w}_{2c}^{(b)})$ are successfully decoded.

Define an event:

$$E_{ijk} = \left\{\left(u^N(i,j), u_1^N((i,j), \hat{w}_{1c}^{(b)}), x_1^N((i,j), \hat{w}_{1c}^{(b)}, k), u_2^N((i,j), \hat{w}_{2c}^{(b)}), y_1^{N,(b)}\right) \in A_\epsilon^{(N)}\right\}.$$

By AEP, the first type of error becomes negligible. Hence, we focus only on the second type of error. Using the union bound, we get

$$\begin{aligned}
&\Pr\left(\bigcup_{(i,j,k)\neq(1,1,1)} E_{ijk}\right) \leq \sum_{i\neq 1, j\neq 1, k\neq 1} \Pr(E_{ijk}) + \sum_{i\neq 1, j\neq 1, k=1} \Pr(E_{ij1}) + \sum_{i\neq 1, j=1, k\neq 1} \Pr(E_{i1k}) \\
&+ \sum_{i\neq 1, j=1, k=1} \Pr(E_{i11}) + \sum_{i=1, j\neq 1, k\neq 1} \Pr(E_{1jk}) + \sum_{i=1, j\neq 1, k=1} \Pr(E_{1j1}) + \sum_{i=1, j=1, k\neq 1} \Pr(E_{11k}) \\
&\leq 2^{N(R_{1c}+R_{2c}+R_{1p}-I(U,X_1,U_2;Y_1)+4\epsilon)} + 2^{N(R_{1c}+R_{2c}-I(U,X_1,U_2;Y_1)+4\epsilon)} \\
&+ 2^{N(R_{1c}+R_{1p}-I(U,X_1,U_2;Y_1)+4\epsilon)} + 2^{N(R_{1c}-I(U,X_1,U_2;Y_1)+4\epsilon)} + 2^{N(R_{2c}+R_{1p}-I(U,X_1,U_2;Y_1)+4\epsilon)} \\
&+ 2^{N(R_{2c}-I(U,X_1,U_2;Y_1)+4\epsilon)} + 2^{N(R_{1p}-I(X_1;Y_1|U,U_1,U_2)+4\epsilon)}.
\end{aligned} \tag{2.51}$$

From (2.50) and (2.51), we can say that the error probability can be made arbitrarily small if

$$\begin{cases} R_{2c} & \leq I(U_2; Y_1|X_1, U) \\ R_{1p} & \leq I(X_1; Y_1|U_1, U_2, U) \\ R_{1c} + R_{1p} + R_{2c} & \leq I(U, X_1, U_2; Y_1) \end{cases} \tag{2.52}$$

$$\begin{cases} R_{1c} & \leq I(U_1; Y_2|X_2, U) \\ R_{2p} & \leq I(X_2; Y_2|U_1, U_2, U) \\ R_{2c} + R_{2p} + R_{1c} & \leq I(U, X_2, U_1; Y_2). \end{cases} \tag{2.53}$$

Fourier-Motzkin Elimination: Applying Fourier-Motzkin elimination, we easily obtain the desired inequalities. There are several steps to remove R_{1p}, R_{2p}, R_{1c}, and R_{2c}, successively. First substitute $R_{1p} = R_1 - R_{1c}$ and $R_{2p} = R_2 - R_{2c}$ to get:

$$R_{2c} \le I(U_2; Y_1|X_1, U) \qquad := a_1 \tag{2.54}$$
$$R_1 - R_{1c} \le I(X_1; Y_1|U_1, U_2, U) \qquad := a_2 \tag{2.55}$$
$$R_1 + R_{2c} \le I(U, X_1, U_2; Y_1) \qquad := a_3 \tag{2.56}$$
$$R_{1c} \le I(U_1; Y_2|X_2, U) \qquad := b_1 \tag{2.57}$$
$$R_2 - R_{2c} \le I(X_2; Y_2|U_1, U_2, U) \qquad := b_2 \tag{2.58}$$
$$R_2 + R_{1c} \le I(U, X_2, U_1; Y_2) \qquad := b_3 \tag{2.59}$$
$$-R_{1c} \le 0 \tag{2.60}$$
$$-R_1 + R_{1c} \le 0 \tag{2.61}$$
$$-R_{2c} \le 0 \tag{2.62}$$
$$-R_2 + R_{2c} \le 0 \tag{2.63}$$

Categorize the above inequalities into the following three groups: (1) group 1 not containing R_{1c}; (2) group 2 containing *negative* R_{1c}; (3) group 3 containing *positive* R_{1c}. By adding each inequality from groups 2 and 3, we remove R_{1c}. Rearranging the inequalities with respect to R_{2c}, we get:

$$R_1 \le b_1 + a_2 \tag{2.64}$$
$$R_2 + R_1 \le b_5 + a_2 \tag{2.65}$$
$$-R_1 \le 0 \tag{2.66}$$
$$R_{2c} \le a_1 \tag{2.67}$$
$$R_1 + R_{2c} \le a_5 \tag{2.68}$$
$$-R_2 + R_{2c} \le 0 \tag{2.69}$$
$$R_2 - R_{2c} \le b_2 \tag{2.70}$$
$$-R_{2c} \le 0. \tag{2.71}$$

Adding each inequality from groups 2 and 3, we remove R_{2c} and finally obtain:

$$R_1 \le \min(a_5, b_1 + a_2) \tag{2.72}$$
$$R_2 \le \min(b_5, a_1 + b_2) \tag{2.73}$$
$$R_1 + R_2 \le \min(b_5 + a_2, a_5 + b_2). \tag{2.74}$$

2.9.3 Converse Proof of Theorem 5

For completeness, we provide the detailed proof, although there are many overlaps with the proof in Theorem 3. The main point of the converse is how to introduce an auxiliary random

variable U which satisfies that given U_i, X_{1i} is conditionally independent of X_{2i}. Claim 4 gives hint into this. It gives the choice of $U_i := (V_1^{i-1}, V_2^{i-1})$.

First we consider the upper bound of an individual rate.

$$NR_1 = H(W_1) \overset{(a)}{\leq} I(W_1; Y_1^N) + N\epsilon_N \overset{(b)}{\leq} \sum H(Y_{1i}) + N\epsilon_N$$

where (a) follows from Fano's inequality and (b) follows from the fact that entropy is non-negative and conditioning reduces entropy.

Now consider the second bound.

$$\begin{aligned}
NR_1 &= H(W_1) = H(W_1|W_2) \\
&\leq I(W_1; Y_1^N|W_2) + N\epsilon_N \leq I(W_1; Y_1^N, Y_2^N|W_2) + N\epsilon_N \\
&\overset{(a)}{=} \sum H(Y_{1i}, Y_{2i}|W_2, Y_1^{i-1}, Y_2^{i-1}) + N\epsilon_N \\
&\overset{(b)}{=} \sum H(Y_{1i}, Y_{2i}|W_2, Y_1^{i-1}, Y_2^{i-1}, X_2^i) + N\epsilon_N \\
&\overset{(c)}{=} \sum H(Y_{2i}|W_2, Y_1^{i-1}, Y_2^{i-1}, X_2^i) \\
&\quad + \sum H(Y_{1i}|W_2, Y_1^{i-1}, Y_2^{i-1}, X_2^i, Y_{2i}, V_1^i) + N\epsilon_N \\
&\overset{(d)}{\leq} \sum [H(Y_{2i}|X_{2i}, U_i) + H(Y_{1i}|V_{1i}, V_{2i}, U_i)] + N\epsilon_N
\end{aligned}$$

where (a) follows from the fact that (Y_1^N, Y_2^N) is a function of (W_1, W_2); (b) follows from the fact that X_2^i is a function of (W_2, Y_2^{i-1}); (c) follows from the fact that V_1^i is a function of (Y_2^i, X_2^i); (d) follows from the fact that V_1^{i-1} is a function of (Y_2^{i-1}, X_2^{i-1}), $V_2^{i-1}$is a function of X_2^{i-1}, and conditioning reduces entropy. Similarly we get the outer bound for R_2.

The sum rate bound is given as follows.

$$\begin{aligned}
N(R_1 + R_2) &= H(W_1) + H(W_2) = H(W_1|W_2) + H(W_2) \\
&\leq I(W_1; Y_1^N|W_2) + I(W_2; Y_2^N) + N\epsilon_N \\
&= H(Y_1^N|W_2) + I(W_2; Y_2^N) + N\epsilon_N \\
&= H(Y_1^N|W_2) + H(Y_2^N) \\
&\quad - \left\{H(Y_1^N, Y_2^N|W_2) - H(Y_1^N|Y_2^N, W_2)\right\} + N\epsilon_N \\
&= H(Y_1^N|Y_2^N, W_2) - H(Y_2^N|Y_1^N, W_2) + H(Y_2^N) + N\epsilon_N \\
&\overset{(a)}{=} \sum H(Y_{1i}|Y_1^{i-1}, Y_2^N, W_2, X_2^i, V_1^i) + H(Y_2^N) + N\epsilon_N \\
&\overset{(b)}{\leq} \sum [H(Y_{1i}|V_{1i}, V_{2i}, U_i) + H(Y_{2i})] + N\epsilon_N
\end{aligned}$$

where (a) follows from the fact that X_2^i is a function of (W_2, Y_2^{i-1}) and V_1^i is a function of (X_2^i, Y_2^i); (b) follows from the fact that V_2^i is a function of X_2^i and conditioning reduces

entropy. Similarly, we get the other outer bound:

$$N(R_1 + R_2) \le \sum [H(Y_{2i}|V_{1i}, V_{2i}, U_i) + H(Y_{1i})] + N\epsilon_N.$$

Now let a time index Q be a random variable uniformly distributed over the set $\{1, 2, \cdots, N\}$ and independent of $(W_1, W_2, X_1^N, X_2^N, Y_1^N, Y_2^N)$. We define

$$\begin{aligned} &X_1 = X_{1Q},\ V_1 = V_{1Q};\ X_2 = X_{2Q},\ V_2 = V_{1Q}, \\ &Y_1 = Y_{1Q},\ Y_2 = Y_{2Q};\ U = (U_Q, Q). \end{aligned} \tag{2.75}$$

If (R_1, R_2) is achievable, then $\epsilon_N \to 0$ as $N \to \infty$. By Claim 4, an input joint distribution satisfies $p(u, x_1, x_2) = p(u)p(x_1|u)p(x_2|u)$. This establishes the converse.

Claim 4. *Given $U_i = (V_1^{i-1}, V_2^{i-1})$, X_{1i} and X_{2i} are conditionally independent.*

Proof. The proof is based on the dependence-balance-bound technique in [68, 33]. For completeness we describe details. First we show that $I(W_1; W_2|U_i) = 0$, which implies that W_1 and W_2 are independent given U_i. Based on this, we show that X_{1i} and X_{2i} are conditionally independent given U_i.

Consider

$$\begin{aligned} 0 &\le I(W_1; W_2|U_i) \overset{(a)}{=} I(W_1; W_2|U_i) - I(W_1; W_2) \\ &\overset{(b)}{=} -H(W_1) - H(W_2) - H(U_i) + H(W_1, W_2) + H(W_1, U_i) + H(W_2, U_i) - H(W_1, W_2, U_i) \\ &\overset{(c)}{=} -H(U_i) + H(U_i|W_1) + H(U_i|W_2) \\ &= \sum_{j=1}^{i-1} \left[-H(V_{1j}, V_{2j}|V_1^{j-1}, V_2^{j-1}) + H(V_{1j}, V_{2j}|W_1, V_1^{j-1}, V_2^{j-1}) + H(V_{1j}, V_{2j}|W_2, V_1^{j-1}, V_2^{j-1})\right] \\ &\overset{(d)}{=} \sum_{j=1}^{i-1} \left[-H(V_{1j}, V_{2j}|V_1^{j-1}, V_2^{j-1}) + H(V_{2j}|W_1, V_1^{j}, V_2^{j-1}) + H(V_{1j}|W_2, V_1^{j-1}, V_2^{j})\right] \\ &= \sum_{j=1}^{i-1} \left[-H(V_{1j}|V_1^{j-1}, V_2^{j-1}) + H(V_{1j}|W_2, V_1^{j-1}, V_2^{j}) - H(V_{2j}|V_1^{j}, V_2^{j-1}) + H(V_{2j}|W_1, V_1^{j}, V_2^{j-1})\right] \\ &\overset{(e)}{\le} 0 \end{aligned}$$

where (a) follows from $I(W_1; W_2) = 0$; (b) follows from the chain rule; (c) follows from the chain rule and $H(U_i|W_1, W_2) = 0$; (d) follows from the fact that V_1^j is a function of (W_1, V_2^{j-1}) and V_2^j is a function of (W_2, V_1^{j-1}) (see Claim 5); (e) follows from the fact that conditioning reduces entropy. Therefore, $I(W_1; W_2|U_i) = 0$, which shows the independence of W_1 and W_2 given U_i.

Notice that X_{1i} is a function of (W_1, V_2^{i-1}) and X_{2i} is a function of (W_2, V_1^{i-1}) (see Claim 5). Hence, it follows easily that

$$I(X_{1i}; X_{2i}|U_i) = I(X_{1i}; X_{2i}|V_1^{i-1}, V_2^{i-1}) = 0, \tag{2.76}$$

which proves the independence of X_{1i} and X_{2i} given U_i. □

Claim 5. *For $i \geq 1$, X_1^i is a function of (W_1, V_2^{i-1}). Similarly, X_2^i is a function of (W_2, V_1^{i-1}).*

Proof. By symmetry, it is enough to prove it only for X_1^i. Since the channel is deterministic (noiseless), X_1^i is a function of (W_1, W_2). In Fig. 2.10, we see that information of W_2 to the first link pair must pass through V_{2i}. Also note that X_{1i} depends on the past output sequences until $i-1$ (due to feedback delay). Therefore, X_1^i is a function of (W_1, V_2^{i-1}). □

2.9.4 Proof of Lemma 2

Let $\mathsf{INR} = \mathsf{SNR}^\alpha$. Then, by (29) in [41] and (77*) in [42], we get

$$R_{\mathsf{sym}} = \log\left(\frac{1 + \mathsf{SNR} + \mathsf{SNR}^\alpha + 2\rho^*\mathsf{SNR}^{\frac{\alpha+1}{2}}}{1 + (1-\rho^{*2})\mathsf{SNR}^\alpha}\right), \tag{2.77}$$

where ρ^* is the solution between 0 and 1 such that

$$\begin{aligned} &2\mathsf{SNR}^{\frac{3\alpha+1}{2}}\rho^{*4} + \mathsf{SNR}^\alpha\rho^{*3} - 4(\mathsf{SNR}^{\frac{3\alpha+1}{2}} + \mathsf{SNR}^{\frac{\alpha+1}{2}})\rho^{*2} \\ &- (2 + \mathsf{SNR} + 2\mathsf{SNR}^\alpha)\rho^* + 2(\mathsf{SNR}^{\frac{3\alpha+1}{2}} + \mathsf{SNR}^{\frac{\alpha+1}{2}}) = 0. \end{aligned}$$

Notice that for $0 \leq \alpha \leq \frac{1}{3}$ and for the high SNR regime, SNR is a dominant term and $0 < \rho^* < 1$. Hence, we get $\rho^* \approx 2\mathsf{SNR}^{\frac{3\alpha-1}{2}}$. This gives $\lim_{\mathsf{SNR}\to\infty} \frac{R_{\mathsf{sym}}}{\log(\mathsf{SNR})} = 1 - \alpha$. For $\frac{1}{3} < \alpha < 1$, the first and second dominant terms become $\mathsf{SNR}^{\frac{3\alpha+1}{2}}$ and SNR respectively. Also for this regime, $\rho^* \approx 1$. Hence, we approximately get $1 - \rho^{*2} \approx \mathsf{SNR}^{\frac{-3\alpha+1}{4}}$. This gives $\lim_{\mathsf{SNR}\to\infty} \frac{R_{\mathsf{sym}}}{\log(\mathsf{SNR})} = \frac{3-\alpha}{4}$. For $\alpha \geq 1$, note that the first and second dominant terms are $\mathsf{SNR}^{\frac{3\alpha+1}{2}}$ and SNR; and ρ^* is very close to 1. So we get $1 - \rho^{*2} \approx \mathsf{SNR}^{-\frac{\alpha+1}{4}}$. This gives the desired result in the last case.

Chapter 3

Interference Alignment for Cellular Networks

3.1 Introduction

One of the key performance metrics in the design of cellular systems is that of cell-edge spectral efficiency. As a result, fourth-generation (4G) cellular systems, such as 3GPP-LTE [1] and WiMAX [2], require at least a doubling in cell-edge throughput over previous 3G systems [1]. Given the disparity between average and cell-edge spectral efficiencies (ratios of about 4:1) [2], the desire to improve cell-edge throughput performance is likely to continue.

Since the throughput of cell-edge users is greatly limited by the presence of co-channel interference from other cells, developing an intelligent interference management scheme is the key to improving cell-edge throughput. One interesting recent development, called *interference alignment* (IA) [44, 14], manages interference by aligning multiple interference signals in a signal subspace with dimension smaller than the number of interferers. While most of the work on IA [14, 31, 50] has focused on K point-to-point interfering links, in this work we show that that IA can be used to improve the cell-edge user throughput in a cellular network. Specifically we show that *near interference-free throughput* performance can be achieved in the cellular network.

While IA promises substantial theoretical gain in cellular networks, it comes with challenges in implementation. First, our IA scheme that will described in Section 3.2 requires extensive channel-state-information (CSI) to be exchanged over the backhaul *between base-stations (BSs) of different cells*. A second challenge comes from realistic cellular environments that involve multiple unaligned out-of-cell interferers. Lastly, the integration of IA with other system issues, such as scheduling, needs to be addressed.

We propose a new IA technique for downlink cellular systems that addresses many of these practical concerns. Unlike the uplink IA scenario, our downlink IA scheme requires feedback only within a cell. As a consequence, our technique can be implemented with small

changes to existing 4G standards where the within-a-cell feedback mechanism is already being considered for supporting multi-user MIMO.

This IA technique aims to cancel interference only from one neighboring BS, which does well in a two-cell layout. In particular, the IA technique in Section 3.2 gives up the opportunity of providing matched-filtered gain (also called beam-forming gain in the case of multiple antennas) in the presence of a large number of interferers. This new technique balances the two advantages of interference cancellation and matched-filtering gain, inspired by the idea of the standard MMSE receiver that unifies a zero-forcing receiver (optimal in the high SNR regime) and a matched filter (optimal in the low SNR regime). Through simulations, we show that this provides approximately 60% and 28% gain in cell-edge throughput performance for a linear cell layout and 19 hexagonal wrap-around-cell layout respectively, as compared to a standard multi-user MIMO technique. We also find that our scheme has the potential to provide significant performance for heterogeneous networks [4], e.g., macro-pico cellular networks where dominant interference can be much stronger than the residual interference. For instance, pico-users can be significantly interfered with by the nearby macro-BS, as compared to the aggregated remaining BSs. We show that for these networks our scheme can give around 40% to 200% gain over the standard technique. Furthermore, our scheme is easily combined with a widely-employed opportunistic scheduler [65] for significant multi-user-diversity gain.

Next, in order to mitigate the interference from multiple dominant interferers, we propose another IA scheme, which we call *subspace interference alignment*. This scheme aligns the interference of multiple interferers onto a restricted subspace *simultaneously* at multiple non-intended receivers, whose dimension is negligible as compared to that of the subspace spanned by the desired signals, thus achieving almost interference-free degrees-of-freedom even in the multiple (more than 2) cellular networks. A key property of this scheme is that the simultaneous interference alignment is achieved using only a *finite* number of dimensions. This is in stark contrast to Cadambe-Jafar's IA scheme which employs an *infinite* number of dimensions to achieve the simultaneous interference alignment.

3.2 Uplink Interference Alignment

System Model: We develop uplink IA in [60]. Fig. 3.1 illustrates an example for the case of two isolated cells α and β. Suppose that there are K users in each cell and each user (e.g., user k in cell α) sends one symbol (or stream) $x_{\alpha k} \in \mathbb{C}$ along a transmitted vector $\mathbf{v}_{\alpha k} \in \mathbb{C}^M$. We can generate multiple dimensions by using subcarriers (in an OFDM system), antennas, or both:

$$M = (\#\text{ of subcarriers}) \times (\#\text{ of transmit antennas}). \tag{3.1}$$

We avoid employing multiple time slots for creating dimensions. This is because the interference alignment technique (to be described shortly) requires knowledge of the CSI, but

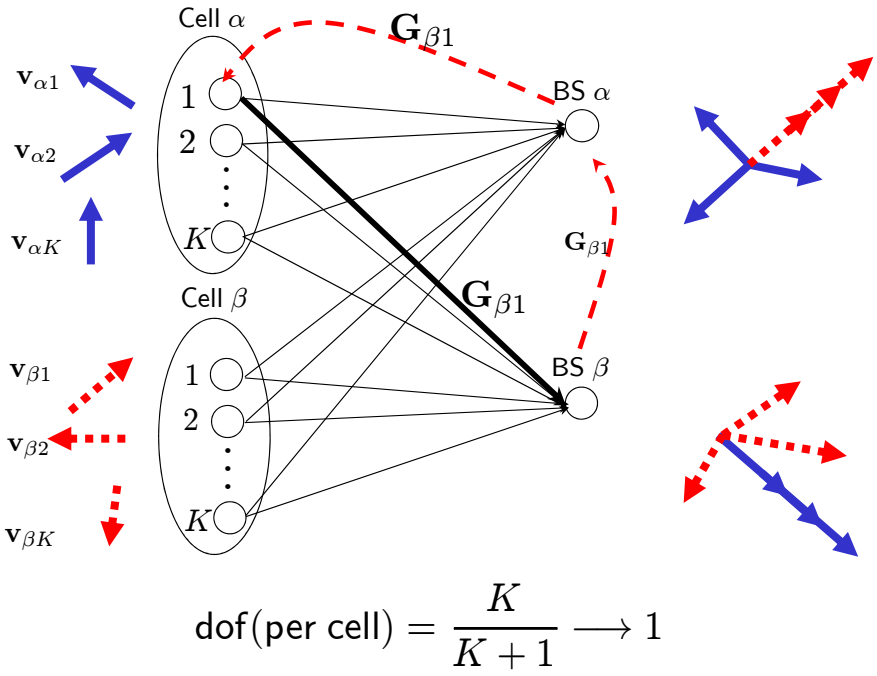


Figure 3.1: Uplink interference alignment. Interference-free degrees-of-freedom can be asymptotically achieved with an increase in K. While this scheme provides promising theoretical gain, it comes with implementation challenge. The scheme requires each user to know its *cross*-channel information to the other BS and this may require exchange of cross-channel information over the backhaul between BSs of different cells.

the future CSI is not available beforehand due to causality. Let S be the total number of streams. In this case, $S = K$, as all of the users are sending their own symbols.

The received signal of BS α is given by

$$\mathbf{y}_\alpha = \sum_{k=1}^{K} (\mathbf{H}_{\alpha k}\mathbf{v}_{\alpha k})x_{\alpha k} + \sum_{k=1}^{K} (\mathbf{G}_{\alpha k}\mathbf{v}_{\beta k})x_{\beta k} + \mathbf{z}_\alpha, \tag{3.2}$$

where $\mathbf{H}_{\alpha k} \in \mathbb{C}^{N\times M}$ indicates direct-channel from user k of cell α to BS α, and $\mathbf{G}_{\alpha k} \in \mathbb{C}^{N\times M}$ denotes cross-channel from user k of cell β to BS α. We assume that the channels are constant over a few time slots with respect to channel estimation and CSI feedback procedures. Here N is the number of dimensions at the receiver: $N =$ (# of subcarriers) $\times$ (# of receive antennas). We focus on the symmetric configuration, i.e., $M = N$. In fact, the extension to the asymmetric case is not straightforward, although we will provide a natural, but potentially suboptimal, variant of the IA scheme (to be described) in Section 3.4.4. We will discuss more details in Section 3.4.4. Note that the combined use of antennas and subcarriers induces a block-diagonal structure for the channel matrices. We assume that

noise is additive white Gaussian and without loss of generality assume that it has unit power, i.e., $\mathbf{z}_\alpha \sim \mathcal{CN}(0, \mathbf{I})$.

Description: The idea of interference alignment is to design the transmitted vectors so that they are aligned onto a one-dimensional linear subspace at the other BS. Specifically, user k in cell β sets its transmitted vector as $\mathbf{v}_{\beta k} = \mathbf{G}_{\alpha k}^{-1}\mathbf{v}_{\mathsf{ref}}$, where $\mathbf{v}_{\mathsf{ref}} \in \mathbb{C}^M$ is an arbitrary non-zero vector that can be fixed, independent of channel state information, e.g., $\mathbf{v}_{\mathsf{ref}} = [1, \cdots, 1]^t$, where $[\cdot]^t$ indicates a transpose. Similarly user k in cell α sets its transmitted vector as $\mathbf{v}_{\alpha k} = \mathbf{G}_{\beta k}^{-1}\mathbf{v}_{\mathsf{ref}}$. We use the same $\mathbf{v}_{\mathsf{ref}}$ across the cells, although it can be different. The received signal of BS α is then

$$\mathbf{y}_\alpha = \sum_{k=1}^{K}(\mathbf{H}_{\alpha k}\mathbf{G}_{\beta k}^{-1}\mathbf{v}_{\mathsf{ref}})x_{\alpha k} + \mathbf{v}_{\mathsf{ref}}\left(\sum_{k=1}^{K} x_{\beta k}\right) + \mathbf{z}_\alpha. \tag{3.3}$$

Notice that the interference space collapses to a one-dimensional linear subspace spanned by the $\mathbf{v}_{\mathsf{ref}}$. On the other hand, due to the randomness in wireless channels, the transmitted vectors associated with the desired symbols $x_{\alpha k}$'s are likely to be linearly independent. Note that for $M = K+1$, $\mathsf{rank}\left[\mathbf{H}_{\alpha 1}\mathbf{G}_{\beta 1}^{-1}\mathbf{v}_{\mathsf{ref}}, \cdots, \mathbf{H}_{\alpha K}\mathbf{G}_{\beta K}^{-1}\mathbf{v}_{\mathsf{ref}}\right] = K$, while the interference signals only occupy a one-dimensional subspace. Hence, the BS can recover K desired symbols using $K+1$ dimensions. Notice that this full rank condition holds with high probability under typical wireless channels and for the block-diagonal structure of the channel matrices. The performance in the interference-limited regime can be captured by a notion of degrees-of-freedom (dof). Here, the $\mathsf{dof\ per\ cell} = \frac{K}{K+1}$. We use the notion normalized by the total number $M = K+1$ of dimensions. Notice that as K gets large, we can asymptotically achieve interference-free $\mathsf{dof} = 1$.

While this IA technique provides promising theoretical gain, it comes with some implementation challenge. The IA scheme requires each user to know its *cross*-channel information to the other BS. While in a time-division-multiplexing system, channels can be estimated using reciprocity, in a frequency-division-multiplexing system, an implementation issue arises. One way to obtain the cross-channel is that the other-cell BS directly feeds back the cross-channel information to the users. However, this requires additional communication sessions between different cells, thus increasing the control channel overhead. Another way (possibly more plausible) is to exchange such channel knowledge over the backhaul between BSs of different cells. Fig. 3.1 shows a route to obtain the CSI of $\mathbf{G}_{\beta 1}$: $BS\ \beta \rightarrow backhaul \rightarrow BS\ \alpha \rightarrow feedback \rightarrow user\ 1\ of\ cell\ \alpha$. However, this requires the use of additional links (backhaul). On the contrary, in the downlink, we show that IA can be applied without inter-cell communication sessions or backhaul cooperation, thereby resolving this implementation issue.

3.3 Downlink Interference Alignment

Description: Fig. 3.2 illustrates an example of downlink IA where there are two users

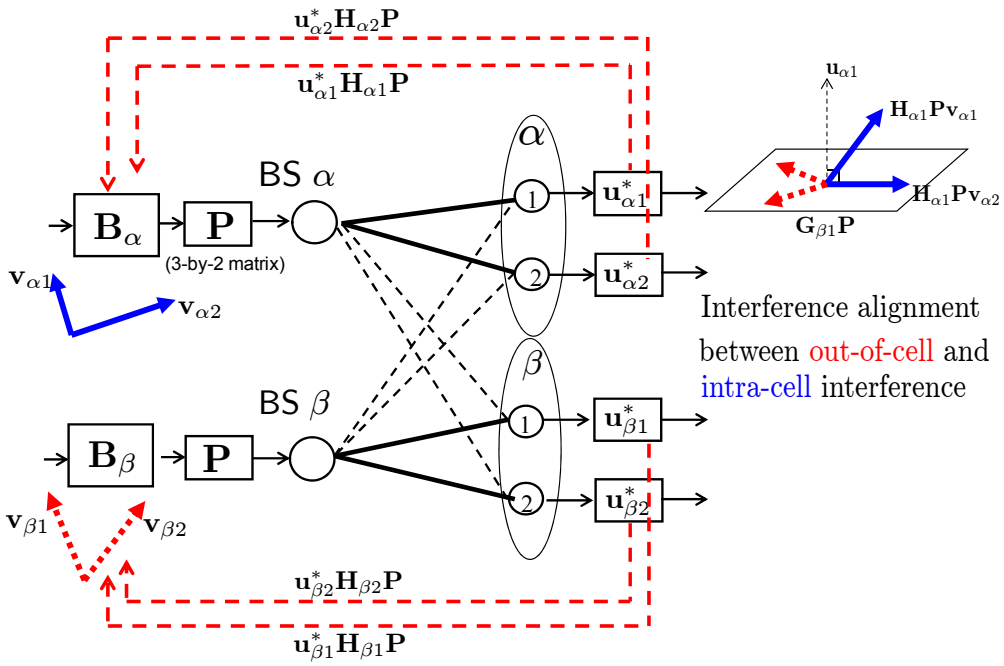


Figure 3.2: Downlink interference alignment. Interference alignment is achieved between out-of-cell and intra-cell interference vectors at multiple users at the same time. Unlike the uplink IA, our downlink IA scheme does not require backhaul cooperation or inter-cell communication sessions.

($K = 2$) in each cell. The uplink-downlink duality theorem [60, 67, 37] states that the dof of the uplink is the same as that of the downlink. Hence, in this example, the dof per cell $= \frac{K}{K+1} = \frac{2}{3}$. To achieve this, each BS needs to send two streams $S = 2$ over three dimensions $M = 3$. The idea is similar to that of the uplink IA in a sense that two dimensions are used for transmitting desired signals and the remaining one dimension is reserved for interference signals. However, the method of interference alignment is different.

While in the uplink, we set the reference vector $\mathbf{v}_{\mathsf{ref}}$ at receivers, in the downlink, we fix a M-by-S precoder matrix $\mathbf{P}$ at transmitters. Remember that $M = 3$ and $S = 2$ in this example. Notice that this fixed precoder is independent of channel gains. For simplicity we use the same precoder, although it can be different across cells. Each BS (e.g., BS α) has a second precoder $\mathbf{B}_{\alpha} = [\mathbf{v}_{\alpha 1}, \mathbf{v}_{\alpha 2}] \in \mathbb{C}^{2\times 2}$, which precedes the fixed precoder. Using these two cascaded precoders, it sends two symbols $(x_{\alpha 1}, x_{\alpha 2})$, each of which is intended for each user in the cell. The received signal of user k in cell α is then given by

$$\mathbf{y}_{\alpha k} = \mathbf{H}_{\alpha k}\mathbf{P}(\mathbf{v}_{\alpha 1}x_{\alpha 1} + \mathbf{v}_{\alpha 2}x_{\alpha 2}) + \underbrace{\mathbf{G}_{\beta k}\mathbf{P}\sum_{k=1}^{2}\mathbf{v}_{\beta k}x_{\beta k}}_{\text{out-of-cell interference}} + \mathbf{z}_{\alpha k},$$

where $\mathbf{H}_{\alpha k} \in \mathbb{C}^{3\times 3}$ indicates the direct-channel from BS α to user k of cell α, and $\mathbf{G}_{\beta k} \in \mathbb{C}^{3\times 3}$ denotes the cross-channel from BS β. With a minor abuse of notation, we use the same notation as we did in the uplink. We assume that $\mathbf{z}_{\alpha k} \sim \mathcal{CN}(0, \mathbf{I})$.

Next user k in cell α estimates the interference $\mathbf{G}_{\beta k}\mathbf{P}$ using pilots or a preamble. It then generates a null vector $\mathbf{u}_{\alpha k}$ such that $\mathbf{u}_{\alpha k}^* \mathbf{G}_{\beta k}\mathbf{P} = 0$ (and $||\mathbf{u}_{\alpha k}|| = 1$). Since the $\mathbf{G}_{\beta k}\mathbf{P}$ is of dimension 3-by-2, such a vector $\mathbf{u}_{\alpha k}$ always exists, and when applied to the received signal, it will null out the out-of-cell interference: $\tilde{y}_{\alpha k} := \mathbf{u}_{\alpha k}^* \mathbf{y}_{\alpha k} = \mathbf{u}_{\alpha k}^* \mathbf{H}_{\alpha k}\mathbf{P}(\mathbf{v}_{\alpha 1} x_{\alpha 1} + \mathbf{v}_{\alpha 2} x_{\alpha 2}) + \tilde{z}_{\alpha k}$, where $\tilde{z}_{\alpha k} := \mathbf{u}_{\alpha k}^* \mathbf{z}_{\alpha k} \sim \mathcal{CN}(0, 1)$. Note that the receive vector $\mathbf{u}_{\alpha k}$ does not guarantee the cancellation of intra-cell interference intended for the other user in the same cell α. This is accomplished as follows. User k feeds back its equivalent channel $\mathbf{u}_{\alpha k}^* \mathbf{H}_{\alpha k}\mathbf{P}$ (obtained after applying the receive vector) to its own BS α. BS α then applies the following zero-forcing precoder $\mathbf{B}_\alpha$ (which precedes the fixed precoder $\mathbf{P}$):

$$\mathbf{B}_\alpha := [\mathbf{v}_{\alpha 1}, \mathbf{v}_{\alpha 2}] = \begin{bmatrix} \mathbf{u}_{\alpha 1}^* \mathbf{H}_{\alpha 1}\mathbf{P} \\ \mathbf{u}_{\alpha 2}^* \mathbf{H}_{\alpha 2}\mathbf{P} \end{bmatrix}^{-1} \begin{bmatrix} \gamma_1 & 0 \\ 0 & \gamma_2 \end{bmatrix} \in \mathbb{C}^{2\times 2},$$

where γ_k is a normalization factor for meeting the transmit power constraint. Considering user 1's received signal, this zero-forcing precoder guarantees that user 2's transmitted signal $\mathbf{H}_{\alpha 1}\mathbf{P}\mathbf{v}_{\alpha 2}$ lies in the interference space $\mathbf{G}_{\beta 1}\mathbf{P}$. Note that $\mathbf{u}_{\alpha 1}^*(\mathbf{H}_{\alpha 1}\mathbf{P}\mathbf{v}_{\alpha 2}) = 0$. This enables user 1 to recover its own signal. Similarly, user 2 can recover its signal and therefore BS α can send 2 symbols using 3 dimensions, thus achieving $\mathsf{dof\ per\ cell} = \frac{2}{3}$. In fact, a series of these operations enables interference alignment, as will be explained in Remark 11. Also this scheme makes use of zero-forcing receive vector. Hence, we call this scheme *zero-forcing IA*.

Remark 11 (**Interference Alignment Interpretation**). *Observing the interference plane of user 1 in cell α, we can see this scheme achieves interference alignment. Note that three interference vectors - two out-of-cell interference vectors and one intra-cell interference vector - are aligned onto a two-dimensional linear subspace. Interference alignment is achieved between out-of-cell and intra-cell interference signals. Without carefully designing the transmit-and-receive vector pairs, three interfering vectors span three dimensions in general. However, our IA technique enables us to constrain the interference within only two dimensions (not three), thus enabling us to transmit in one dimension interference-free.* ■

Remark 12 (**Feedback Mechanism**). *Note two key system aspects of the technique. First, unlike the uplink IA, the exchange of cross-channel information between BSs or between users in different cells is not needed. Each BS can fix precoder $\mathbf{P}$, independent of channel gains. Each user can then specify the null space orthogonal to the out-of-cell interference signal space. This enables the user to design a zero-forcing receive vector without knowing the interfering vectors that were actually transmitted. For example, user 1 in cell α can compute $\mathbf{u}_{\alpha 1}$ without knowing $\mathbf{B}_\beta\mathbf{P}$ (the interfering vectors actually transmitted). Each user then feeds back its equivalent channel $\mathbf{u}_{\alpha k}\mathbf{H}_{\alpha k}\mathbf{P}$ and the BS forms the zero-forcing transmit*

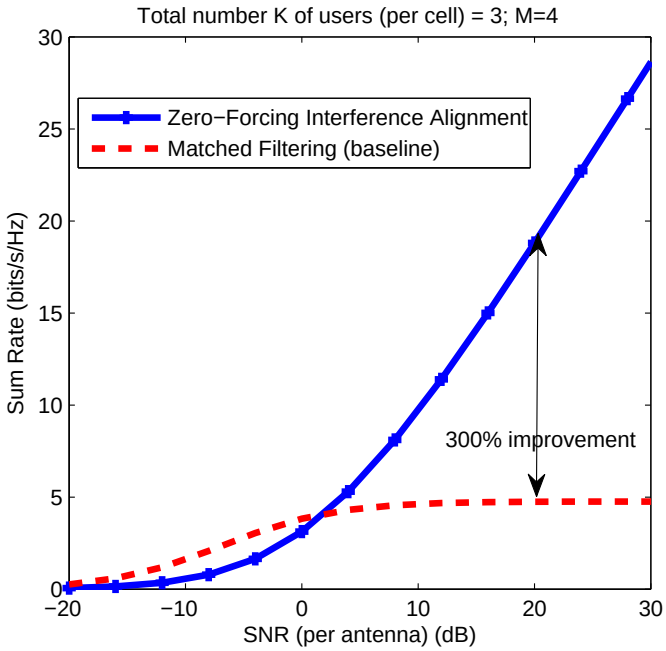


Figure 3.3: Performance of zero-forcing interference alignment for a two-isolated cell layout where $M = 4$ (e.g., a 4-by-4 antenna configuration), the number S of streams is $M - 1 = 3$ and the total number K of users in each cell is 3.

vectors only with the feedback of the equivalent channels. Hence, the scheme requires only within-a-cell feedback mechanism. This is contrast to the uplink IA which requires inter-cell communication sessions or backhaul cooperation between different BSs.

Secondly, while feedback is required from the user to the BS, this feedback is the same as the feedback used for standard multi-user MIMO techniques. The only difference is that in downlink IA, two cascaded precoders (e.g., $\mathbf{B}_\alpha$ and $\mathbf{P}$) are used and the receive vector of each user is chosen as a null vector of out-of-cell interference signal space. Therefore, the scheme can be implemented with little change to an existing cellular system supporting multi-user MIMO. ■

Performance and Limitations: Fig. 3.3 shows the sum-rate performance of zero-forcing IA in a two-isolated cell layout where $M = 4$ (e.g., a 4-by-4 antenna configuration), the number S of streams is $M - 1 = 3$ and the total number K of users in each cell is 3. As a baseline scheme, we use a *matched filter receiver*: one of the standard multi-user MIMO techniques [48, 30]. This baseline uses the dominant left-singular vector of the direct-channel as a receive vector:

$$\mathbf{u}_{\alpha k}^{\mathsf{MF}} = \text{a maximum left-singular vector of } \mathbf{H}_{\alpha k}. \tag{3.4}$$

Note that the matched filter receiver maximizes beam-forming gain while ignoring the interference signal space. We assume a zero-forcing vector at the transmitter to null out intra-cell interference. Nulling intra-cell interference is important as its power has the same order as the desired signal power. The zero-forcing transmit vectors are designed as:

$$[\mathbf{v}_{\alpha 1}^{\mathsf{ZF}}, \cdots, \mathbf{v}_{\alpha S}^{\mathsf{ZF}}] = \mathbf{H}^{*}(\mathbf{H}\mathbf{H}^{*})^{-1}\mathsf{diag}\left\{\gamma_1, \cdots, \gamma_S\right\} \in \mathbb{C}^{M \times S}, \tag{3.5}$$

where γ_k is a normalization factor and $\mathbf{H} := [\mathbf{u}_{\alpha 1}^{\mathsf{MF}*}\mathbf{H}_{\alpha k}; \cdots; \mathbf{u}_{\alpha S}^{\mathsf{MF}*}\mathbf{H}_{\alpha k}] \in \mathbb{C}^{S \times M}$ denotes the composite matrix.

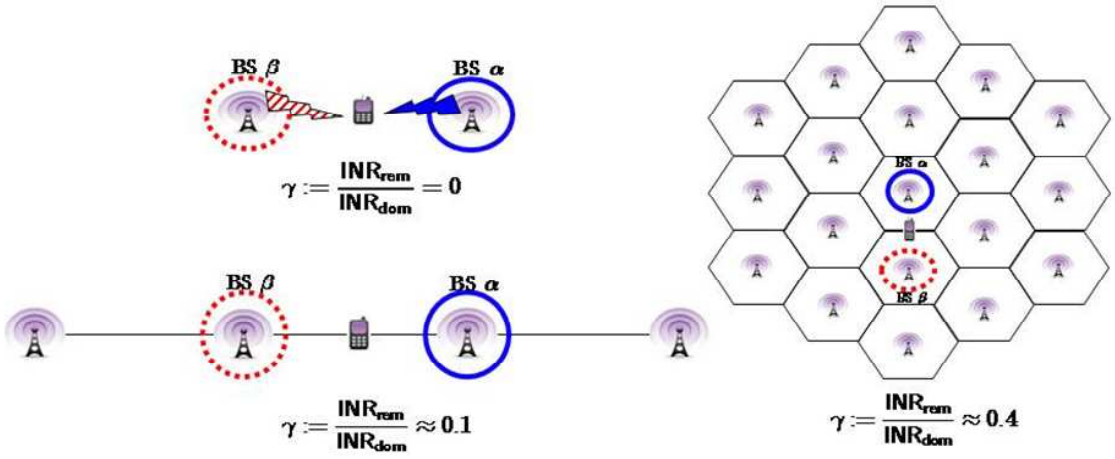


Figure 3.4: Different layouts in a downlink cellular system. A parameter γ indicates the relative strength of the interference power from a dominant interferer to the remaining interference power (summed from the other BSs).

Note that in (3.4), receiver vectors are initially chosen as dominant left-singular vectors of the channels, as transmit vectors are not decided yet. However, once transmit vectors are designed as above, we can now update the receiver vectors so as to maximize beam-forming gain by aligning them into the determined direction of the transmitted signals. Given the updated received vectors, we can also update the transmit vectors accordingly. This iterative algorithm was introduced in [48, 30] and we call this scheme *iterative matched filtering*.

While in matched filtering, this iterative procedure updates the receive-and-transmit vector pairs to potentially improve the performance, in the zero-forcing IA, it does not change the vector pairs. Recall that the receive vector in the IA scheme depends only on the interference space, so it is irrelevant to the transmit vectors. Hence, for fair comparison of CSI overhead, we assume no iteration for the matched filtering in Fig 3.3: the receive-and-transmit vectors are designed successively according to (3.4) and (3.5) without any iterations.

In Fig. 3.3, one can clearly see that the zero-forcing IA provides significant performance gain over the matched filtering. In fact, for large SNR, the scheme provides the asymptotically optimal performance, since it achieves the optimal dof [60]. The gain comes from the fact that in the two-isolated-cell case, there exists only a single interferer (no residual interferers) and our IA scheme completely removes the interference from the single interferer.

However, for realistic multi-cellular environments, the performance may not be very good due to the remaining interferers. In order to take multi-cellular environments into account, we introduce a parameter γ that captures the relative strength of the interference power from a dominant interferer to the remaining interference power (summed from the other BSs):

$$\gamma := \frac{\mathsf{INR}_{\mathsf{rem}}}{\mathsf{INR}_{\mathsf{dom}}}, \tag{3.6}$$

where $\mathsf{INR}_{\mathsf{dom}}$ and $\mathsf{INR}_{\mathsf{rem}}$ denote the ratios of the dominant and aggregate interference power over the noise power, respectively. Note that by adapting γ, one can cover arbitrary mobile location and cellular layouts.

While, at one extreme ($\gamma = 0$), the zero-forcing IA provides significant performance, at the other extreme ($\gamma \gg 1$), the scheme may not be good as it completely loses receive beam-forming gain. Remember that the zero-forcing IA receiver depends only on the interference space and therefore it is independent of the direct-channel, thus losing beam-forming gain. In this case, one can expect that matched filtering will perform much better than the IA scheme. This motivates the need for developing a new IA technique that can balance the degrees-of-freedom gain with the matched-filtered power gain depending on the value of γ.

3.4 MMSE-based Downlink Inteference Alignment

3.4.1 Scheme Description

The zero-forcing IA and matched filtering schemes are analogous to a conventional zero-forcing receiver and a matched-filter receiver in a point-to-point channel with colored noise. So it is natural to think of a unified technique like the standard MMSE receiver. However, in our cellular context, a straightforward design of an MMSE receiver requires the knowledge of transmitted vectors from the other cell. Moreover, a chicken-and-egg problem arises between different cells, due to the interconnection of the transmit-and-receive vector pairs. In order to *decouple* the vector design between cells, we consider uncoordinated systems, i.e., transmit vector information is not exchanged between different cells. Under this assumption, a goal is to mimic an MMSE receiver.

Idea: The idea for accomplishing this goal consists of three parts: (1) *coloring* an interference signal space, independent of the actually transmitted vectors; (2) designing a coloring parameter κ (to be defined shortly) to unify the two extreme cases: $\gamma \ll 1$ and $\gamma \gg 1$; (3) designing an MMSE-like receiver based on the coloring parameter.

Coloring the interference space: We employ two cascaded precoders: (1) a *fixed* precoder $\bar{\mathbf{P}} \in \mathbb{C}^{M\times M}$ located at the front-end; and (2) a zero-forcing precoder $\mathbf{B}_\alpha \in \mathbb{C}^{M\times S}$ which precedes the $\bar{\mathbf{P}}$. To differentiate with the precoder $\mathbf{P}$ used for the zero-forcing IA, we use different notation $\bar{\mathbf{P}}$. With this fixed precoder, we can color the interference space, to some extent, to be independent of the zero-forcing precoder. To see this, we first consider the covariance matrix of interference-plus-noise at user k in cell α:

$$\Phi_k = (1+\mathsf{INR}_{\mathsf{rem}})\mathbf{I} + \frac{\mathsf{SNR}}{S}(\mathbf{G}_{\beta k}\bar{\mathbf{P}}\mathbf{B}_\beta\mathbf{B}_\beta^*\bar{\mathbf{P}}^*\mathbf{G}_{\beta k}^*), \tag{3.7}$$

where S is the total number of streams ($S \le M$) and $\mathbf{B}_\beta$ indicates the zero-forcing precoder of a dominant interferer (BS β): $\mathbf{B}_\beta = [\mathbf{v}_{\beta 1}, \cdots, \mathbf{v}_{\beta S}] \in \mathbb{C}^{M\times S}$. Here we make several assumptions: noise power is normalized to 1 (without loss of generality); the total transmission power is equally allocated to each stream; and the aggregate interference except the dominant interference is white Gaussian. To be more accurate, we may include two or three dominant interferers in the process of computing Φ_k, assuming that the remaining interference except the multiple dominant interferers is white Gaussian. We will further discuss this issue in Section 3.4.4.

Since we consider uncoordinated systems, $\mathbf{B}_\beta$ is unknown to each user in cell α and therefore it is impossible to compute Φ_k. This motivates us to use the expected value of the covariance matrix averaged over $\mathbf{B}_\beta$: $\bar{\Phi}_k := \mathbb{E}[\Phi_k] = (1+\mathsf{INR}_{\mathsf{rem}})\mathbf{I} + \frac{\mathsf{SNR}}{S}(\mathbf{G}_{\alpha k}\bar{\mathbf{P}}\mathbb{E}[\mathbf{B}_\beta\mathbf{B}_\beta^*]\bar{\mathbf{P}}^*\mathbf{G}_{\alpha k}^*)$. Without the knowledge of $\mathbf{B}_\beta$, we can then control the coloredness of interference signals by carefully designing $\bar{\mathbf{P}}$. The idea is to differently weight the last $(M-S)$ columns of $\bar{\mathbf{P}}$ with a parameter κ ($0 \le \kappa \le 1$):

$$\bar{\mathbf{P}} = [\mathbf{f}_1, \cdots, \mathbf{f}_S, \kappa\mathbf{f}_{S+1}, \cdots, \kappa\mathbf{f}_M] \in \mathbb{C}^{M\times M}, \tag{3.8}$$

where $[\mathbf{f}_1, \cdots, \mathbf{f}_M]$ is an orthogonal matrix.

Before describing how to design κ and seeing how this colors interference signals, we will first explain how to compute $\mathbb{E}[\mathbf{B}_\beta\mathbf{B}_\beta^*]$ and how to decide the norm of each column vector of $\bar{\mathbf{P}}$ for meeting the transmit power constraint. In computing $\mathbb{E}[\mathbf{B}_\beta\mathbf{B}_\beta^*]$, we assume the statistics of $\mathbf{B}_\beta$. Since $\mathbf{B}_\beta$ is the zero-forcing precoder of BS β, it has the following form: $\mathbf{B}_\beta = \mathbf{H}^*\left(\mathbf{H}\mathbf{H}^*\right)^{-1}\mathsf{diag}\left\{\gamma_1, \cdots, \gamma_S\right\}$, where $\mathbf{H} := \left[\mathbf{u}_{\beta 1}^*\mathbf{H}_{\beta 1}\bar{\mathbf{P}}; \cdots; \mathbf{u}_{\beta S}^*\mathbf{H}_{\beta S}\bar{\mathbf{P}}\right]$. Note that $\mathbf{B}_\beta$ is coupled with $\bar{\mathbf{P}}$, so its statistics depend on κ. With a close observation of $\mathbf{B}_\beta$, one can see that the last $(M-S)$ rows of $\mathbf{B}_\beta$ is biased by a factor of κ. This motivates us to assume that each entry of $\mathbf{B}_\beta$ of the first S rows is i.i.d $\mathcal{CN}\left(0, \frac{1}{S+(M-S)\kappa^2}\right)$ and each entry of the last $(M-S)$ rows is i.i.d. $\mathcal{CN}\left(0, \frac{\kappa^2}{S+(M-S)\kappa^2}\right)$. Under this assumption, we can then compute $\mathbb{E}[\mathbf{B}_\beta\mathbf{B}_\beta^*]$ to get:

$$\bar{\Phi}_k := \mathbb{E}[\Phi_k] = (1+\mathsf{INR}_{\mathsf{rem}})\mathbf{I} + \frac{\mathsf{SNR}}{S+(M-S)\kappa^2}\left(\mathbf{G}_{\alpha k}\bar{\mathbf{P}}\begin{bmatrix}\mathbf{I}_S & \mathbf{0}\\ \mathbf{0} & \kappa^2\mathbf{I}_{M-S}\end{bmatrix}\bar{\mathbf{P}}^*\mathbf{G}_{\alpha k}^*\right). \tag{3.9}$$

Considering the transmit power constraint, we can now decide the norm of each column vector of $\bar{\mathbf{P}}$: $||\mathbf{f}_i||^2 = \frac{S+(M-S)\kappa^2}{S+(M-S)\kappa^4}, \forall i$. Note that this choice satisfies the transmit power constraint, i.e., $\mathsf{trace}\left[\bar{\mathbf{P}}\mathbb{E}[\mathbf{B}_\beta\mathbf{B}_\beta^*]\bar{\mathbf{P}}^*\right] = S$.

Designing a coloring parameter κ: We now present how to design the coloring parameter κ. Two extreme cases give insights into this design. When the residual interference is negligible, i.e., $\gamma \ll 1$, the scheme should mimic the zero-forcing IA, so $\bar{\mathbf{P}}$ should be rank-deficient, i.e., $\kappa = 0$. In this case, the null space of the interference signals can be specified, independent of $\mathbf{B}_\beta$. As a result, the *expected* covariance matrix acts as the *actual* covariance matrix, thus inducing the same solution as the zero-forcing IA. At the other extreme ($\gamma \gg 1$), the scheme should mimic matched filtering. This motivates us to choose a unitary matrix $\bar{\mathbf{P}}$ (i.e., $\kappa = 1$) so that the $\bar{\Phi}_k$ is close to $a\mathbf{I}$ for some scalar a. For an intermediate value of γ, we propose the following to sweep between the two cases:

$$\kappa = \min\left(\gamma^{1/4}, 1\right). \tag{3.10}$$

The use of the function $(\cdot)^{1/4}$ which relates γ to κ is our heuristic choice based on simulation results for some particular values of γ, SNR and other configurations. Specifically, for a 19 hexagonal cellular layout ($\gamma \approx 0.4$) and ($\mathsf{SNR} = 20$ dB$, M = 4, S = 3, K = 3$), we plot the sum-rate of the proposed scheme as a function of κ and then find κ that maximizes the sum-rate (via a grid search). From this experiment, we conjecture the relationship between γ and κ. We find that the function $(\cdot)^{1/4}$ well matches the relationship, thus proposing this heuristic. One may optimize κ in a more precise manner. For example, one may choose optimal κ case-by-case for each configuration and with a finer grid-step-size.

In the above choice, κ varies with mobile location, since $\mathsf{INR}_{\mathsf{rem}}$ is a function of mobile location. This can be undesirable because it requires frequent adaptation of BS precoder which supports users from the cell center to the cell edge. Therefore, *we propose to fix κ.* In the interests of improving the worst-case performance (cell-edge performance), we fix κ, based on the cell-edge mobile location for a given network layout. For example, we use $\kappa \approx 0.57$ for the linear cell layout and $\kappa \approx 0.80$ for the 19 hexagonal wrap-around cell layout (see Fig. 3.4). Since our choice focuses on improving the cell-edge throughput, less performance gain is expected for cell-interior users.

Alternatively, we can have different κ factors, depending on whether the BS is precoding for cell-edge vs. cell-interior users. This would require different $\bar{\mathbf{P}}$ matrices, and would add to the complexity of the system, but would optimize performance for all users in the cell.

Designing a MMSE-like receiver: With the above $\bar{\Phi}_k$, we then use the standard formula of an MMSE receiver: $\mathbf{u}_{\alpha k} = \frac{\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}}{||\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}||} \in \mathbb{C}^M$. Similar to the iterative matched filtering technique, we also employ an iterative approach to compute transmit-and-receive vector pairs.

Integration with a scheduler: We consider integration of our scheme with a scheduler: one of the important system issues that need to be considered in cellular systems. Designing the coloring parameter κ and controlling the number S of streams, our proposed scheme

balances the degrees-of-freedom gain (IA gain) with matched-filtered power gain depending on the value of γ. Importantly, scheduler gain is closely coupled with these gains, as it can play significant role in providing beam-forming power gain. For instance, an opportunistic scheduler [65] exploits multi-user diversity to provide good signal separation and power gain, thus inducing the high SINR regime where degrees-of-freedom gain affects the performance more significantly than beam-forming power gain does. Hence, it is important to carefully design the scheme considering the integration with a scheduler, so as to well balance the degrees-of-freedom gain and power gain.

In this work, we employ an opportunistic scheduler [65], which chooses a set of S users out of total K users such that the sum rate is maximized. We consider uncoordinated schedulers, i.e., scheduling information is not exchanged between different BSs.

Algorithm Description: We now describe an algorithm of the proposed IA scheme incorporating an opportunistic scheduler. Here is the algorithm.

1. (*Intialization*): Each user initializes its receive vector as follows: $\forall k \in \{1, \cdots, K\}$,

$$\mathbf{u}_{\alpha k}^{(0)} = \frac{\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}^{(0)}}{||\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}^{(0)}||} \in \mathbb{C}^M, \tag{3.11}$$

   where we set $\mathbf{v}_{\alpha k}^{(0)}$ as a maximum eigenvector of $\bar{\mathbf{P}}^*\mathbf{H}_{\alpha k}^*\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}$ to initially maximize beam-forming gain. Each user then feeds back the equivalent channel $\mathbf{u}_{\alpha k}^{(0)*}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}$ to its own BS.

2. (*Designing Transmit Vectors*): Fix a set $A \in \mathcal{K}$ where $\mathcal{K}$ is a collection of subsets of $\{1, \cdots, K\}$. As for the elements in $\mathcal{K}$, we consider all of the possible candidates that have cardinality S, i.e., $|\mathcal{K}| = \binom{K}{S}$. For the given A, with the feedback information, the BS computes zero-forcing transmit vectors

$$\mathbf{B}_\alpha := [\mathbf{v}_{\alpha k_1}^{(1)}, \cdots, \mathbf{v}_{\alpha k_S}^{(1)}] = \mathbf{H}^{(1)*}(\mathbf{H}^{(1)}\mathbf{H}^{(1)*})^{-1}\mathsf{diag}\left\{\gamma_1^{(1)}, \cdots, \gamma_S^{(1)}\right\} \in \mathbb{C}^{M\times S},$$

   where $k_l \in A$, $\gamma_l^{(1)}$ is a normalization factor, and $\mathbf{H}^{(1)} := [\mathbf{u}_{\alpha k_1}^{(0)*}\mathbf{H}_{\alpha k_1}\bar{\mathbf{P}}; \cdots; \mathbf{u}_{\alpha k_S}^{(0)*}\mathbf{H}_{\alpha k_S}\bar{\mathbf{P}}] \in \mathbb{C}^{S\times M}$. Remember that the fixed precoder $\bar{\mathbf{P}}$ is designed so that each column vector of $\mathbb{E}[\bar{\mathbf{P}}\mathbf{B}_\alpha]$ is normalized. So $\bar{\mathbf{P}}\mathbf{B}_\alpha$ is not guaranteed to be normalized. Hence, the BS re-normalizes $\bar{\mathbf{P}}\mathbf{B}_\alpha$ with $\tilde{\gamma}_l^{(1)}$ so that each column vector of $\bar{\mathbf{P}}\mathbf{B}_\alpha\mathsf{diag}\left\{\tilde{\gamma}_1^{(1)}, \cdots, \tilde{\gamma}_S^{(1)}\right\}$ is normalized.

3. (*Opportunistic Scheduling*): The BS finds A^* such that

$$A^* = \arg\max_{A\in\mathcal{K}} \sum_{k\in A} \log\left(1 + \frac{\frac{\mathsf{SNR}}{S}||\tilde{\gamma}_k^{(1)}\mathbf{u}_{\alpha k}^{(0)*}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}^{(1)}||^2}{1 + \mathsf{INR}_{\mathsf{rem}}}\right).$$

4. (*Iteration*): For the A^*, we iterate the following. The BS informs each user of $\mathbf{v}_{\alpha k}^{(i)}$ via precoded pilots. Each user updates the receive vector as follows:

$$\mathbf{u}_{\alpha k}^{(i)} = \frac{\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}^{(i)}}{||\bar{\Phi}_k^{-1}\mathbf{H}_{\alpha k}\bar{\mathbf{P}}\mathbf{v}_{\alpha k}^{(i)}||} \in \mathbb{C}^M, \quad k \in A^*.$$

   Each user then feeds back the updated equivalent channel to its own BS. With this feedback information, the BS computes zero-forcing transmit vectors $\mathbf{v}_{\alpha k}^{(i+1)}$.

Remark 13. *Although users can see out-of-cell interference, the scheduler at BS cannot compute it without some side-information from the users. Hence, we assume that the scheduler makes a decision assuming no dominant interference. Note that the denominator inside the logarithmic term contains only noise and residual interference. To reduce CSI overhead, we assume that a scheduler decision is made before the iteration step.* ■

In practice, we may prefer not to iterate, since it requires more feedback information. Note that the feedback overhead is exactly the same as that of iterative matched-filtering (baseline). The only difference is that we use the fixed precoder $\bar{\mathbf{P}}$ and the MMSE-like receiver employing $\bar{\Phi}_k$. *This requires little change to an existing cellular system supporting multi-user MIMO.*

3.4.2 Performance Evaluation: Simulation Results

Setup: Through simulations, we evaluate the performance of the proposed scheme for downlink cellular systems. We consider one of the possible antenna configurations in the 4G standards [2, 1]: 4 transmit and 4 receive antennas. To minimize the change to the existing 4G systems, we consider using only antennas for the multiple dimensions, i.e., $M = 4$. We focus on three different cellular layouts, illustrated in Fig. 3.4.

In the interests of improving the worst-case throughput performance, we consider a cell-edge mobile location. Specifically, we assume that all of the K users in each cell are placed at the mid-point between two adjacent cells. This simulation setup can reflect the scenarios where user locations, once chosen, are almost static, e.g., working places located in the cell-edge. On the other hand, one may be interested in simulating per-user throughput distribution assuming different user locations, so as to evaluate the system-wide benefits of the proposed scheme. In this case, we expect less performance gain of our proposed IA scheme, as it considers a single γ and the corresponding κ, which are based on cell-edge users. Evaluating this system-level performance more precisely is beyond the scope of this work, but eventually this needs to be considered as future work.

We use the standard ITU-Ped path-loss model, with i.i.d. Rayleigh fading components for each of the antenna. We assume that inter-BS distance is 1 km and path-loss exponent is 3.76. As for an interference model, we exactly model the interference of the neighboring BS (the dominant interferer), while assuming that the aggregated interference of the remaining

BSs is white Gaussian. This white Gaussian assumption on the residual interference provides the lower bound of the performance of all the schemes we will consider shortly. This is because each of the techniques can exploit the knowledge of interference, and the white interference is a worst case assumption.

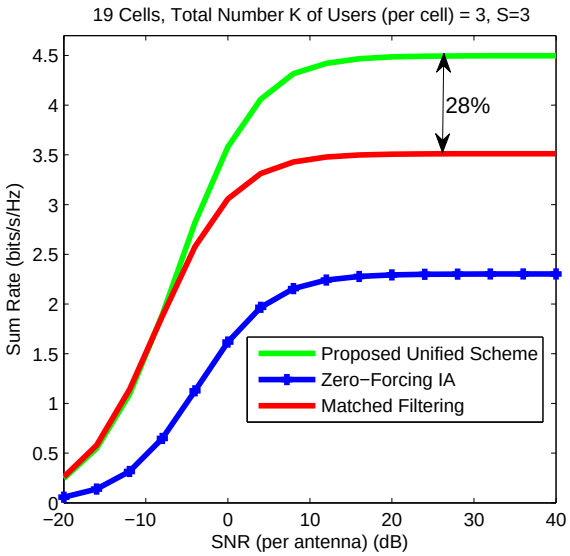
Performance: Fig. 3.5 shows the sum-rate performance for a 19 hexagonal cellular layout where $\gamma \approx 0.4$. We assume that total number K of users in each cell is 3 and consider the number $S = 3$ of streams. Note that the zero-forcing IA is worse than the matched filtering (baseline). This implies that when $\gamma \approx 0.4$ (residual interference is not negligible), boosting power gain gives better performance than mitigating dominant out-of-cell interference. However, the proposed unified IA technique outperforms both of them for all regimes. It gives approximately 28% throughput gain when $\mathsf{SNR} = 20$ dB.

We also investigate the convergence of the proposed scheme. Note in Fig. 3.5(b) that the proposed scheme converges to the limits very fast, i.e., even one iteration is enough to derive most of the asymptotic performance gain. This means that additional iterations provide marginal gain, while requiring a larger overhead of CSI feedback. Another observation is that the converged limits of the proposed technique is invariant to the initial values of transmit-and-receive vectors. Note that random initialization induces the same limits as that of our carefully chosen initial values, but it requires more iterations to achieve the limits. Therefore, the initial values need to be carefully chosen to minimize the overhead of CSI feedback. Through simulations, we have observed the same convergence behavior in many other scenarios (different cellular layouts and different K, M and S), although it is not proved here. So we conjecture that this convergence behavior occurs in general.

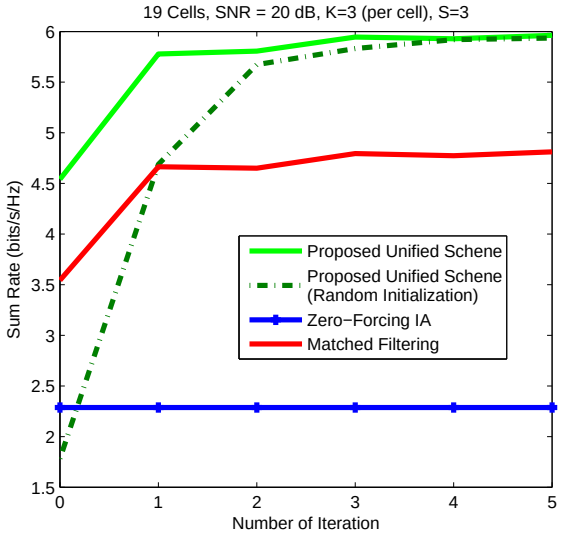
Fig. 3.6 shows the sum-rate performance when considering a scheduler. We assume that $K = 10$ and consider an opportunistic scheduler. In fact, the number S of streams is related to the scheduling effect. For a large value of K, the opportunistic scheduler provides good signal separation and power gain, thereby inducing the high SINR regime where multiplexing gain is more significant than the beamforming power gain. In this case, using more streams provides better performance. We find through simulations that using three streams provides the best performance for a practical number of users per cell (around 10). Hence, we consider $S = 3$. The sum-rate reflects the 3 cell-edge users who are chosen at a time out of 10 via the scheduler.

As shown in Fig. 3.6 (a), as compared to the non-scheduler case, the performance of zero-forcing IA is significantly improved, although it is still worse than matched filtering. Zero-forcing IA can now achieve power gain with the scheduler. Notice that the power gain due to the scheduler is significant, thus making the additional matched-filter power gain marginal. Our proposed scheme still outperforms both schemes, providing approximately 28% over the matched filtering.

Fig. 3.6 (b) shows the sum-rate performance for a linear cellular layout where $\gamma \approx 0.1$. In this case, the residual interference is reduced to $\gamma \approx 0.1$, so mitigating dominant out-of-cell interference improves the performance more significantly than beam-forming does. The gain of the proposed scheme is significant, i.e., approximately 60% in the high SNR regime of

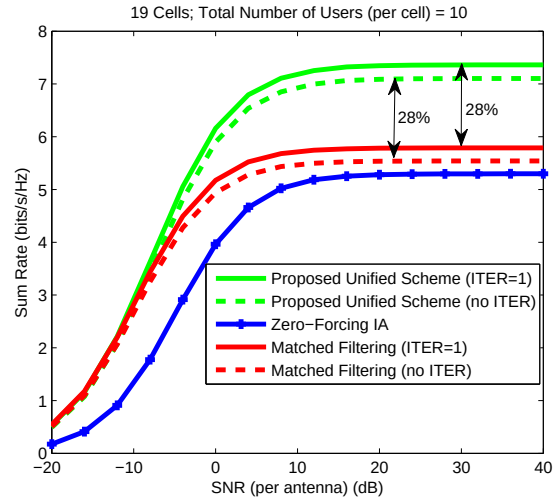


(a)

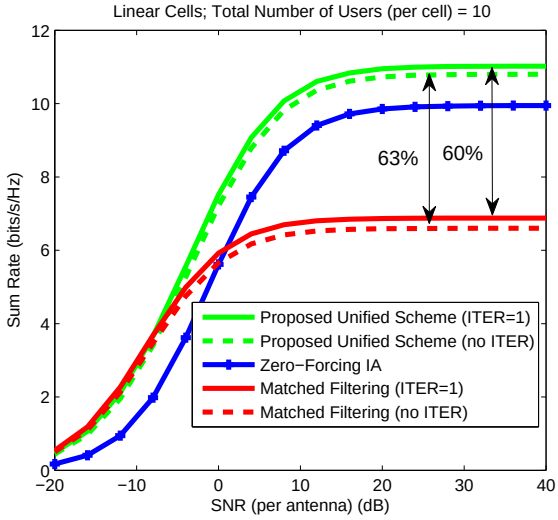


(b)

Figure 3.5: Sum-rate performance for a 19 hexagonal cell layout where $M = 4$, the number K of users per cell is 3 and the number S of streams is 3: (a) as a function of SNR (no iteration); (b) as a function of the number of iteration.



(a)



(b)

Figure 3.6: Sum-rate performance of the schemes integrated with an opportunistic scheduler when the number K of users per cell is 10 and the number S of streams is 3: (a) 19 hexagonal cell layout; (b) linear cell layout. The opportunistic scheduler chooses a set of 3 users out of 10 such that the sum-rate is maximized.

interest. Notice that a crossover point between the zero-forcing IA and the matched filtering occurs at around $\mathsf{SNR} = 0$ dB. The benefit of the zero-forcing IA is substantial.

Remark 14 (**Comparison to Other Techniques**). *In addition to the matched-filtering scheme, as other baselines, one may consider resource partitioning and cooperative scheduling [29]. However, these techniques are not fair enough to be compared to our IA scheme, since these incur signalling overhead while our scheme does not. Resource partitioning requires explicit coordination of frequency resources for many neighboring cells, thus incurring signalling overhead. Cooperative scheduling [29] requires additional communication between different BSs to deliver user scheduling information across cells. On the contrary, our IA scheme does not require explicit coordination, as it adapts only the number of streams under frequency reuse of 1. While in this work, detailed comparisons are not provided, doing comparative study needs to be done as future work especially for designing practical cellular systems [1, 2], where many system factors should be simultaneously taken into consideration with different weights of importance. In fact, this comparative study might give some insights into developing another scheme which combines the IA scheme and cooperative scheduling to provide the further performance gain.*

3.4.3 Application to Macro-pico Cellular Networks

We have observed that our scheme shows promise especially when dominant interference is much stronger than the remaining interference, i.e., $\gamma \ll 1$. Such scenario occurs often in heterogeneous networks [4] which use a mix of macro, pico, femto, and relay BSs to enable flexible and low-cost deployment. In this section, we focus on a scenario of the macro-pico cell deployment, illustrated in Fig. 3.7.

As shown in the figure, suppose that pico-BS is deployed at a distance d from the nearby macro-BS and a user is connected to the pico-BS. The pico-user can then see significant interference from the nearby macro-BS, and this interference can be much stronger than the aggregated interference from the remaining macro-BSs, especially when d is small. The interference problem can be further aggravated due to range extension techniques[1] [4] and the disparity between the transmit power levels of the macro-BS and the pico-BS. This motivates the need for intelligent interference management techniques. We show that our IA scheme can resolve this problem to provide substantial gain.

To show this, we evaluate the sum-rate performance of pico-users in the simple scenario shown in Fig. 3.7. We assume the 19 hexagonal wrap-around cellular layout, and on top of it we deploy one pico-BS, apart from the nearby macro-BS by a distance d. Based on [4], we consider the power levels of 46 dBm and 30 dBm for the macro-BS and the pico-BS, respectively, so the difference is 16 dB. This scenario reflects the case where the pico-cell,

[1]Range extension extends the footprint of pico-cells by allowing more users to connect even if users do not see the pico-BS as the strongest downlink received power. The purpose for this is to better utilize cell-splitting and maximize cell offloading gain.

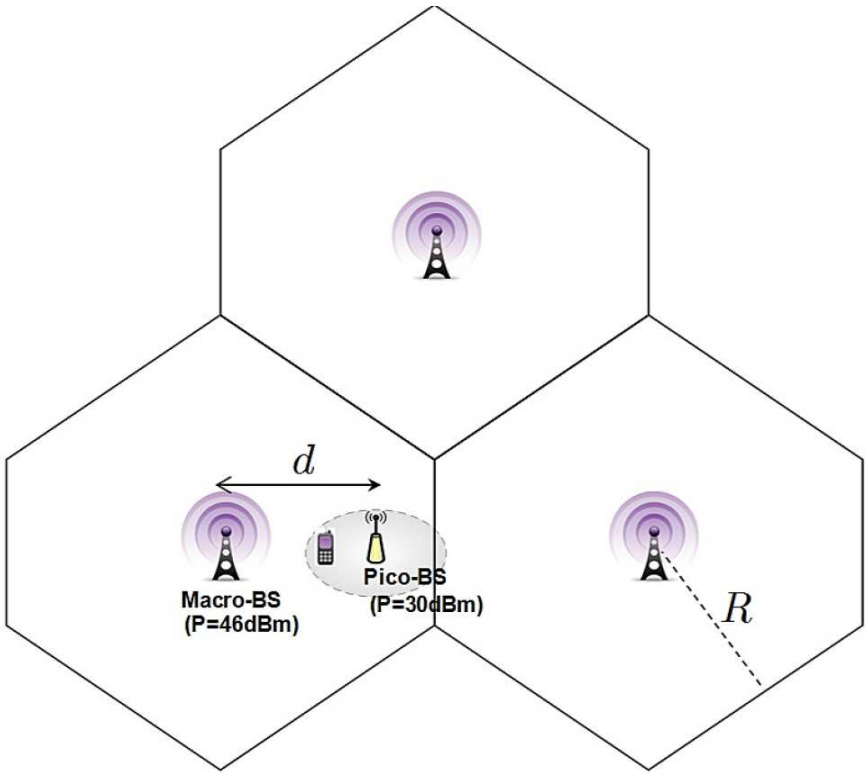
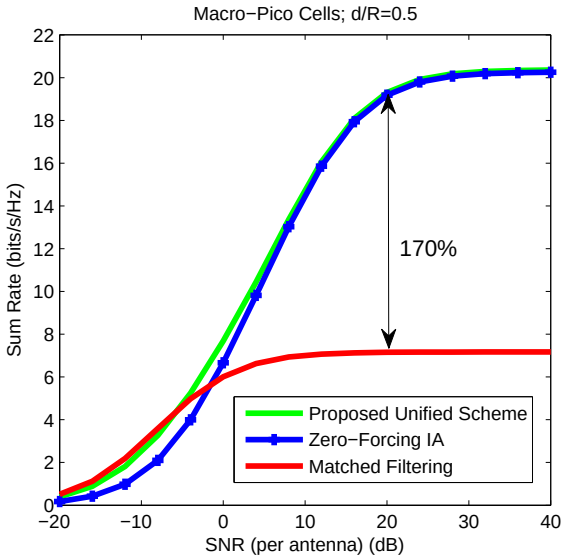
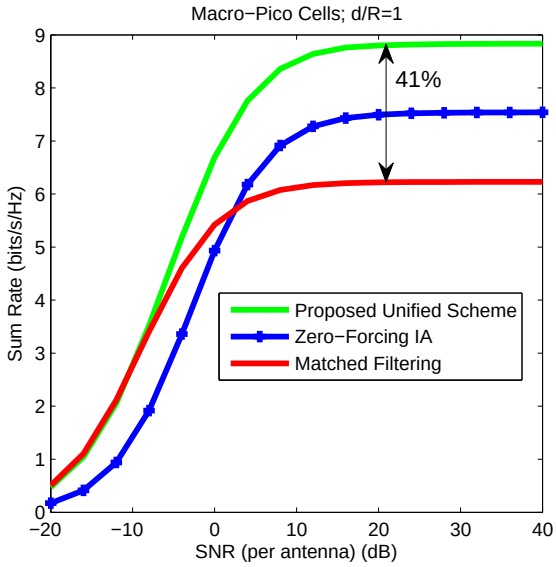


Figure 3.7: Macro-pico cellular networks. The pico-user can see significant interference from the nearby macro-BS. The interference problem can be further aggravated when the pico-BS is close to the nearby macro-BS (small d) and the power levels of the two BSs are quite different.



(a)



(b)

Figure 3.8: Sum-rate performance of pico-users for a macro-pico cell layout where a single pico-cell is deployed on top of 19 wrap-around macro cells (cell radius R) and the pico-BS is separated from the nearby macro-BS by a distance d: (a) $\frac{d}{R} = 0.5$; (b) $\frac{d}{R} = 1$. The number K of users per pico-cell (or macro-cell) is 10; the number S of streams is 3; and no iteration is performed. The sum-rate reflects the 3 pico-users chosen out of 10 via an opportunistic scheduler.

once chosen, is fixed once and for all. Consistent with previous simulation setups, we consider a specific mobile location where the downlink received power from the pico-BS is the same as that from the nearby macro-BS.Due to the disparity of the power levels, the pico-users are closer to the pico-BS.[2] We assume a 4-by-4 antenna configuration, i.e., $M = 4$. We assume that each of the pico cell and macro-cells has $K = 10$ users placed at the specific location, and 3 users are chosen at a time out of 10 via the opportunistic scheduler. We assume an interference model where the precoder of the nearby macro-BS is actually computed and this interferes with the users of interest, while the aggregated interference of the remaining macro-BSs is white Gaussian.

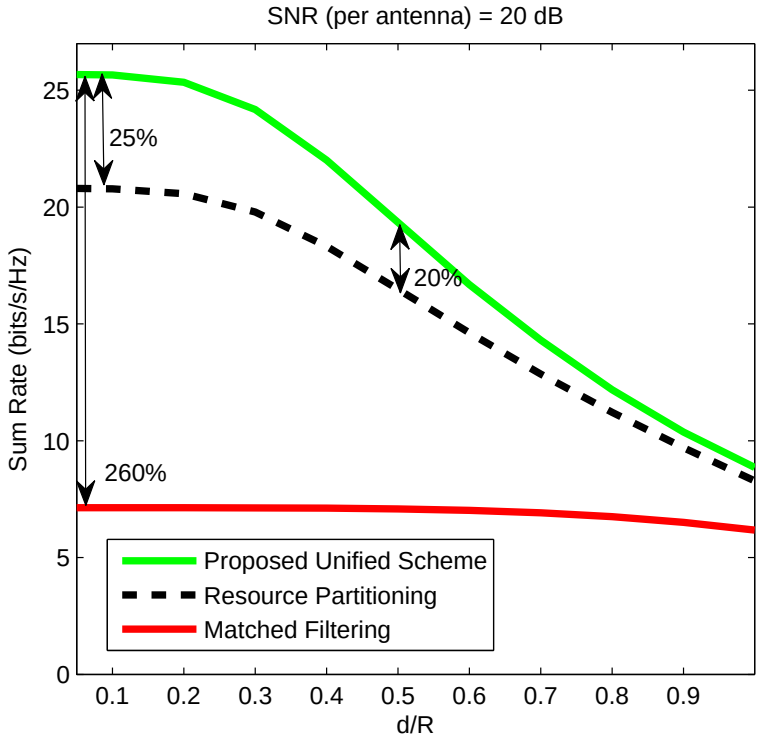


Figure 3.9: Comparison to resource partitioning. The sum-rate performance as a function of $\frac{d}{R}$ for $\mathsf{SNR} = 20$ dB.

Fig. 3.8 shows the sum-rate performance of the pico-users as a function of SNR. We assume that $S = 3$ and no iteration. Fig. 3.8 (a) considers the case of $\frac{d}{R} = 0.5$ where pico-users are interfered with by the nearby macro-BS. In this case, our IA scheme provides 170%

[2]In fact, this specific mobile location - where the downlink received power from the two BSs are the same - is a conservative setting. When employing the range extension technique that expands the footprint of pico-cells, one can expect a larger gain of our IA scheme, as the dominant interference power is stronger.

gain over matched filtering. Fig. 3.8 (b) considers the case of $\frac{d}{R} = 1$ where the minimum gain of our scheme is expected. Even in this case, our proposed scheme gives approximately 41% gain over the matched filtering.

Remark 15 (**Comparison to Resource Partitioning**). *In the macro-pico network scenario, as an alternative to our IA scheme, one may consider resource partitioning to resolve the interference problem. This is because unlike the conventional macro cellular networks containing many neighboring cells, this macro-pico network scenario has a fewer number of dominant interferers, thus making resource coordination simpler [3]. For example, we can use a frequency reuse of $\frac{1}{2}$ for the scenario in Fig. 3.7. So we provide simulation results and find that even in this case, our scheme shows respectable gain over resource partitioning. Fig. 3.9 shows the sum-rate performance of pico-users as a function of $\frac{d}{R}$ when $\mathsf{SNR} = 20$ dB and $K = 10$. We use $S = 3$ for the IA schemes and the matched filtering, while for resource partitioning we optimize the number of streams to plot the best performance curve. In the resource partitioning, we use frequency reuse $\frac{1}{2}$ only between the the nearby macro-cell and the pico-cell, while using frequency reuse 1 for the other macro-cells. Notice that our scheme gives approximately 20% gain for $\frac{d}{R} = 0.5$. The smaller the ratio of $\frac{d}{R}$, the larger the gain, while for large $\frac{d}{R}$, the gain becomes marginal.* ■

3.4.4 Extension

Asymmetric Antenna Configuration: We discuss the asymmetric antenna configuration where the BSs are equipped with more antennas, i.e., $M > N$. The extension to this asymmetric case is not straightforward, since more transmit antennas at BSs provide the possibility to null out interference at mobiles in other cells, thus requiring a sophisticated technique which well combines interference nulling with interference alignment.

Here we instead provide a simple and natural, but possibly suboptimal, variant of the proposed scheme. The scheme is to limit the number of streams with the minimum of M and N, i.e., $S \leq \min(M, N) = N$. Specifically, each BS sets the precoder $\bar{\mathbf{P}}$ as:

$$\bar{\mathbf{P}} = [\mathbf{f}_1, \cdots, \mathbf{f}_S, \kappa\mathbf{f}_{S+1}, \cdots, \kappa\mathbf{f}_M] \in \mathbb{C}^{M \times M}, \tag{3.12}$$

and sets the range of S as $S \leq N$. Other operations remain the same. Each user computes the expected covariance matrix by averaging over the transmitted signals from the other cell and then applies the standard MMSE formula for a receive vector. The BS then computes the zero-forcing transmit vectors with the feedback information. These steps can then be iterated.

Notice that in this scheme, interference alignment interpretation needs to be carefully made. For example, consider 4-by-2 antenna configuration in a two-cell layout. Our scheme allows each BS to send one stream out of two and therefore each user sees only one interference vector from the other cell. There is no aligned interference. Even in this configuration,

however, interference alignment can be achieved if multiple subcarriers are incorporated, as will be discussed in the following section.

Using Subcarriers: Recall in our simulations that only antennas are employed for multiple dimensions. However, we can easily increase M by using multiple subcarriers. With this increase of M, we can make two interesting observations. The first observation is that the performance improves with an increase of M, since the dimension reserved for interference signals becomes negligible as M gets larger. Secondly, increasing M, we can make a chance to achieve interference alignment. To see this, consider 8-by-4 configuration incorporating two subcarriers with a 4-by-2 antenna configuration. We will show that unlike the 4-by-2 configuration, this 8-by-4 configuration enables interference alignment. Suppose there are two cells and each cell has three users. Our scheme allows each BS to transmit three streams out of four and thus each user sees five interfering vectors in total: three *out-of-cell* and two *intra-cell* interfering vectors. Notice the five interfering vectors are aligned onto a three dimensional linear subspace. This implies interference alignment.

Multiple Interferers: Our IA technique removes the interference from a single dominant interferer. However, a slight modification can deal with the case of multiple dominant interferers, to some extent. For example, consider a 19 hexagonal cell layout in Fig. 3.4 and suppose that mobiles are located at the middle point of three neighboring BSs. In this case, mobiles see the two dominant interferers. One simple way is to take multiple dominant interferers into account in the process of computing the expected covariance matrix. Specifically, we use:

$$\bar{\Phi}_k := \mathbb{E}\left[(1+\mathsf{INR}_{\mathsf{rem}})\mathbf{I} + \frac{\mathsf{SNR}}{S}\mathbf{G}_{\beta k}\bar{\mathbf{P}}\mathbf{B}_\beta\mathbf{B}_\beta^*\bar{\mathbf{P}}^*\mathbf{G}_{\beta k}^* + \frac{\mathsf{SNR}}{S}\mathbf{G}_{\gamma k}\bar{\mathbf{P}}\mathbf{B}_\gamma\mathbf{B}_\gamma^*\bar{\mathbf{P}}^*\mathbf{G}_{\gamma k}^*\right], \qquad (3.13)$$

where $\mathbf{G}_{\beta k}$ denotes cross-channel from BS β and $\mathbf{B}_\beta$ indicates the zero-forcing precoder of BS β. Similarly, we denote $(\mathbf{G}_{\gamma k}, \mathbf{B}_\gamma)$ for cell γ. For $\mathbf{B}_\beta$ and $\mathbf{B}_\gamma$, we assume that each entry of the first S rows is i.i.d. $\mathcal{CN}\left(0, \frac{1}{S+(M-S)\kappa^2}\right)$ and each entry of the last $(M-S)$ rows is i.i.d. $\mathcal{CN}\left(0, \frac{\kappa^2}{S+(M-S)\kappa^2}\right)$.

3.5 Subspace Interference Alignment

Our IA technique in Section 3.4 focuses on the removal of the dominant interference while treating the residual interference as noise. This comes with limitations in applying this scheme to many of the realistic scenarios. It can be useful only when the dominant interference is much stronger than the residual interference, although the scenario can often occur in macro-pico cellular networks as shown in Section 3.4.3. While a slightly modified scheme in Section 3.4.4 has been developed in order to mitigate the interference from multiple dominant interferers, it does not ensure the complete removal of the interference even in the high SNR regime.

In this section, we propose a novel IA scheme, which we call *subspace interference alignment*, to address this challenge with respect to multiple dominant interferers. This scheme aligns the interference of multiple interferers onto a restricted subspace *simultaneously* at multiple non-intended receivers, whose dimension is negligible as compared to that of the subspace spanned by the desired signals, thus achieving almost interference-free dof even in the multiple (more than 2) cellular networks. A key property of this scheme is that the simultaneous interference alignment is achieved using only a *finite* number M of dimensions. This is in stark contrast to Cadambe-Jafar's IA scheme which employs an *infinite* number of dimensions to achieve the simultaneous interference alignment. On the flip side, however, this scheme comes with limitations on the wireless channel structure that the technique relies on. It requires a decomposition property of wireless channels that will be described in Section 3.5.1.

For illustrative purpose, we focus on the uplink scenario, while it can be easily adapted to the downlink scenario as we did for the two-cell case in Section 3.3. For simplicity, we do not consider the integration with schedulers that can be easily done as before. We start with the simplest non-trivial case (3-cell scenario) and then generalize it to the G number of cells. In Section 3.5.3, we remark that our subspace IA scheme can be exploited to address one of the significant problems in storage networks, called the storage-node repair problem.

3.5.1 3-cell Scenario

Scheme Description: As shown in Fig. 3.1, achieving interference alignment in the two-cell case is straightforward, since there is a single non-intended BS. We can easily design the transmitted vectors of users in one cell so that those span only one dimensional subspace at the other BS. However, it is not straightforward from the 3-cell case. Unlike the two-cell case, there are multiple non-intended BSs and this requires simultaneous interference alignment. We address this challenge by relaxing the interference alignment constraint: alignment of the interference space into one dimensional subspace. The idea is to align the interference into *multi-dimensional subspace* instead of one dimension.

Fig. 3 illustrates the idea. Here we use the number $M = (\sqrt{K}+1)^2$ of dimension and assume that channel matrix from each user to BS can be decomposed into two sub channel matrices with Kronecker product:

$$\mathbf{H}_{\alpha k}^{\beta} = \mathbf{H}_{\alpha k}^{\beta,(1)} \otimes \mathbf{H}_{\alpha k}^{\beta,(2)}, \tag{3.14}$$

where $\mathbf{H}_{\alpha k}^{\beta} \in \mathbb{C}^M$ denotes channel matrix from user k of cell α to BS β. Note that the dimension of sub channel matrix $\mathbf{H}_{\alpha k}^{\beta,(j)}$ is $\sqrt{M}$-by-$\sqrt{M}$. Later we will show the case where this channel assumption holds. In an attempt to visualize subspace, we abstract the subspace with grids. For instance, one grid represents one dimensional subspace, while two grids represent the two-dimensional one. Each user in a cell transmits one symbol $x_{\alpha k}$ along with its transmitted vector designed using Kronecker product of two sub-vectors, e.g., $\mathbf{v}_{\alpha k} = \mathbf{v}_{\alpha k}^{(1)} \otimes \mathbf{v}_{\alpha k}^{(2)}$.

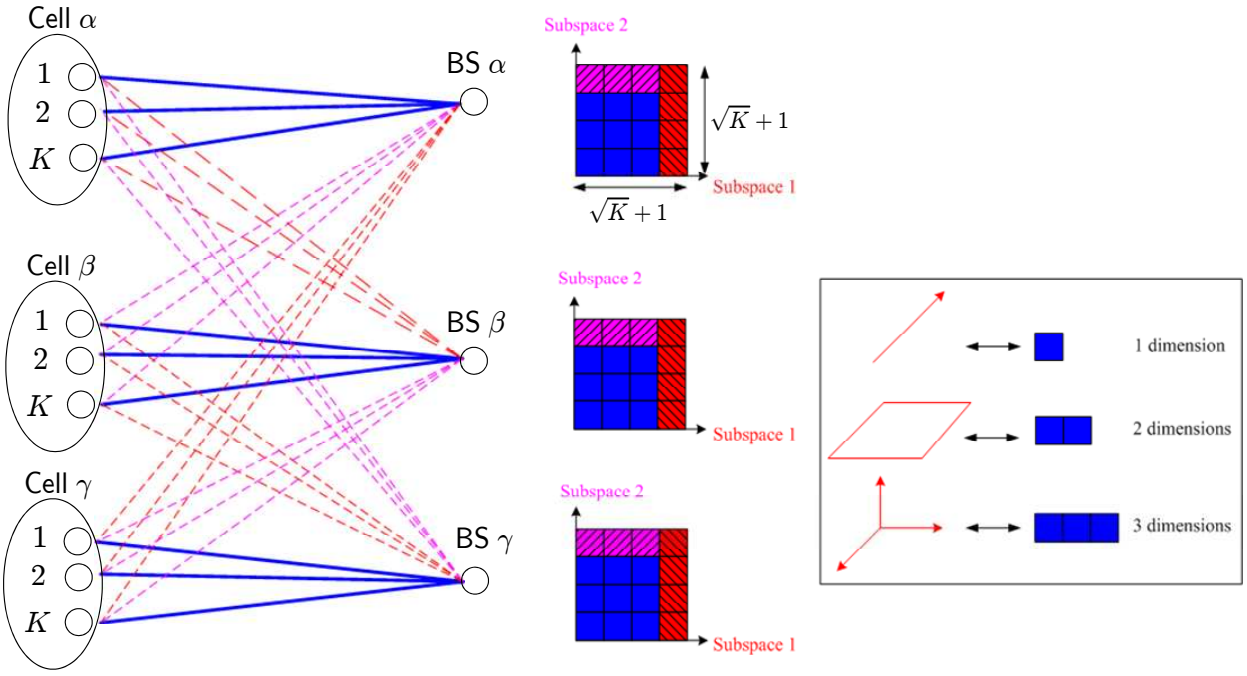


Figure 3.10: Subspace Interference Alignment: aligning interferences into multi-dimensional subspace (instead of one dimension)

Now the idea is to design each sub-vector so that it ensures to achieve IA partially at each of non-intended BSs. For example, user k in cell α designs sub-vectors 1 and 2 so that these are aligned with a fixed reference vector respectively at BS β and γ:

$$\begin{aligned} \mathbf{v}_{\alpha k}^{(1)} &= (\mathbf{H}_{\alpha k}^{\beta,(1)})^{-1}\mathbf{v}_{\mathsf{ref}}, \\ \mathbf{v}_{\alpha k}^{(2)} &= (\mathbf{H}_{\alpha k}^{\gamma,(2)})^{-1}\mathbf{v}_{\mathsf{ref}}, \end{aligned} \tag{3.15}$$

where $\mathbf{v}_{\mathsf{ref}} \in \mathbb{C}^{\sqrt{M}}$ is an arbitrary vector, independent of channels. Similarly we design the transmitted sub-vectors for users in cells β and γ:

$$\begin{aligned} \mathbf{v}_{\beta k}^{(1)} &= (\mathbf{H}_{\beta k}^{\gamma,(1)})^{-1}\mathbf{v}_{\mathsf{ref}}, \quad \mathbf{v}_{\beta k}^{(2)} = (\mathbf{H}_{\beta k}^{\alpha,(2)})^{-1}\mathbf{v}_{\mathsf{ref}}; \\ \mathbf{v}_{\gamma k}^{(1)} &= (\mathbf{H}_{\gamma k}^{\alpha,(1)})^{-1}\mathbf{v}_{\mathsf{ref}}, \quad \mathbf{v}_{\gamma k}^{(2)} = (\mathbf{H}_{\gamma k}^{\beta,(2)})^{-1}\mathbf{v}_{\mathsf{ref}}. \end{aligned} \tag{3.16}$$

Then the received signal $\mathbf{y}_\alpha$ of BS α is given by

$$\begin{aligned}
\mathbf{y}_\alpha &= \sum_{k=1}^{K}(\mathbf{H}_{\alpha k}\mathbf{v}_{\alpha k})x_{\alpha k} + \sum_{k=1}^{K}(\mathbf{H}^{\alpha}_{\beta k}\mathbf{v}_{\beta k})x_{\beta k} + \sum_{k=1}^{K}(\mathbf{H}^{\alpha}_{\gamma k}\mathbf{v}_{\gamma k})x_{\gamma k} + \mathbf{z}_\alpha \\
&\overset{(a)}{=} \sum_{k=1}^{K}\left[(\mathbf{H}^{(1)}_{\alpha k}\mathbf{v}^{(1)}_{\alpha k}) \otimes (\mathbf{H}^{(2)}_{\alpha k}\mathbf{v}^{(2)}_{\alpha k})\right] x_{\alpha k} \\
&\quad + \sum_{k=1}^{K}\left[(\mathbf{H}^{\alpha,(1)}_{\beta k}\mathbf{v}^{(1)}_{\beta k}) \otimes (\mathbf{H}^{\alpha,(2)}_{\beta k}\mathbf{v}^{(2)}_{\beta k})\right] x_{\beta k} + \sum_{k=1}^{K}\left[(\mathbf{H}^{\alpha,(1)}_{\gamma k}\mathbf{v}^{(1)}_{\gamma k}) \otimes (\mathbf{H}^{\alpha,(2)}_{\gamma k}\mathbf{v}^{(2)}_{\gamma k})\right] x_{\gamma k} + \mathbf{z}_\alpha \\
&\overset{(b)}{=} \sum_{k=1}^{K}\left[\left\{\mathbf{H}^{(1)}_{\alpha k}(\mathbf{H}^{\beta,(1)}_{\alpha k})^{-1}\mathbf{v}_{\mathsf{ref}}\right\} \otimes \left\{\mathbf{H}^{(2)}_{\alpha k}(\mathbf{H}^{\gamma,(2)}_{\alpha k})^{-1}\mathbf{v}_{\mathsf{ref}}\right\}\right] x_{\alpha k} \\
&\quad + \sum_{k=1}^{K}\left[\left\{\mathbf{H}^{\alpha,(1)}_{\beta k}(\mathbf{H}^{\gamma,(1)}_{\beta k})^{-1}\mathbf{v}_{\mathsf{ref}})\right\} \otimes \mathbf{v}_{\mathsf{ref}}\right] x_{\beta k} + \sum_{k=1}^{K}\left[\mathbf{v}_{\mathsf{ref}} \otimes \left\{\mathbf{H}^{\alpha,(2)}_{\gamma k}(\mathbf{H}^{\beta,(2)}_{\gamma k})^{-1}\mathbf{v}_{\mathsf{ref}}\right\}\right] x_{\gamma k} + \mathbf{z}_\alpha,
\end{aligned}$$

where (a) follows from a mixed product property as below and (b) follows from (3.15) and (3.16). Note the mixed product property:

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD}),$$

where $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ are matrices with appropriate dimensions.

Now let us consider the interference from cell β. It contains K interfering vectors with dimension of $M = (\sqrt{K}+1)^2 > K$. So the dimension of the interference space can be up to K. However, the dimension is limited by $\sqrt{K}+1$ due to its degenerated structure. Note that the second sub-vector is fixed as $\mathbf{v}_{\mathsf{ref}}$ for all of the users in cell β. The randomness comes only from the first sub-vector $\left\{\mathbf{H}^{\alpha,(1)}_{\beta k}(\mathbf{H}^{\gamma,(1)}_{\beta k})^{-1}\mathbf{v}_{\mathsf{ref}})\right\}$ with dimension of $\sqrt{K}+1$. Therefore, the dimension is limited by $\sqrt{K}+1$. Similarly, the dimension of the interference space with respect to cell γ users is at most $\sqrt{K}+1$.

On the other hand, no alignment is achieved with respect to the desired signals $x_{\alpha k}$'s. Hence, with sufficient randomness of the channels, the dimension of the signal subspace spanned by the desired signals is K. Also with high probability, the subspace is disjoint with the interference subspace with respect to β and γ cell users. So we can decode K symbols using $M = (\sqrt{K}+1)^2$. Similarly, we do the same procedure for the received signals of BS β and γ, thus obtaining $3K$ symbols in total. Therefore, we achieve the following dof:

$$\mathsf{dof}(\mathsf{per\ cell}) = \frac{K}{(\sqrt{K}+1)^2} \longrightarrow 1. \tag{3.17}$$

Notice that our subspace IA scheme aligns K interfering vectors into $\sqrt{K}+1$ dimensional subspace, thus achieving simultaneous IA at multiple non-intended BSs. Since $\sqrt{K}$ is negligible as compared to K for a large value of K, we can approach the interference-free dof with an increase in K.

Wireless Channel Structure: As mentioned earlier, our proposed scheme relies on the decomposition property of wireless channels described in (3.14). We will show in the sequel that this property holds for signle antenna single-path random delay channels.

Let $H[f] = h_\ell W_{N_T}^{\ell f}$ be the frequency response of single-path wireless channels where f denotes subcarrier index and ℓ denotes discrete-time tap delay normalized to symbol rate $\frac{1}{W}$ where W indicates bandwidth. Here h_ℓ denotes the baseband discrete-time channel coefficient with respect to the tap delay ℓ and $W_{N_T} = \exp\left(-j\frac{2\pi}{N_T}\right)$ and N_T is an IDFT/DFT size. We now write f in terms of f_1 and f_2:

$$f = (\sqrt{K}+1)f_1 + f_2,\ \forall\ f_1, f_2 \in \left\{0, 1, \cdots, \sqrt{K}\right\},$$

We can then decompose the channel $H[f]$ as follows:

$$H[f] = \left(h_\ell W_N^{\ell(\sqrt{K}+1)f_1}\right) \cdot \left(W_{N_T}^{\ell f_2}\right). \tag{3.18}$$

Let $H^{(1)}[f_1] = h_\ell W_{N_T}^{\ell(\sqrt{K}+1)f_1}$ and $H^{(2)}[f_2] = W_{N_T}^{\ell f_2}$. We then get:

$$\begin{aligned}\mathbf{H} &= \mathsf{diag}\left\{H[f]\right\}_f = \mathsf{diag}\left\{H^{(1)}[f_1]H^{(2)}[f_2]\right\}_{f_1,f_2} \\ &= \mathsf{diag}\left\{H^{(1)}[f_1]\right\}_{f_1} \otimes \mathsf{diag}\left\{H^{(2)}[f_2]\right\}_{f_2} \\ &= \mathbf{H}^{(1)} \otimes \mathbf{H}^{(2)}.\end{aligned} \tag{3.19}$$

A natural question that arises is to ask whether multi-path frequency-selective channels are decomposable. Unfortunately, the answer is no in general. However, we can apply the subspace IA scheme in an indirect manner. The idea is to chop up the whole band into sub-bands within coherence bandwidth. We now show that the channel is decomposable within the sub-band. To see this, suppose that the channel has two non-zero taps at ℓ_1 and ℓ_2.

$$H[f] = h_{\ell_1} W_{N_T}^{\ell_1 f} + h_{\ell_2} W_{N_T}^{\ell_2 f} = W_{N_T}^{\ell_1 f}\left(h_{\ell_1} + h_{\ell_2} W_{N_T}^{(\ell_2-\ell_1)f}\right).$$

Since the coherence bandwidth is $W_c \triangleq \frac{W}{2(\ell_2-\ell_1)}$, the term $(h_{\ell_1} + h_{\ell_2} W_{N_T}^{(\ell_2-\ell_1)f})$ is almost constant within W_c. This implies that the channel virtually has a single tap within sub-band. Hence, we can apply the subspace IA scheme for each sub-band.

However, highly frequency selective channels come with some challenge in practice. This is because the number of subcarriers within coherence bandwidth can be so small that we significantly lose the efficiency in achieving the dof. While very fine subcarrier spacing might resolve this challenge, it comes with significant inter-carrier interference due to Doppler effect and therefore increase hardware complexity.

3.5.2 Generalization

In this section, we extend the subspace IA scheme to an arbitrary number G of cells. Similar to the 3-cell scenario, we assume that channels have the decomposition property: $\forall i, j \in \{0, 1, \cdots, G-1\}$,

$$\mathbf{H}_{ik}^{j} = \bigotimes_{g=0}^{G-2} \mathbf{H}_{ik}^{j,g}, \tag{3.20}$$

where $\mathbf{H}_{ik}^{j,g} \in \mathbb{C}^{\sqrt[G-1]{M}}$ denotes the gth sub channel matrix from user k in cell i to BS j. Here we use the number $M = (\sqrt[G-1]{K} + 1)^{G-1}$ of dimensions. To consider the general case, we employ heavy notations. The idea is the same as before. So we design the transmitted sub-vectors as: $\forall i, g \in \{0, 1, \cdots, G-1\}$,

$$\mathbf{v}_{ik}^{(g)} = (\mathbf{H}_{ik}^{i+g+1,(g)})^{-1} \mathbf{v}_{\mathsf{ref}}. \tag{3.21}$$

Then, one can easily verify that this aligns the interference space into $\sqrt[G-1]{M}$ dimensional subspace at all of the non-intended BSs while ensuring the full rank condition on the desired signals. So we can achieve the following:

$$\mathsf{dof}(\mathsf{per\ cell}) = \frac{K}{(\sqrt[G-1]{K} + 1)^{G-1}} \longrightarrow 1. \tag{3.22}$$

3.5.3 Application to Storage Networks

Recently, the authors in [13] found that our subspace IA scheme can be exploited to develop a practical repair strategy for failed storage nodes in the context of distributed storage networks. A very promising fact is that while in the cellular network context, this subspace IA scheme requires a special structure on wireless channels (determined by nature), this practical challenge disappears in the distributed storage networks. This is because wireless channels turn out to correspond to storage-code coefficients in the storage networks that are man-made design choices. This will be discussed in more details in the next chapter.

3.6 Summary

We have observed that the zero-forcing IA scheme is analogous to the zero-forcing receiver, and the iterative matched-filtering technique corresponds to the conventional matched-filter receiver. Based on this observation, we proposed a unified IA technique similar to an MMSE receiver that outperforms both techniques for all values of γ, where the power of the dominant interferer may be much greater or smaller than the power of the remaining aggregate interference.

Of practical importance is the fact that our proposed scheme can be implemented with small changes to an existing cellular system supporting multi-user MIMO, as it requires only a localized *within-a-cell* feedback mechanism. This technique can be extended to asymmetric antenna configurations and scenarios with more than one dominant interferer. Our technique also shows even greater performance gains for macro-pico cellular networks where the dominant interference is much stronger than the remaining interference.

We also propose another IA scheme, called subspace IA, in an attempt to mitigate the interference from multiple dominant interferers. We have shown that under some channel conditions, our subspace IA scheme can asymptotically achieve the interference-free dof with an increase in the number K of users in each cell. Unlike Cadambe-Jafar's IA scheme, it uses a finite number of dimensions to achieve simultaneous IA. We also remark that our subspace IA scheme can be well exploited to address one of the significant problems in distributed storage networks: the failed storage-node repair problem.

Chapter 4

Interference Alignment for Storage Networks

4.1 Introduction

In distributed storage systems, maximum distance separable (MDS) erasure codes are well-known coding schemes that can offer maximum reliability for a given storage overhead. For an (n, k) MDS code for storage, a source file of size $\mathcal{M}$ bits is divided equally into k units (of size $\frac{\mathcal{M}}{k}$ bits each), and these k data units are expanded into n encoded units, and stored at n nodes. The code guarantees that a user or Data Collector (DC) can reconstruct the source file by connecting to any arbitrary k nodes. In other words, any $(n-k)$ node failures can be tolerated with a minimum storage cost of $\frac{\mathcal{M}}{k}$ at each of n nodes. While MDS codes are optimal in terms of reliability versus storage overhead, they come with a significant maintenance overhead when it comes to repairing failed encoded nodes to restore the MDS system-wide property. Specifically, consider failure of a single encoded node and the cost needed to restore this node. It can be shown that this repair incurs an aggregate cost of $\mathcal{M}$ bits of information from k nodes. Since each encoded unit contains only $\frac{\mathcal{M}}{k}$ bits of information, this represents a k-fold inefficiency with respect to the repair bandwidth.

This challenge has motivated a new class of coding schemes, called Regenerating Codes [21, 73], which target the information-theoretic optimal tradeoff between storage cost and repair bandwidth. Dimakis-Godfrey-Wu-Wainwright-Ramchandran [21, 73] have translated the regenerating-codes problem into a multicast network problem. Employing the network code results in [5, 40, 34] that well address the multicast network, they have shown that random network coding schemes achieve the optimal repair bandwidth for a given storage cost. On one end of this spectrum of Regenerating Codes are Minimum Storage Regenerating (MSR) codes that can match the minimum storage cost of MDS codes while also significantly reducing repair bandwidth. As shown in [21, 73], the fundamental tradeoff between bandwidth and storage depends on the number of nodes that are connected to repair a failed

node, simply called the degee d where $k \leq d \leq n-1$. The optimal tradeoff is characterized by

$$(\alpha, \gamma) = \left(\frac{\mathcal{M}}{k}, \frac{\mathcal{M}}{k} \cdot \frac{d}{d-k+1} \right), \tag{4.1}$$

where α and γ denote the optimal storage cost and repair bandwidth, respectively for repairing a single failed node, while retaining the MDS-code property for the user. Note that this code requires the same minimal storage cost (of size $\frac{\mathcal{M}}{k}$) as that of conventional MDS codes, while substantially reducing repair bandwidth by a factor of $\frac{k(d-k+1)}{d}$ (e.g., for $(n,k,d) = (31,6,30)$, there is a 5x bandwidth reduction). This (n,k,d) MSR code can be considered as a Repair MDS code (to be specifically defined in Section 4.2.1) that (a) have an (n,k) MDS-code property; and (b) can repair single-node failures with minimum repair bandwidth given a repair-degree of d. In this work, we assume that each repair link has the equal bandwidth and its bandwidth $(\frac{\gamma}{d})$ is normalized to 1, making $\mathcal{M} = k(d-k+1)$. One can partition a whole file into smaller chunks so that each has a size of $k(d-k+1)$.

While Repair MDS codes enjoy substantial benefits over conventional MDS codes, they come with some limitations in construction. Specifically, the achievable schemes in [21, 73] that meet the optimal tradeoff bound of (4.1) restore failed nodes in a *functional* manner only, using a random-network-coding based framework. This means that the replacement nodes maintain the MDS-code property (that any k out of n nodes can allow for the data to be reconstructed) but do not *exactly* replicate the information content of the failed nodes.

Mere functional repair can be limiting. First, in many applications of interest, there is a need to maintain the code in systematic form, i.e., where the user data in the form of k information units are exactly stored at k nodes and parity information (mixtures of k information units) are stored at the remaining $(n-k)$ nodes. Secondly, under functional repair, additional overhead information needs to be exchanged for continually updating repairing-and-decoding rules whenever a failure occurs. This can significantly increase system overhead. A third problem is that the random-network-coding based solution of [21] can require a huge finite-field size, which can significantly increase the computational complexity of encoding-and-decoding[1]. Lastly, functional repair is undesirable in storage security applications in the face of eavesdroppers. In this case, information leakage occurs continually due to the dynamics of repairing-and-decoding rules that can be potentially observed by eavesdroppers [49].

These drawbacks motivate the need for *exact* repair of failed nodes. This leads to the following question: is there a price for attaining the optimal tradeoff of (4.1) with the extra constraint of exact repair: i.e., is there an overhead cost in terms of rate needed? Unlike functional repair, this exact-repair problem can be translated into a *non-multicast* network

[1]Recall that the regenerating-codes problem can be translated into a multicast communication problem where random-network-coding-based schemes require a huge field size especially for large networks. In storage problems, the field size issue is further aggravated by the need to support a dynamically expanding network size due to the need for continual repair.

problem (to be specifically shown in Section 4.2.2) where the cutset bound might not be achievable [74] and linear network codes might not suffice [23]. Due to this nature, the problem has been open in general. The work in [54] sheds some light on this exact-repair problem: specifically, it was shown that under scalar linear codes[2], the optimal tradeoff cannot be achieved when $\frac{k}{n} > \frac{1}{2} + \frac{2}{n}$. For large n, this case boils down to $\frac{k}{n} > \frac{1}{2}$, i.e., redundancy less than two. Now what about for $\frac{k}{n} \leq \frac{1}{2}$?

The first contribution of this work is to resolve this open problem by showing that scalar-linear Exact-Repair MDS codes come with no extra cost over the optimal tradeoff of (4.1) for the case of $\frac{k}{n} \leq \frac{1}{2}$ and $d \geq 2k-1$[3]. Our codes are deterministic and require a field size of at most $2(n-k)$. Our result draws its inspiration from the work in [54], which guarantees exact repair of systematic node, while satisfying the MDS code property, but which does not provide exact repair of failed parity nodes. In providing a constructive solution for the exact repair of all nodes, we use geometric insights to propose a large family of repair codes. This both provides insights into the structure of codes for exact repair of all nodes, as well as opens up a rich design space for constructive solutions. This will be explained in Section 4.4.

The second contribution is to establish the following fact. Under *vector linear* codes which allow for the break-up of stored symbols into arbitrarily small subsymbols, we show the existence of Exact-Repair MDS codes that achieve the optimal tradoff of (4.1) for the entire admissible spectrum of (n,k,d), i.e., $k < n$ and $k \leq d \leq n-1$.[4] That is we show that there is no theoretical gap between exact repair and functional repair codes for the entire range of (n,k,d). This will be explained in Section 4.5.

Our results for both constructive scalar-linear codes and vector-linear codes build on the concept of *interference alignment*, which was introduced in the context of wireless communication networks [44, 14]. The idea of interference alignment is to align multiple interference signals in a signal subspace whose dimension is smaller than the number of interferers. Specifically, consider the following setup where a decoder has to decode one desired signal which is linearly interfered with by two separate undesired signals. How many linear equations (relating to the number of channel uses) does the decoder need to recover its desired input signal? As the aggregate signal dimension spanned by desired and undesired signals is at most three, the decoder can naively recover its signal of interest with access to three linearly independent equations in the three unknown signals. However, as the decoder is interested in only one of the three signals, it can decode its desired unknown signal even if it has access to only two equations, provided the two undesired signals are judiciously aligned in a 1-dimensional subspace. See [44, 14, 60] for details.

We will describe in the sequel how this concept relates intimately to our repair problem. At a high level, the connection comes from our repair problem involving recovery of a subset

[2]In scalar linear codes, symbols are not allowed to be split into arbitrarily small sub-symbols as with vector linear codes. This vector linear code is equivalent to having large block-lengths in the classical setting.

[3]Here we assume that all of the surviving systematic nodes participate in the repair

[4]Independently, Cadambe-Jafar-Maleki [15] have shown the existence of vector linear Exact-Repair MDS codes that attain the optimal tradeoff of (4.1) for (n,k,d) where $k < n$ and $d = n-1$.

(related to the subspace spanned by a failed node) of the overall aggregate signal space (related to the entire user data dimension). There are, however, significant differences some beneficial and some detrimental. On the positive side, while in the wireless problem, the equations are provided by nature (in the form of channel gain coefficients), in our repair problem, the coefficients of the equations are man-made choices, representing a part of the overall design space. On the flip side, however, the MDS requirement of our repair code and the multiple failure configurations that need to be simultaneously addressed with a single code design generate multiple interference alignment constraints that need to be simultaneously satisfied. This is particularly acute for a large value of k, as the number of possible failure configurations increases with n (which increases with k). Finally, another difference comes from the finite-field constraint of our repair problem.

4.2 Problem Statement

4.2.1 Definition of Repair MDS codes

While conventional MDS erasure codes are completely characterized by their encoding (generator) matrix, Repair MDS codes need more. They require not only the MDS property (as in the classical case), but have the additional repair constraints corresponding to all single-node failure patterns. This makes the code design problem considerably more challenging. We discuss this here by defining Repair MDS codes through their complete code-design space characterization. In the interests of keeping the notation simple without sacrificing the conceptual insights behind this characterization, we will consciously avoid the formalism associated with a general setting, and instead use illuminating examples to illustrate our results while reserving the detailed formal proofs to the appendices.

Consider a simple example of a systematic $(n, k, d) = (4, 2, 3)$ code in Fig. 4.1. Note that the degree d indicates the number of nodes that are connected to repair a failed node. We introduce matrix notation for illustrative purpose. This code has $k(= 2)$ information units. Let $\mathbf{a} = (a_1, \cdots, a_\alpha)^t$ and $\mathbf{b} = (b_1, \cdots, b_\alpha)^t$ be α-dimensional information-unit vectors, where α denotes storage cost and $(\cdot)^t$ indicates a transpose. Systematic node 1 and 2 store uncoded information in the form of row vectors, i.e., $\mathbf{a}^t$ and $\mathbf{b}^t$, respectively. Let $\mathbf{A}_i$ and $\mathbf{B}_i$ be α-by-α encoding submatrices (i.e., $[\mathbf{A}_i; \mathbf{B}_i]$ corresponds to generator submatrices) for parity node i $(i = 1, 2)$. For example, parity node 1 stores information in the form of $\mathbf{a}^t\mathbf{A}_1 + \mathbf{b}^t\mathbf{B}_1$. The encoding submatrices for systematic nodes are not explicitly defined, since those are trivially inferred.

A failed node is repaired through the specification of α-dimensional projection vectors associated with each survivor node that participates in the repair. As we assume a unit per-link repair-bandwidth cost ($\frac{\gamma}{d} = 1$), each survivor node projects its data into a scalar. In the example, $\mathbf{v}_{\alpha i}$ $(i = 1, 2, 3)$ are defined as the projection vectors needed for repair of systematic node 1. A Repair MDS code is thus defined as having two functional components

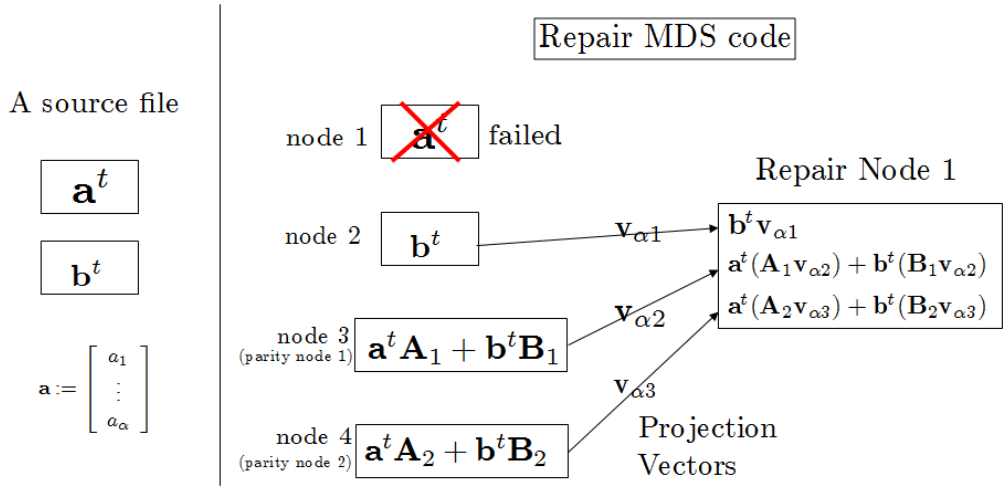


Figure 4.1: Definition of a Repair MDS code through the complete characterization of the code design space using the example of a systematic $(n, k, d) = (4, 2, 3)$ code. This is illustrated for the case when systematic node 1 fails, and a unit per-link repair-bandwidth cost is assumed. Let $\mathbf{a} = (a_1, \cdots, a_\alpha)^t$ and $\mathbf{b} = (b_1, \cdots, b_\alpha)^t$ be α-dimensional information-unit vectors, where α denotes the storage cost per node. Systematic node 1 and 2 store uncoded information in the form of row vectors, i.e., $\mathbf{a}^t$ and $\mathbf{b}^t$, respectively. Let $\mathbf{A}_i$ and $\mathbf{B}_i$ be α-by-α encoding submatrices (i.e., $[\mathbf{A}_i; \mathbf{B}_i]$ corresponds to generator submatrices) for parity node i $(i = 1, 2)$. A failed node is repaired through the specification of α-dimensional projection vectors associated with each surviving node that participates in the repair. In the example, $\mathbf{v}_{\alpha i}$ $(i = 1, 2, 3)$ are defined as the projection vectors needed for repair of systematic node 1. A Repair MDS code is thus defined as having two functional components that have to be designed *jointly*: (1) the encoding (generator) matrix associated with the storage nodes; and (2) the projection vectors needed for node repair. Note that in this example, the repair code involves 4 encoding submatrices and 12 projection vectors (3 projection vector for each of 4 possible failure configurations) that need to be designed jointly.

that have to be designed *jointly*:

1. the encoding (generator) matrix associated with the storage nodes;

2. the projection vectors needed for node repair.

Note that in this example, the repair code involves 4 α-by-α encoding submatrices and 12 projection vectors (3 projection vectors for each of 4 possible failure configurations) that need to be jointly designed.

We categorize the Repair MDS code depending on whether or not the failed nodes are exactly repaired. The code is called a *functional*-repair code if the repaired system maintains the MDS-code property (the repaired node can however be different from that of the failed node). The code is called an *exact*-repair code if the failed nodes are exactly repaired, thus restoring lost encoded fragments with their exact replicas. The code is called a *partial exact*-repair code if only the systematic nodes are repaired exactly, while parity nodes are repaired only functionally. Finally, the code is also called the MSR code that achieves the optimal tradeoff of (4.1).

The repair problem is to construct the repair code. For instance, the exact-repair problem is to jointly design (1) the encoding (generator) matrix and (2) the projection vectors such that the failed nodes are exactly repaired.

4.2.2 Translation into a Non-Multicast Network Problem

Unlike functional repair which is equivalent to a multicast network problem [21, 73], the exact-repair problem we study here is a more complicated non-multicast network problem which in general is an open problem in network coding today. It is known that in general non-multicast networks, the cutset bound might not be achievable [74] and linear codes might not suffice [23]. In this section, we explicitly show this translation to highlight the difficulty of our exact-repair problem. As we will show in the sequel, we show that exploiting the special structure of our non-multicast problem due to the exact repair constraints, we can solve the problem for admissible values of (n, k, d).

Fig. 4.2 shows the translation of the $(4, 2, 3)$ Exact-Repair MDS code into a non-multicast network where destination nodes have asymmetric traffic demands. A source has $k(= 2)$ information units $\mathbf{a}$ and $\mathbf{b}$, each having α symbols. We have $n(= 4)$ storage nodes. The two systematic nodes store $\mathbf{a}^t$ and $\mathbf{b}^t$, respectively, while the two parity nodes store mixtures of $\mathbf{a}$ and $\mathbf{b}$. Here we consider linear combination mixtures, although the mixtures can also be arbitrary non-linear functions of the information. We have 4 repair nodes. When node 1 fails, repair node 1 (denoted by R_1) needs to decode $\hat{\mathbf{a}}$ by connecting to $d(= 3)$ survivor nodes. Similarly we have the other three repair nodes. In addition to this, due to the MDS-code constraint, there are $\binom{n}{k} = \binom{4}{2} = 6$ destination nodes which need to decode all of the information units. Clearly the resulting network is a non-multicast network which contains two types of destination nodes: (1) 4 destination nodes want the individual

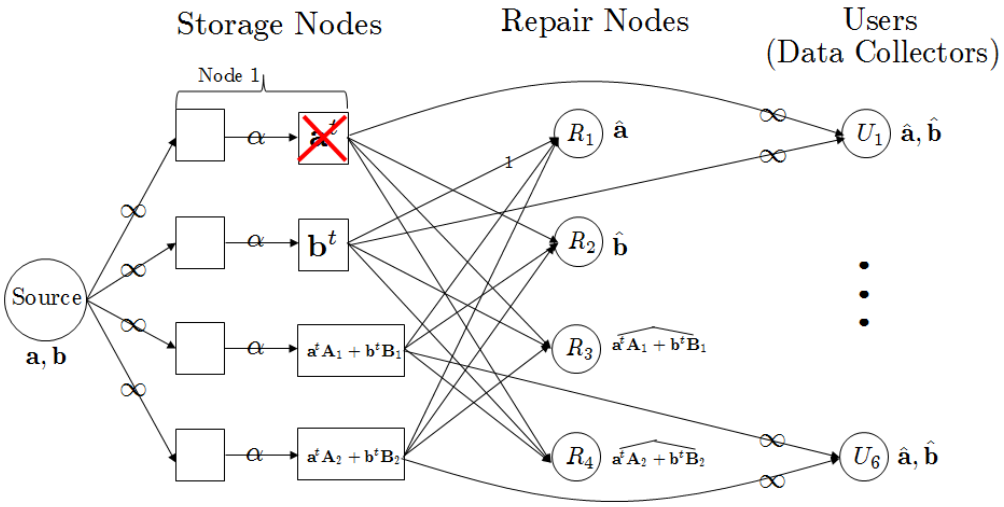


Figure 4.2: Translation of the $(4, 2, 3)$ Exact-Repair MDS code into a non-multicast network problem. A source has $k(= 2)$ information units $\mathbf{a}$ and $\mathbf{b}$, each having α symbols. We have $n(= 4)$ storage nodes. The two systematic nodes store $\mathbf{a}^t$ and $\mathbf{b}^t$, respectively, while the two parity nodes store mixtures of $\mathbf{a}$ and $\mathbf{b}$. When node 1 fails, repair node 1 (denoted by R_1) needs to decode $\hat{\mathbf{a}}$ by connecting to $d(= 3)$ survivor nodes. Similarly we have the other three repair nodes. In addition to this, due to the MDS-code constraint, there are $\binom{n}{k} = \binom{4}{2} = 6$ destination nodes which need to decode all of the information units.

traffic corresponding to the storage node content; (2) 6 destination nodes have the multicast demand. Therefore, the exact-repair problem is to design a network code which satisfies all of these 10 constraints. Specifically, designing the first component of the repair code corresponds to designing local encoding submatrices for the storage nodes, i.e., $\mathbf{A}_i$'s and $\mathbf{B}_i$'s. The second component corresponds to designing coding coefficients for the links between the storage nodes and repair nodes. Notice that as code parameters (n, k, d) get large, the number of constraints grows exponentially, thereby making the problem harder.

4.2.3 Related Work

As stated earlier, Regenerating Codes, which cover an entire spectrum of optimal tradeoffs between repair bandwidth and storage cost, were introduced in [21, 73]. As discussed, Repair MDS codes (also called MSR codes) occupy one end of this spectrum corresponding to minimum storage. At the other end of the spectrum live Minimum Bandwidth Regenerating (MBR) repair codes corresponding to minimum repair bandwidth. The optimal tradeoffs described in [21, 73] are based on random-network-coding based approaches, which guarantee only functional repair.

The topic of exact-repair codes has received attention in the recent literature [72, 52, 54, 20, 70]. Wu and Dimakis in [72] showed that the MSR point (4.1) can be attained for the cases of: $k = 2$ and $k = n - 1$. Rashmi-Shah-Kumar-Ramchandran in [52] showed that for $d = n - 1$, the optimal MBR point can be achieved with a deterministic scheme requiring a small finite-field size and zero repair-coding-cost. Subsequently, Shah-Rashmi-Kumar-Ramchandran in [54] developed partial exact-repair codes for the MSR point corresponding to $\frac{k}{n} \leq \frac{1}{2} + \frac{2}{n}$, where exact repair is limited to the systematic component of the code. See Fig. 4.3. Finding the fundamental limits under exact repair of *all* nodes (including parity) remained an open problem. The first contribution of this work is to resolve this open problem by construction Exact-Repair MDS codes that attain the optimal tradeoff of (4.1) for the case of $\frac{k}{n} \leq \frac{1}{2}$ and $d \geq 2k - 1$. Here we assume that d helper nodes participating in the repair contain all of the survivor systematic nodes. The second contribution is to show the existence of Exact-Repair codes that achieves the optimal tradeoff (4.1) for all admissible values of (n, k, d).

The constructive framework proposed in [54] forms the inspiration for our first result. Indeed, we show that the partial exact-repair code introduced in [4] (meant for exact repair of the systematic nodes only) can also be used to repair the non-systematic (parity) node failures exactly, provided the second component of the repair code (i.e., the projection vectors needed for node repair) are appropriately designed. Designing the projection-vectors of exact repair codes is challenging and had remained an open problem: resolving this for the case of $\frac{k}{n} \leq \frac{1}{2}$ and $d \geq 2k - 1$ is our contribution. We also provide the systematic development of a family of code structures. This family of codes provides conceptual insights into the structure of solutions for the exact repair problem, while also offering a new large constructive design space of solutions.

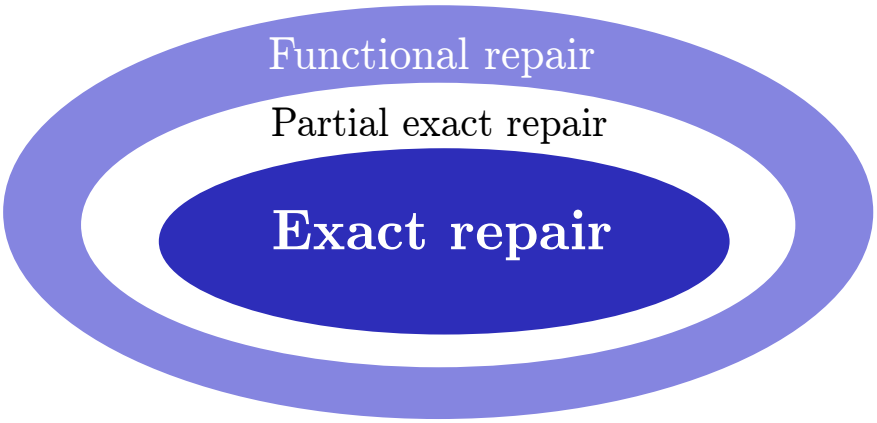


Figure 4.3: Repair models for distributed storage systems. In exact-repair, the failed nodes are exactly regenerated, thus restoring lost encoded fragments with their exact replicas. In functional-repair, the requirement is relaxed: the newly generated node can contain different data from that of the failed node as long as the repaired system maintains the MDS-code property. In partial exact-repair, only systematic nodes are repaired exactly, while parity nodes are repaired only functionally.

The interference alignment scheme by Cadambe and Jafar [14] that permits an arbitrarily large number of symbol extensions forms the basis of our second result. Building on the connection described in [59] between the wireless interference channel problem and the storage repair problem, we leverage the scheme introduced in [14] for our exact-repair problem, showing the existence of Exact-Repair MDS codes that achieve minimum repair bandwidth (matching the cutset lower bound) for all admissible values of (n, k, d).

4.3 Role of Interference Alignment

Network coding [5, 40, 34] (that allows multiple messages to be combined at network nodes) has been established recently as a useful tool for addressing interference issues even in wireline networks where all the communication links are orthogonal and non-interfering. This attribute was first observed in [72], where it was shown that interference alignment could be exploited for storage networks, specifically for Exact-Repair MDS codes having small k ($k = 2$). However, generalizing interference alignment to large values of k (even $k = 3$) proves to be challenging, as we describe in the sequel. In order to appreciate this better, let us first review the scheme of [72] that was applied to the exact repair problem. We will then address the difficulty of extending interference alignment for larger systems and describe how to address this in Sections 4.4 and 4.5.

$(4, 2)$ **Exact-Repair MDS Codes:** Fig. 4.4 illustrates an interference alignment scheme for a $(4, 2, 3)$ Exact-Repair MDS code defined over $\mathsf{GF}(5)$. First one can easily check the MDS

property of the code, i.e., all the source files can be reconstructed from any $k(=2)$ nodes out of $n(=4)$ nodes. Let us see how failed node 1 (storing (a_1, a_2)) can be exactly repaired. Assume a source file size $\mathcal{M}$ is 4 and repair-bandwidth-per-link $\frac{\gamma}{d} = 1$. The cutset bound (4.1) then gives the fundamental limits of storage cost $\alpha = 2$.

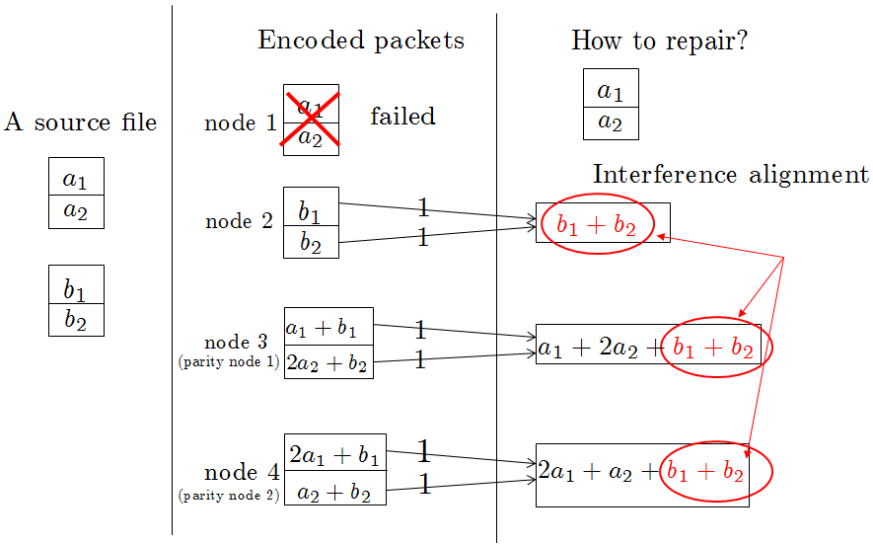


Figure 4.4: Interference alignment for a $(4, 2, 3)$ Exact-Repair MDS code defined over GF(5) [72]. Designing appropriate projection vectors, we can align interference space of (b_1, b_2) into one-dimensional linear space spanned by $[1,\ 1]^t$. As a result, we can successfully decode 2 desired unknowns (a_1, a_2) from 3 equations containing 4 unknowns (a_1, a_2, b_1, b_2).

The example illustrated in Fig. 4.4 shows that the parameter set described above is achievable using interference alignment. Here is a summary of the scheme. First notice that since the bandwidth-per-link is 1, each survivor node uses a projection vector to project its data into a scalar. Choosing appropriate projection vectors, we get the equations: $(b_1 + b_2)$; $a_1+2a_2+(b_1+b_2)$; $2a_1+a_2+(b_1+b_2)$. Observe that the undesired signals (b_1, b_2) (interference) are aligned onto an 1-dimensional linear subspace, thereby achieving interference alignment. Therefore, we can successfully decode (a_1, a_2) with three equations although there are four unknowns. Similarly, we can repair (b_1, b_2) when it has failed.

For parity node repair, a remapping technique is introduced. The idea is to define parity node symbols with new variables as follows:

$$\text{Node 3: } a'_1 := a_1 + b_1; \quad a'_2 := 2a_2 + b_2;$$
$$\text{Node 4: } b'_1 := 2a_1 + b_1; \quad b'_2 := a_2 + b_2.$$

We can then rewrite (a_1, a_2) and (b_1, b_2) with respect to (a'_1, a'_2) and (b'_1, b'_2). In terms of prime notation, parity nodes turn into systematic nodes and vice versa. With this remapping, one can easily design projection vectors for exact repair of parity nodes.

Geometric Interpretation: Using matrix notation, we provide geometric interpretation of interference alignment for the same example in Fig. 4.4. Let $\mathbf{a} = (a_1, a_2)^t$ and $\mathbf{b} = (b_1, b_2)^t$ be 2-dimensional information-unit vectors. Let $\mathbf{A}_i$ and $\mathbf{B}_i$ be 2-by-2 encoding submatrices for parity node i ($i = 1, 2$). Define 2-dimensional projection vectors $\mathbf{v}_{\alpha i}$'s ($i = 1, 2, 3$).

Let us consider exact repair of systematic node 1. By connecting to three nodes, we get: $\mathbf{b}^t\mathbf{v}_{\alpha 1}$; $\mathbf{a}^t(\mathbf{A}_1\mathbf{v}_{\alpha 2}) + \mathbf{b}^t(\mathbf{B}_1\mathbf{v}_{\alpha 2})$; $\mathbf{a}^t(\mathbf{A}_2\mathbf{v}_{\alpha 3}) + \mathbf{b}^t(\mathbf{B}_2\mathbf{v}_{\alpha 3})$. Recall the goal of decoding 2 desired unknowns out of 3 equations including 4 unknowns. To achieve this goal, we need:

$$\mathsf{rank}\left(\begin{bmatrix} (\mathbf{A}_1\mathbf{v}_{\alpha 2})^t \\ (\mathbf{A}_2\mathbf{v}_{\alpha 3})^t \end{bmatrix}\right) = 2; \quad \mathsf{rank}\left(\begin{bmatrix} \mathbf{v}_{\alpha 1}^t \\ (\mathbf{B}_1\mathbf{v}_{\alpha 2})^t \\ (\mathbf{B}_2\mathbf{v}_{\alpha 3})^t \end{bmatrix}\right) = 1. \tag{4.2}$$

The second condition can be met by setting $\mathbf{v}_{\alpha 2} = \mathbf{B}_1^{-1}\mathbf{v}_{\alpha 1}$ and $\mathbf{v}_{\alpha 3} = \mathbf{B}_2^{-1}\mathbf{v}_{\alpha 1}$. This choice forces the interference space to be collapsed into a one-dimensional linear subspace, thereby achieving interference alignment. With this setting, the first condition now becomes

$$\mathsf{rank}\left(\begin{bmatrix}\mathbf{A}_1\mathbf{B}_1^{-1}\mathbf{v}_{\alpha 1} & \mathbf{A}_2\mathbf{B}_2^{-1}\mathbf{v}_{\alpha 1}\end{bmatrix}\right) = 2. \tag{4.3}$$

It can be easily verified that the choice of $\mathbf{A}_i$'s and $\mathbf{B}_i$'s given in Figs. 4.4 and 4.5 guarantees the above condition. When the node 2 fails, we get a similar condition:

$$\mathsf{rank}\left(\begin{bmatrix}\mathbf{B}_1\mathbf{A}_1^{-1}\mathbf{v}_{\beta 1} & \mathbf{B}_2\mathbf{A}_2^{-1}\mathbf{v}_{\beta 1}\end{bmatrix}\right) = 2, \tag{4.4}$$

where $\mathbf{v}_{\beta i}$'s denote projection vectors for node 2 repair. This condition also holds under the given choice of encoding matrices. With this remapping, one can easily design projection vectors for exact repair of parity nodes.

Connection To the Wireless Interference Channel Problem: Observe the three equations shown in Fig. 4.5:

$$\underbrace{\begin{bmatrix} \mathbf{0} \\ (\mathbf{A}_1\mathbf{v}_{\alpha 2})^t \\ (\mathbf{A}_2\mathbf{v}_{\alpha 3})^t \end{bmatrix}\mathbf{a}}_{desired\ signals} + \underbrace{\begin{bmatrix} \mathbf{v}_{\alpha 1}^t \\ (\mathbf{B}_1\mathbf{v}_{\alpha 2})^t \\ (\mathbf{B}_2\mathbf{v}_{\alpha 3})^t \end{bmatrix}\mathbf{b}}_{interference}.$$

Separating into two parts, we can view this problem as a wireless communication problem, wherein a subset of the information is desired to be decoded in the presence of interference. Note the following analogy for the terms of $\mathbf{A}_1$ and $\mathbf{v}_{\alpha 2}$.

	Storage Repair	Wireless Problem
$\mathbf{A}_1$:	Encoding Submatix	Wireless Channel
$\mathbf{v}_{\alpha 2}$:	Projection Vector	Beamforming Vector

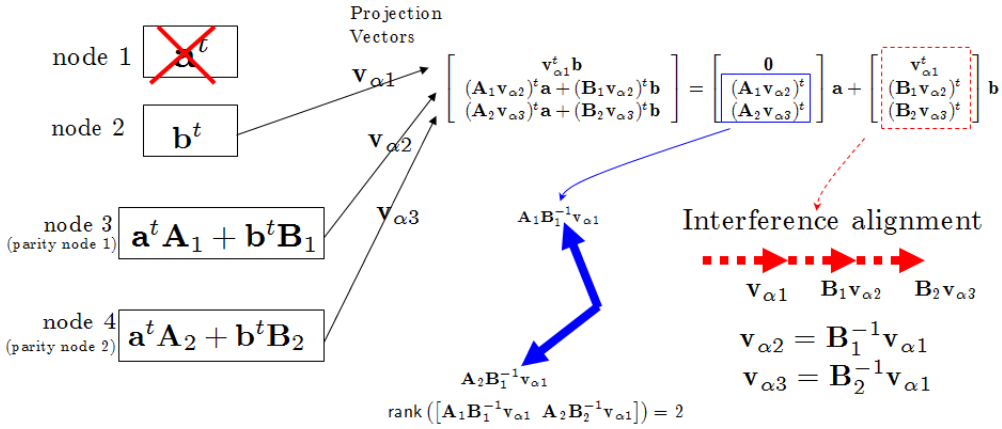


Figure 4.5: Geometric interpretation of interference alignment. The blue solid-line and red dashed-line vectors indicate linear subspaces with respect to "**a**" and "**b**", respectively. The choice of $\mathbf{v}_{\alpha 2} = \mathbf{B}_1^{-1}\mathbf{v}_{\alpha 1}$ and $\mathbf{v}_{\alpha 3} = \mathbf{B}_2^{-1}\mathbf{v}_{\alpha 1}$ enables interference alignment. For the specific example of Fig. 4.4, the corresponding encoding submatrices are $\mathbf{A}_1 = [1, 0; 0, 2]$, $\mathbf{B}_1 = [1, 0; 0, 1]$. $\mathbf{A}_2 = [2, 0; 0, 1]$, $\mathbf{B}_2 = [1, 0; 0, 1]$.

The matrix $\mathbf{A}_1$ and vector $\mathbf{v}_{\alpha 2}$ correspond respectively to the channel matrix and beamforming vector in the wireless problem.

There are, however, significant differences. In the wireless communication problem, the channel matrices are provided by nature and therefore not controllable. The transmission strategy alone (vector variables) can be controlled for achieving interference alignment. On the other hand, in our storage repair problems, both matrices and vectors are controllable, i.e., projection vectors and encoding submatrices can be arbitrarily designed, resulting in more flexibility. However, our storage repair problem comes with unparalleled challenges due to the MDS requirement and the multiple failure configurations. These induce multiple interference alignment constraints that need to be simultaneously satisfied. What makes this difficult is that the encoding submatrices, once designed, must be the same for all repair configurations. This is particularly acute for large values of k (even $k = 3$), as the number of possible failure configurations increases with n (which increases with k).

Remark 16 (Benefits of the Storage-Code Design Flexibility)**.** *As a side-note, we emphasize the great potential to make use of the storage-code design flexibility in achieving interference alignment. In addition to the work [59], a recent work [13] shows this potential as well. Specifically [13] exploits the so-called subspace interference alignment [60] - which uses a finite number of symbol extensions but faces some constraint on wireless channel structures - to the storage-repair problem which is free from the constraint, thereby developing a practical code construction in the exact repair problem.* □

4.4 Framework 1: Code Construction

We propose a *common-eigenvector* based constructive design framework to address the exact repair problem. This framework draws its inspiration from the work in [54] which guarantees the exact repair of systematic nodes, while satisfying the MDS code property, but which does not provide exact repair of failed parity nodes. In providing a constructive solution for the exact repair of *all* nodes, we use geometric insights to propose a large family of repair codes. This both provides insights into the structure of codes for exact repair of all nodes (particularly the projection-vectors code component), as well as opens up a rich and large design space for constructive solutions. Specifically, we propose a common-eigenvector based approach building on a certain *elementary matrix* property [35, 11]. This structure provides the key geometric insights needed to facilitate the design of the key projection-vectors code component of exact repair codes. Moreover, our proposed coding schemes are deterministic and constructive, requiring a symbol alphabet-size of at most $(2n - 2k)$.

Our framework consists of four components: (1) developing a family of codes[5] for exact repair of systematic codes based on the common-eigenvector concept; (2) drawing a *dual* relationship between the systematic and parity node repair; (3) guaranteeing the MDS-code property; (4) constructing codes with finite-field alphabets. Step (2) of our framework is a significant distinction from that of [54] and is needed to tackle the full exact repair problem not addressed there. The framework covers the case of $n \geq 2k$ and $d \geq 2k - 1$. It turns out that the $(2k, k, 2k - 1)$ code case contains the key design ingredients and the case of $n \geq 2k$ and $d \geq 2k - 1$ can be derived from this (see Section 4.4.4). Hence, we first focus on the simplest example: $(6, 3, 5)$ Exact-Repair MDS codes. Later in Section 4.4.4, we will generalize this to arbitrary (n, k, d) repair codes in the class.

4.4.1 Systematic Node Repair

For $k \geq 3$ (more-than-two interfering information units), achieving interference alignment for exact repair turns out to be significantly more complex than the $k = 2$ case. Fig. 4.6 illustrates this difficulty through the example of repairing node 1 for a $(6, 3, 5)$ code. By the optimal tradeoff (4.1), the choice of $\mathcal{M} = 9$ and $\frac{\gamma}{d} = 1$ gives $\alpha = 3$. Let $\mathbf{a} = (a_1, a_2, a_3)^t$, $\mathbf{b} = (b_1, b_2, b_3)^t$ and $\mathbf{c} = (c_1, c_2, c_3)^t$. We define 3-by-3 encoding submatrices of $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{C}_i$ (for $i = 1, 2, 3$); and 3-dimensional projection vectors $\mathbf{v}_{\alpha i}$'s.

[5]Recall that our repair code consists of two components: (1) the encoding (generator) matrix; (2) the projection vectors needed for node repair. Interestingly, the encoding matrix component of the code in [54] turns out to work for the exact repair of both systematic and parity nodes provided the second component of the repair code (projection vectors needed for repair) are appropriately designed.

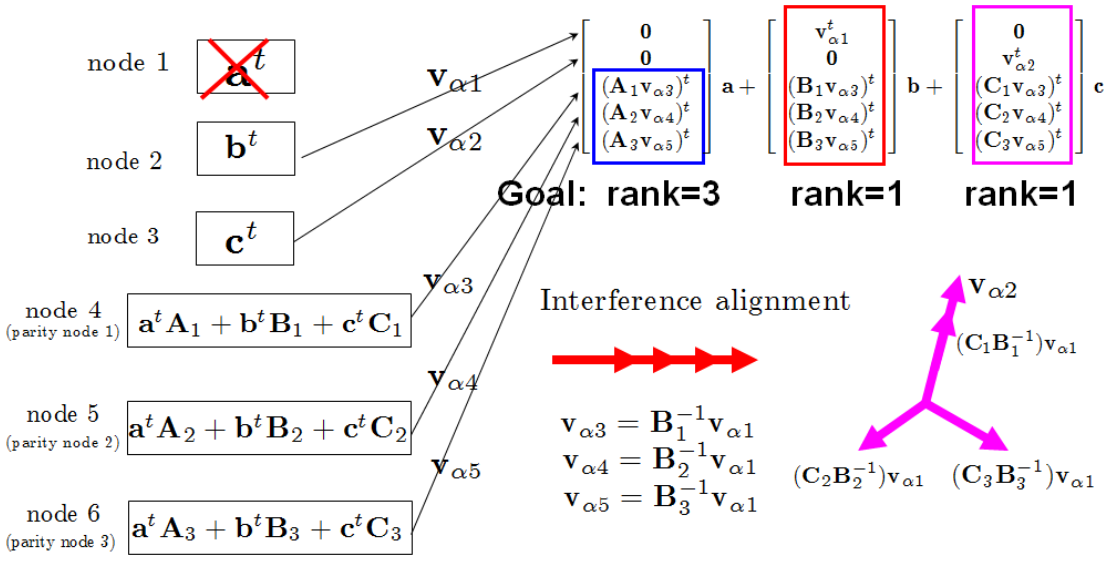


Figure 4.6: Difficulty of achieving simultaneous interference alignment.

Consider the 5 ($= d$) equations downloaded from the nodes:

$$\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ (\mathbf{A}_1\mathbf{v}_{\alpha 3})^t \\ (\mathbf{A}_2\mathbf{v}_{\alpha 4})^t \\ (\mathbf{A}_3\mathbf{v}_{\alpha 5})^t \end{bmatrix} \mathbf{a} + \begin{bmatrix} \mathbf{v}_{\alpha 1}^t \\ \mathbf{0} \\ (\mathbf{B}_1\mathbf{v}_{\alpha 3})^t \\ (\mathbf{B}_2\mathbf{v}_{\alpha 4})^t \\ (\mathbf{B}_3\mathbf{v}_{\alpha 5})^t \end{bmatrix} \mathbf{b} + \begin{bmatrix} \mathbf{0} \\ \mathbf{v}_{\alpha 2}^t \\ (\mathbf{C}_1\mathbf{v}_{\alpha 3})^t \\ (\mathbf{C}_2\mathbf{v}_{\alpha 4})^t \\ (\mathbf{C}_3\mathbf{v}_{\alpha 5})^t \end{bmatrix} \mathbf{c}.$$

In order to successfully recover the desired signal components of "**a**", the matrices associated with **b** and **c** should have rank 1, respectively, while the matrix associated with **a** should have full rank of 3. In accordance with the $(4, 2, 3)$ code example in Fig. 4.5, if one were to set $\mathbf{v}_{\alpha 3} = \mathbf{B}_1^{-1}\mathbf{v}_{\alpha 1}$, $\mathbf{v}_{\alpha 4} = \mathbf{B}_2^{-1}\mathbf{v}_{\alpha 1}$ and $\mathbf{v}_{\alpha 5} = \mathbf{B}_3^{-1}\mathbf{v}_{\alpha 1}$, then it is possible to achieve interference alignment with respect to **b**. However, this choice also specifies the interference space of **c**. If the $\mathbf{B}_i$'s and $\mathbf{C}_i$'s are not designed judiciously, interference alignment is not guaranteed for **c**. Hence, it is not evident how to achieve interference alignment at the same time.

In order to address the challenge of simultaneous interference alignment, we invoke a common eigenvector concept. The idea consists of two parts: (i) designing the $(\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i)$'s such that $\mathbf{v}_1$ is a common eigenvector of the $\mathbf{B}_i$'s and $\mathbf{C}_i$'s, but not of $\mathbf{A}_i$'s[6]; (ii) repairing by having survivor nodes *project* their data onto a linear subspace spanned by this common eigenvector $\mathbf{v}_1$. We can then achieve interference alignment for **b** and **c** at the same time,

[6]Of course, five additional constraints also need to be satisfied for the other five failure configurations for this $(6, 3, 5)$ code example.

by setting $\mathbf{v}_{\alpha i} = \mathbf{v}_1, \forall i$. As long as $[\mathbf{A}_1\mathbf{v}_1, \mathbf{A}_2\mathbf{v}_1, \mathbf{A}_3\mathbf{v}_1]$ is invertible, we can also guarantee the decodability of $\mathbf{a}$. See Fig. 4.7.

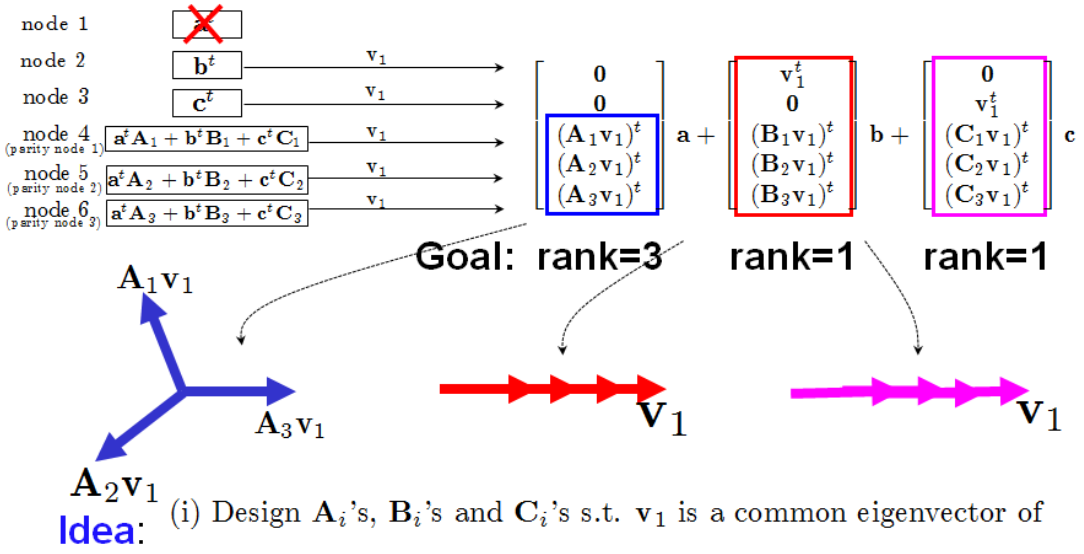


Figure 4.7: Illustration of exact repair of systematic node 1 for $(6, 3, 5)$ exact-repair MDS codes. The idea consists of two parts: (i) designing $(\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i)$'s such that $\mathbf{v}_1$ is a common eigenvector of the $\mathbf{B}_i$'s and $\mathbf{C}_i$'s, but not of $\mathbf{A}_i$'s; (ii) repairing by having survivor nodes project their data onto a linear subspace spanned by this common eigenvector $\mathbf{v}_1$.

The challenge is now to design encoding submatrices to guarantee the existence of a common eigenvector while also satisfying the decodability of desired signals. The difficulty comes from the fact that in our $(6, 3, 5)$ repair code example, these constraints need to be satisfied for *all* six possible failure configurations. The structure of elementary matrices [35, 11] (generalized matrices of Householder and Gauss matrices) gives insights into this. To see this, consider a 3-by-3 elementary matrix $\mathbf{A}$:

$$\mathbf{A} = \mathbf{u}\mathbf{v}^t + \alpha\mathbf{I}, \tag{4.5}$$

where $\mathbf{u}$ and $\mathbf{v}$ are 3-dimensional vectors. Here is an observation that motivates our proposed structure: the dimension of the null space of $\mathbf{v}$ is 2 and the null vector $\mathbf{v}^\perp$ is an eigenvector

of $\mathbf{A}$, i.e., $\mathbf{A}\mathbf{v}^{\perp} = \alpha\mathbf{v}^{\perp}$. This motivates the following structure:

$$\begin{aligned}
&\mathbf{A}_1 = \mathbf{u}_1\mathbf{v}_1^t + \alpha_1\mathbf{I};\ \mathbf{B}_1 = \mathbf{u}_1\mathbf{v}_2^t + \beta_1\mathbf{I};\ \mathbf{C}_1 = \mathbf{u}_1\mathbf{v}_3^t + \gamma_1\mathbf{I}\\
&\mathbf{A}_2 = \mathbf{u}_2\mathbf{v}_1^t + \alpha_2\mathbf{I};\ \mathbf{B}_2 = \mathbf{u}_2\mathbf{v}_2^t + \beta_2\mathbf{I};\ \mathbf{C}_2 = \mathbf{u}_2\mathbf{v}_3^t + \gamma_2\mathbf{I}\\
&\mathbf{A}_3 = \mathbf{u}_3\mathbf{v}_1^t + \alpha_3\mathbf{I};\ \mathbf{B}_3 = \mathbf{u}_3\mathbf{v}_2^t + \beta_3\mathbf{I};\ \mathbf{C}_3 = \mathbf{u}_3\mathbf{v}_3^t + \gamma_3\mathbf{I},
\end{aligned} \tag{4.6}$$

where $\mathbf{v}_i$'s are 3-dimensional linearly independent vectors and so are $\mathbf{u}_i$'s. The values of the α_i's, β_i's and γ_i's can be arbitrary non-zero values. First consider the simple design where the $\mathbf{v}_i$'s are *orthonormal*. This is for conceptual simplicity. Later we will generalize to the case where the $\mathbf{v}_i$'s need not be orthogonal but only linearly independent. We see that for $i = 1, 2, 3$,

$$\begin{aligned}
\mathbf{A}_i\mathbf{v}_1 &= \alpha_i\mathbf{v}_1 + \mathbf{u}_i,\\
\mathbf{B}_i\mathbf{v}_1 &= \beta_i\mathbf{v}_1,\\
\mathbf{C}_i\mathbf{v}_1 &= \gamma_i\mathbf{v}_1.
\end{aligned} \tag{4.7}$$

Importantly, notice that $\mathbf{v}_1$ is a common eigenvector of the $\mathbf{B}_i$'s and $\mathbf{C}_i$'s, while simultaneously ensuring that the vectors of $\mathbf{A}_i\mathbf{v}_1$ are linearly independent. Hence, setting $\mathbf{v}_{\alpha i} = \mathbf{v}_1$ for all i, it is possible to achieve simultaneous interference alignment while also guaranteeing the decodability of the desired signals. See Fig. 4.7. On the other hand, this structure also guarantees exact repair for $\mathbf{b}$ and $\mathbf{c}$. We use $\mathbf{v}_2$ for exact repair of $\mathbf{b}$. It is a common eigenvector of the $\mathbf{C}_i$'s and $\mathbf{A}_i$'s, while ensuring $[\mathbf{B}_1\mathbf{v}_2, \mathbf{B}_2\mathbf{v}_2, \mathbf{B}_3\mathbf{v}_2]$ invertible. Similarly, $\mathbf{v}_3$ is used for exact repair of $\mathbf{c}$.

We will see that a *dual basis* property gives insights into the general case where $\{\mathbf{v}\} := (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ is not orthogonal but only linearly independent. In this case, defining a dual basis $\{\mathbf{v}'\} := (\mathbf{v}'_1, \mathbf{v}'_2, \mathbf{v}'_3)$ gives the solution:

$$\left[\begin{array}{c} \mathbf{v}_1'^t \\ \hline \mathbf{v}_2'^t \\ \hline \mathbf{v}_3'^t \end{array}\right] := \left[\ \mathbf{v}_1 \ \middle|\ \mathbf{v}_2 \ \middle|\ \mathbf{v}_3\ \right]^{-1}.$$

The definition gives the following property: $\mathbf{v}_i'^t\mathbf{v}_j = \delta(i-j), \forall i, j$. Using this property, one can see that $\mathbf{v}'_1$ is a common eigenvector of the $\mathbf{B}_i$'s and $\mathbf{C}_i$'s while ensuring the invertibility of the desired signals $\mathbf{a}$:

$$\begin{aligned}
\mathbf{A}_i\mathbf{v}'_1 &= \alpha_i\mathbf{v}'_1 + \mathbf{u}_i,\\
\mathbf{B}_i\mathbf{v}'_1 &= \beta_i\mathbf{v}'_1,\\
\mathbf{C}_i\mathbf{v}'_1 &= \gamma_i\mathbf{v}'_1.
\end{aligned} \tag{4.8}$$

So it can be used as a projection vector for exact repair of $\mathbf{a}$. Similarly, we can use $\mathbf{v}'_2$ and $\mathbf{v}'_3$ for exact repair of $\mathbf{b}$ and $\mathbf{c}$, respectively.

4.4.2 Parity Node Repair

We have seen so far how to ensure exact repair of the systematic nodes. We have known that if $\{\mathbf{v}\}$ is linearly independent and so $\{\mathbf{u}\}$ is, then using the structure of (4.6) together with projection vectors enables repair, for arbitrary values of $(\alpha_i, \beta_i, \gamma_i)$'s. A natural question is now: will this structure also guarantee exact repair of parity nodes? It turns out that for exact repair of all nodes, we need a special relationship between $\{\mathbf{v}\}$ and $\{\mathbf{u}\}$ through the correct choice of the $(\alpha_i, \beta_i, \gamma_i)$'s.

We will show that parity nodes can be repaired by drawing a *dual* relationship with systematic nodes. The procedure has two steps. The first is to remap parity nodes with $\mathbf{a}'$, $\mathbf{b}'$, and $\mathbf{c}'$, respectively:

$$\begin{bmatrix} \mathbf{a}' \\ \mathbf{b}' \\ \mathbf{c}' \end{bmatrix} := \begin{bmatrix} \mathbf{A}_1^t & \mathbf{B}_1^t & \mathbf{C}_1^t \\ \mathbf{A}_2^t & \mathbf{B}_2^t & \mathbf{C}_2^t \\ \mathbf{A}_3^t & \mathbf{B}_3^t & \mathbf{C}_3^t \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \\ \mathbf{c} \end{bmatrix}.$$

Systematic nodes can then be rewritten in terms of the prime notations:

$$\begin{aligned} \mathbf{a}^t &= \mathbf{a}'^t\mathbf{A}_1' + \mathbf{b}'^t\mathbf{B}_1' + \mathbf{c}'^t\mathbf{C}_1', \\ \mathbf{b}^t &= \mathbf{a}'^t\mathbf{A}_2' + \mathbf{b}'^t\mathbf{B}_2' + \mathbf{c}'^t\mathbf{C}_2', \\ \mathbf{c}^t &= \mathbf{a}'^t\mathbf{A}_3' + \mathbf{b}'^t\mathbf{B}_3' + \mathbf{c}'^t\mathbf{C}_3', \end{aligned} \tag{4.9}$$

where the newly mapped encoding submatrices $(\mathbf{A}_i', \mathbf{B}_i', \mathbf{C}_i)$'s are defined as:

$$\begin{bmatrix} \mathbf{A}_1' & \mathbf{A}_2' & \mathbf{A}_3' \\ \mathbf{B}_1' & \mathbf{B}_2' & \mathbf{B}_3' \\ \mathbf{C}_1' & \mathbf{C}_2' & \mathbf{C}_3' \end{bmatrix} := \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \mathbf{A}_3 \\ \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 \\ \mathbf{C}_1 & \mathbf{C}_2 & \mathbf{C}_3 \end{bmatrix}^{-1}. \tag{4.10}$$

With this remapping, one can dualize the relationship between systematic and parity node repair. Specifically, if all of the $\mathbf{A}_i'$'s, $\mathbf{B}_i'$'s, and $\mathbf{C}_i'$'s are *elementary matrices* and form a similar structure as in (4.6), exact repair of the parity nodes becomes transparent.

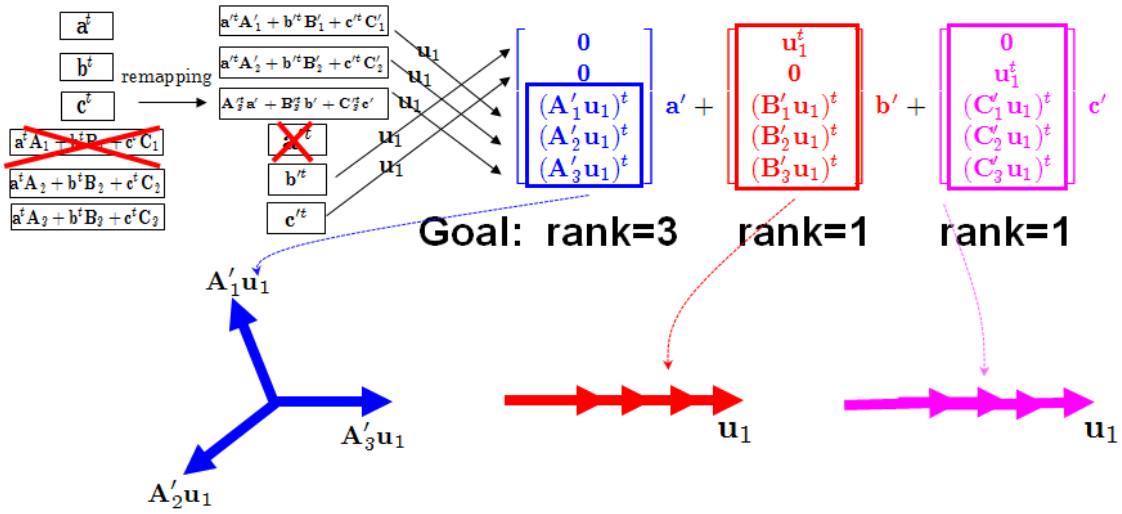
The challenge is now how to guarantee the dual structure. In Lemma 3, we show that a special relationship between $\{\mathbf{u}\}$ and $\{\mathbf{v}\}$ through $(\alpha_i,\ \beta_i,\ \gamma_i)$'s can guarantee this dual relationship of (4.13).

Lemma 3. *Suppose*

$$\mathbf{P} := \left[\begin{array}{c|c|c} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{array}\right] \text{ is invertible.} \tag{4.11}$$

Also assume

$$\kappa\mathbf{U} = \mathbf{V}'\mathbf{P}. \tag{4.12}$$



We provide sufficient conditions to ensure the *dual* structure:

$$(i)\ \kappa[\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3] = [\mathbf{v}'_1, \mathbf{v}'_2, \mathbf{v}'_3] \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix};$$

$$(ii)\ \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix} \text{ is invertible.}$$

Figure 4.8: Exact repair of parity node 1 for a $(6, 3, 5)$ exact-repair MDS code. The idea is to construct the *dual* structure of (4.13) by remapping parity nodes and then adding sufficient conditions of (4.11) and (4.12).

where $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]$, $\mathbf{V}' = [\mathbf{v}'_1, \mathbf{v}'_2, \mathbf{v}'_3]$, $\{\mathbf{v}'\} := \{\mathbf{v}'_1, \mathbf{v}'_2, \mathbf{v}'_3\}$ *is the dual basis of* $\{\mathbf{v}\}$, *i.e.,* $\mathbf{v}'^t_i \mathbf{v}_j = \delta(i-j)$ *and* κ *is an arbitrary non-zero value s.t.* $1 - \kappa^2 \neq 0$. *Then, we can obtain the following structure dual to (4.6):*

$$\begin{aligned}
\mathbf{A}'_1 &= \frac{1}{1-\kappa^2}\left(\mathbf{v}'_1\mathbf{u}'^t_1 - \kappa^2\alpha'_1\mathbf{I}\right); \mathbf{B}'_1 = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_1\mathbf{u}'^t_2 - \kappa^2\alpha'_2\mathbf{I}\right); \mathbf{C}'_1 = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_1\mathbf{u}'^t_3 - \kappa^2\alpha'_3\mathbf{I}\right) \\
\mathbf{A}'_2 &= \frac{1}{1-\kappa^2}\left(\mathbf{v}'_2\mathbf{u}'^t_1 - \kappa^2\beta'_1\mathbf{I}\right); \mathbf{B}'_2 = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_2\mathbf{u}'^t_2 - \kappa^2\beta'_2\mathbf{I}\right); \mathbf{C}'_2 = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_2\mathbf{u}'^t_3 - \kappa^2\beta'_3\mathbf{I}\right) \\
\mathbf{A}'_3 &= \frac{1}{1-\kappa^2}\left(\mathbf{v}'_3\mathbf{u}'^t_1 - \kappa^2\gamma'_1\mathbf{I}\right); \mathbf{B}'_3 = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_3\mathbf{u}'^t_2 - \kappa^2\gamma'_2\mathbf{I}\right); \mathbf{C}'_3 = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_3\mathbf{u}'^t_3 - \kappa^2\gamma'_3\mathbf{I}\right),
\end{aligned} \tag{4.13}$$

where $\{\mathbf{u}'\}$ *is the dual basis of* $\{\mathbf{u}\}$, *i.e.,* $\mathbf{u}'^t_i \mathbf{u}_j = \delta(i-j)$ *and* $(\alpha'_i, \beta'_i, \gamma'_i)$*'s are the dual basis vectors of* $(\alpha_i, \beta_i, \gamma_i)$*'s, i.e.,* $< (\alpha'_i, \beta'_i, \gamma'_i), (\alpha_j, \beta_j, \gamma_j) >= \delta(i-j)$:

$$\left[\begin{array}{ccc} \alpha'_1 & \beta'_1 & \gamma'_1 \\ \hline \alpha'_2 & \beta'_2 & \gamma'_2 \\ \hline \alpha'_3 & \beta'_3 & \gamma'_3 \end{array}\right] := \left[\begin{array}{c|c|c} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{array}\right]^{-1}. \tag{4.14}$$

Proof. See Appendix 4.7.1. □

Remark 17. *The dual structure (4.13) now gives projection vector solutions for parity node repair. For exact repair of parity node 1, we can use vector* $\mathbf{u}_1$ *(a common eigenvector of the* $\mathbf{B}'_i$*'s and* $\mathbf{C}'_i$*'s), since it enables simultaneous interference alignment for* $\mathbf{b}'$ *and* $\mathbf{c}'$, *while ensuring the decodability of* $\mathbf{a}'$. *See Fig. 4.8. Notice that more conditions of (4.11) and (4.12) are added to ensure exact repair of all nodes, while these conditions were unnecessary for exact repair of systematic nodes only. Also note that these are only sufficient conditions.*

Remark 18. *Note that the dual structure (4.13) is quite similar to the primary structure (4.6). The only difference is that in the dual structure,* $\{\mathbf{u}\}$ *and* $\{\mathbf{v}\}$ *are interchanged to form a* transpose-like *structure. This reveals insights into how to design projection vectors for exact repair of parity nodes in a transparent manner.*

4.4.3 MDS-Code Property

The third part of the framework is to guarantee the MDS-code property, which allows us to identify specific constraints on the $(\alpha_i, \beta_i, \gamma_i)$'s and/or ($\{\mathbf{v}\}$, $\{\mathbf{u}\}$). Consider four cases, associated in the Data Collector (DC) who is intended in the source file data: (a) 3 systematic nodes; (b) 3 parity nodes; (c) 2 systematic and 1 parity nodes; (d) 1 systematic and 2 parity nodes.

The first is a trivial case. The second case has been already verified in the process of forming the dual structure (4.13). The invertibility condition of (4.11) together with (4.12) suffices to ensure the invertibility of the composite matrix $[\mathbf{A}_1\ \mathbf{A}_2\ \mathbf{A}_3; \mathbf{B}_1\ \mathbf{B}_2\ \mathbf{B}_3; \mathbf{C}_1\ \mathbf{C}_2\ \mathbf{C}_3]$.

The third case requires the invertibility of all of each encoding submatrix. In this case, it is necessary that the α_i's, β_i's and γ_i's are non-zero values; otherwise, each encoding submatrix has rank 1. Also the non-zero values together with (4.12) guarantee the invertibility of each encoding submatrix. To see this, for example, consider

$$\mathbf{V}^t\mathbf{A}_1\mathbf{V}' = (\mathbf{V}^t\mathbf{u}_1)\mathbf{e}_1^t + \alpha_1\mathbf{I} = \begin{bmatrix} \frac{\alpha_1}{\kappa}+\alpha_1 & 0 & 0 \\ \frac{\beta_1}{\kappa} & \alpha_1 & 0 \\ \frac{\gamma_1}{\kappa} & 0 & \alpha_1 \end{bmatrix},$$

where the second equality follows from $\mathbf{v}_1^t\mathbf{u}_1 = \frac{\alpha_1}{\kappa}$, $\mathbf{v}_2^t\mathbf{u}_1 = \frac{\beta_1}{\kappa}$ and $\mathbf{v}_3^t\mathbf{u}_1 = \frac{\gamma_1}{\kappa}$ due to (4.12). Here $\mathbf{e}_1$ indicates a standard basis, i.e., $\mathbf{e}_1 = (1,0,0)^t$. Clearly this resulting matrix is invertible. Since $\mathbf{V}$ is invertible, so is $\mathbf{A}_1$.

The last case requires some non-trivial work. Consider a specific example where the DC connects to nodes 3, 4 and 5. In this case, we first recover $\mathbf{c}$ from node 3 and subtract the terms associated with $\mathbf{c}$ from nodes 4 and 5. We then get:

$$\begin{bmatrix} \mathbf{a}^t & \mathbf{b}^t \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{B}_1 & \mathbf{B}_2 \end{bmatrix}. \tag{4.15}$$

Now consider

$$\begin{bmatrix} \mathbf{V}^t & \mathbf{0} \\ \mathbf{0} & \mathbf{V}^t \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{B}_1 & \mathbf{B}_2 \end{bmatrix} \begin{bmatrix} \mathbf{V}' & \mathbf{0} \\ \mathbf{0} & \mathbf{V}' \end{bmatrix} = \left[\begin{array}{ccc|ccc} \frac{\alpha_1}{\kappa}+\alpha_1 & 0 & 0 & \frac{\alpha_2}{\kappa}+\alpha_2 & 0 & 0 \\ \frac{\beta_1}{\kappa} & \alpha_1 & 0 & \frac{\beta_2}{\kappa} & \alpha_2 & 0 \\ \frac{\gamma_1}{\kappa} & 0 & \alpha_1 & \frac{\gamma_2}{\kappa} & 0 & \alpha_2 \\ \hline \beta_1 & \frac{\alpha_1}{\kappa} & 0 & \beta_2 & \frac{\alpha_2}{\kappa} & 0 \\ 0 & \beta_1+\frac{\beta_1}{\kappa} & 0 & 0 & \beta_2+\frac{\beta_2}{\kappa} & 0 \\ 0 & \frac{\gamma_1}{\kappa} & \beta_1 & 0 & \frac{\gamma_2}{\kappa} & \beta_2 \end{array}\right],$$

where the equality follows from the fact that $\mathbf{V}^t\mathbf{A}_i\mathbf{V}' = (\mathbf{V}^t\mathbf{u}_i)\mathbf{e}_1^t + \alpha_i\mathbf{I}$ and $\mathbf{V}^t\mathbf{B}_i\mathbf{V}' = (\mathbf{V}^t\mathbf{u}_i)\mathbf{e}_2^t + \beta_i\mathbf{I}$, for $i = 1, 2$. Using a Gaussian elimination method, one can now easily show that this resulting matrix is invertible and so is $[\mathbf{A}_1\ \mathbf{A}_2; \mathbf{B}_1\ \mathbf{B}_2]$ if

$$\mathbf{P}_2 := \left[\begin{array}{c|c} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \end{array}\right] \text{ is invertible.} \tag{4.16}$$

Considering the above 4 cases, the following condition together with (4.11) and (4.12) suffices for guaranteeing the MDS-code property:

$$\textit{Any submatrix of } \mathbf{P} \textit{ of (4.11) is invertible.} \tag{4.17}$$

Code Construction with Finite-Field Alphabets: The last part is to design $\mathbf{P}$ of (4.11) and $\{\mathbf{v}\} := (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ in (4.6) such that $\{\mathbf{v}\}$ is linearly independent and the conditions of (4.12) and (4.17) are satisfied. As for the matrices that satisfy (4.17), one

can think of a Cauchy matrix or a Vandermonde matrix [9, 54]. Specifically, we employ the Cauchy matrix to construct explicit codes with the guarantee on the minimum finite-field size. Notice that the Cauchy matrix is an example that guarantees (4.17). One may use any other matrices that satisfy (4.17).

Definition 1 (A Cauchy Matrix [9])**.** *A Cauchy matrix* $\mathbf{P}$ *is an* $m \times n$ *matrix with entries* p_{ij} *in the form:*

$$p_{ij} = \frac{1}{x_i - y_j}, \forall i = 1, \cdots, m, j = 1, \cdots n, x_i \neq y_j,$$

where x_i *and* y_j *are elements of a field and* $\{x_i\}$ *and* $\{y_j\}$ *are injective sequences, i.e., elements of the sequence are distinct.*

The injective property of $\{x_i\}$ and $\{y_j\}$ requires a finite field size of $2s$ for an s-by-s Cauchy matrix. Therefore, in our $(6, 3, 5)$ repair code example, the finite field size of 6 suffices. The field size condition for guaranteeing linear independence of $\{\mathbf{v}\}$ is more relaxed.

Using the structure of (4.6) and the conditions of (4.11), (4.12) and (4.17), we can now state the following theorem.

Theorem 6 ($(6, 3, 5)$ Exact-Repair MDS Codes)**.** *Suppose* $\mathbf{P}$ *of (4.11) is a Cauchy matrix, i.e., every submatrix of is invertible. Each element of* $\mathbf{P}$ *is in* $\mathsf{GF}(q)$ *and* $q \geq 6$*. Suppose encoding submatrices form the structure of (4.6),* $\{\mathbf{v}\} := (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ *is linearly independent, and* $\{\mathbf{u}\}$ *satisfies the condition of (4.12). Then, the repair code comprising the encoding matrix and the projection vectors achieves the optimal tradeoff of (4.1).*

We provide two numerical examples: (1) $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$ is orthogonal, e.g., $\mathbf{V} = \mathbf{I}$; (2) $\mathbf{U}$ is orthogonal, e.g., $\mathbf{U} = \mathbf{I}$. We will also discuss the complexity of repair construction schemes for each of these examples. It turns out that the first code has significantly lower complexity for exact repair of systematic nodes, as compared to that of parity nodes. On the other hand, the second case provides much simpler parity-node repair schemes instead. Depending on applications of interest, one can choose an appropriate code among our family of codes.

Example 1 ($\mathbf{V} = \mathbf{I}$): We present an example of $(6, 3, 5)$ Exact-Repair MDS codes defined over $\mathsf{GF}(5)$ where $\mathbf{V} = \mathbf{I}$ and

$$\mathbf{P} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 4 \end{bmatrix}, \mathbf{U} = \kappa^{-1}\mathbf{V}'\mathbf{P} = 3 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 4 \end{bmatrix} = \begin{bmatrix} 3 & 3 & 3 \\ 3 & 1 & 4 \\ 3 & 4 & 2 \end{bmatrix},$$

where $\mathbf{U}$ is set based on (4.12) and $\kappa = 2$. Notice that we employ a *non-Cauchy*-type matrix to construct a field-size $q = 5$ code (smaller than $q = 6$ required when using a Cauchy matrix). Remember that a Cauchy matrix provides only a sufficient condition for

ensuring the invertibility of any submatrices of $\mathbf{P}$. By (4.6) and (4.13), the primary and dual structures for encoding matrices are given by

$$\mathbf{G} = \left[\begin{array}{ccc|ccc|ccc} 4&0&0&4&0&0&4&0&0\\ 3&1&0&1&1&0&4&1&0\\ 3&0&1&4&0&1&2&0&1\\ \hline 1&3&0&2&3&0&3&3&0\\ 0&4&0&0&3&0&0&2&0\\ 0&3&1&0&4&2&0&2&3\\ \hline 1&0&3&3&0&3&4&0&3\\ 0&1&3&0&3&1&0&4&4\\ 0&0&4&0&0&2&0&0&1 \end{array}\right]; \mathbf{G}^{-1} = \left[\begin{array}{ccc|ccc|ccc} 4&1&4&3&0&0&2&0&0\\ 0&3&0&1&4&4&0&2&0\\ 0&0&3&0&0&3&1&1&1\\ \hline 4&2&2&1&0&0&1&0&0\\ 0&3&0&1&3&2&0&1&0\\ 0&0&3&0&0&1&1&2&3\\ \hline 1&2&4&1&0&0&2&0&0\\ 0&2&0&4&3&4&0&2&0\\ 0&0&2&0&0&1&4&2&1 \end{array}\right], \tag{4.18}$$

where

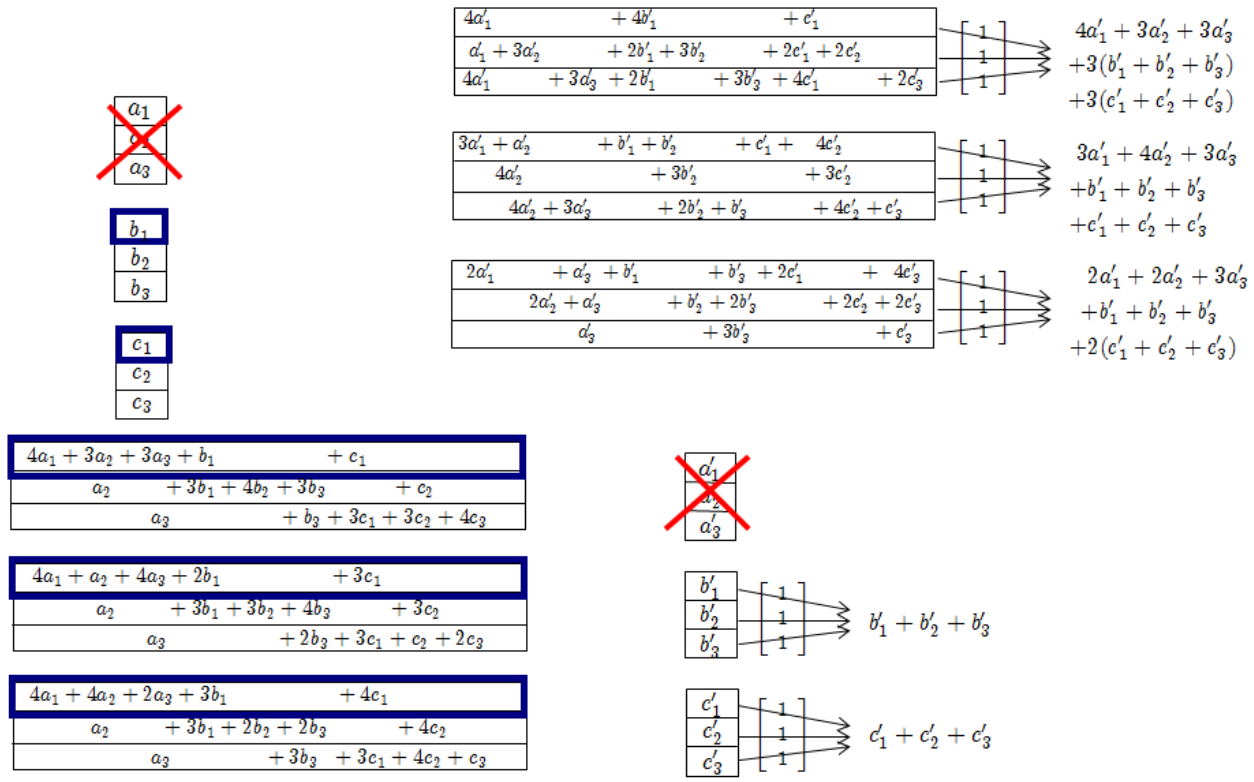
$$\mathbf{G} := \left[\begin{array}{ccc} \mathbf{A}_1&\mathbf{A}_2&\mathbf{A}_3\\ \mathbf{B}_1&\mathbf{B}_2&\mathbf{B}_3\\ \mathbf{C}_1&\mathbf{C}_2&\mathbf{C}_3 \end{array}\right]; \mathbf{G}^{-1} = \left[\begin{array}{ccc} \mathbf{A}'_1&\mathbf{A}'_2&\mathbf{A}'_3\\ \mathbf{B}'_1&\mathbf{B}'_2&\mathbf{B}'_3\\ \mathbf{C}'_1&\mathbf{C}'_2&\mathbf{C}'_3 \end{array}\right]. \tag{4.19}$$

Fig. 4.9 shows an example for exact repair of (a) systematic node 1 and (b) parity node 1. Note that the projection vector solution for systematic node repair is quite simple: $\mathbf{v}_{\alpha i} = \mathbf{v}_1 = (1,0,0)^t, \forall i$. We download only the first equation from each survivor node. Notice that the downloaded five equations contain only five unknown variables of $(a_1, a_2, a_3, b_1, c_1)$ and three equations associated with $\mathbf{a}$ are linearly independent. Hence, we can successfully recover $\mathbf{a}$.

On the other hand, exact repair of parity nodes seems non-straightforward. However, our framework provides quite a simple repair scheme: setting all of the projection vectors as $2^{-1}\mathbf{u}_1 = (1,1,1)^t$. This enables simultaneous interference alignment, while guaranteeing the decodability of $\mathbf{a}'$. Notice that (b'_1, b'_2, b'_3) and (c'_1, c'_2, c'_3) are aligned into $b'_1 + b'_2 + b'_3$ and $c'_1 + c'_2 + c'_3$, respectively, while three equations associated with $\mathbf{a}'$ are linearly independent.

As one can see, the complexity of systematic node repair is a little bit lower than that of parity node repair, although both repair schemes are simple. Hence, one can expect that this example is useful for the applications where the complexity of systematic node repair needs to be significantly low.

Example 2 ($\mathbf{U} = \mathbf{I}$): We provide another example of $(6,3,5)$ Exact-Repair MDS codes where $\mathbf{U}$ is orthogonal. We use the same field size of 5 and the same $\mathbf{P}$. Instead we choose a non-orthogonal $\mathbf{V}$ in order to significantly reduce the complexity of parity node repair. Our framework provides a concrete guideline for accomplishing this. Remember that the projection vector solutions are $\mathbf{u}_1$, $\mathbf{u}_2$ and $\mathbf{u}_3$ for exact repair of each parity node, respectively. For low complexity, we can first set $\mathbf{U} = \mathbf{I}$. The condition (4.12) then gives the



(a) Exact repair of systematic node 1

(b) Exact repair of parity node 1

Figure 4.9: Example 1: $\mathbf{V} = \mathbf{I}$. A $(6, 3, 5)$ Exact-Repair MDS code defined over $\mathsf{GF}(5)$. The projection vector solution for systematic node repair is quite simple: $\mathbf{v}_{\alpha i} = \mathbf{v}_1 = (1, 0, 0)^t, \forall i$. This example employs the same encoding matrix and projection vectors for systematic node repair as those in [54]. We download only the first equation from each survivor node; For parity node repair, our new framework provides a simple scheme: setting all of the projection vectors as $2^{-1}\mathbf{u}_1 = (1, 1, 1)^t$. This enables simultaneous interference alignment, while guaranteeing the decodability of $\mathbf{a}$.

following choice:

$$\mathbf{V} = \mathbf{P}^t \kappa^{-1} = \begin{bmatrix} 3 & 3 & 3 \\ 3 & 1 & 4 \\ 3 & 4 & 2 \end{bmatrix},$$

where we use $\kappa = 2$. By (4.6) and (4.13), the primary and dual structures are given by

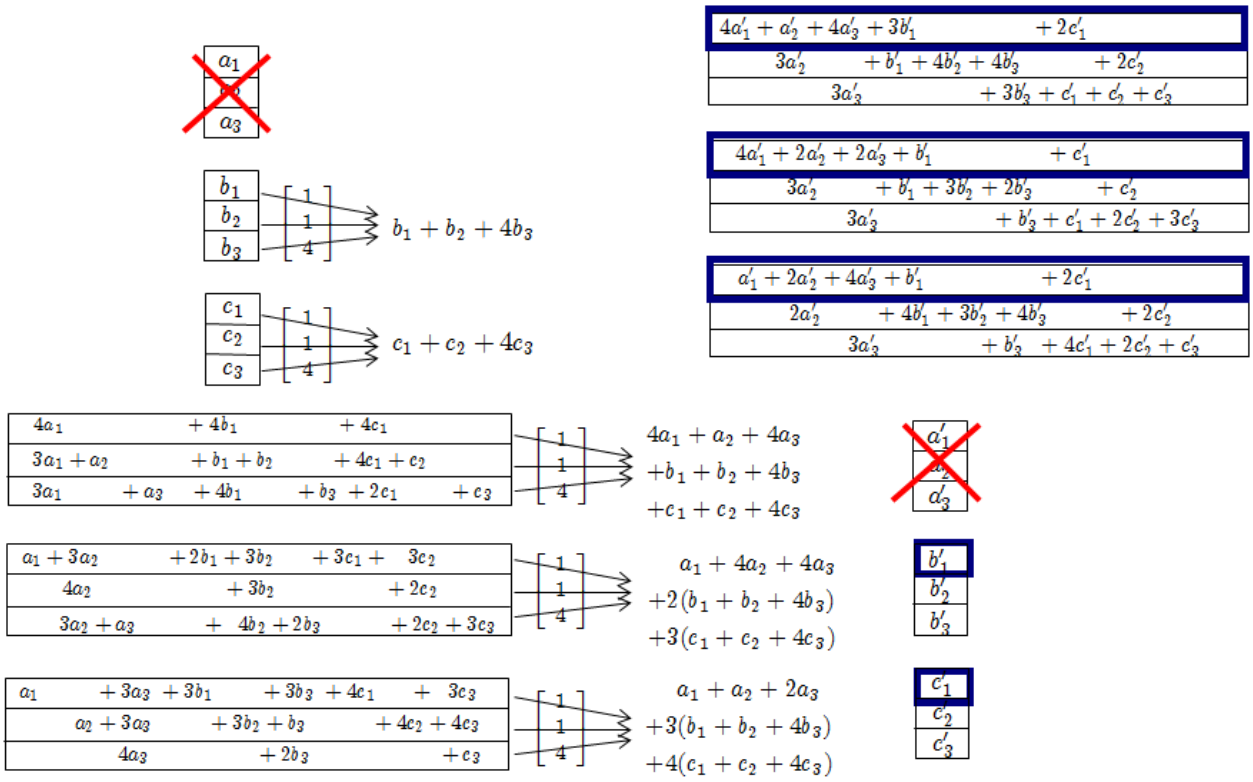
$$\mathbf{G} = \left[\begin{array}{ccc|ccc|ccc} 4 & 3 & 3 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 3 & 4 & 3 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 3 & 3 & 4 \\ \hline 4 & 1 & 4 & 2 & 0 & 0 & 3 & 0 & 0 \\ 0 & 1 & 0 & 3 & 3 & 4 & 0 & 3 & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & 3 & 1 & 2 \\ \hline 4 & 4 & 2 & 3 & 0 & 0 & 4 & 0 & 0 \\ 0 & 1 & 0 & 3 & 2 & 2 & 0 & 4 & 0 \\ 0 & 0 & 1 & 0 & 0 & 3 & 3 & 4 & 1 \end{array}\right]; \mathbf{G}^{-1} = \left[\begin{array}{ccc|ccc|ccc} 4 & 0 & 0 & 4 & 0 & 0 & 1 & 0 & 0 \\ 1 & 3 & 0 & 2 & 3 & 0 & 2 & 2 & 0 \\ 4 & 0 & 3 & 2 & 0 & 3 & 4 & 0 & 3 \\ \hline 3 & 1 & 0 & 1 & 1 & 0 & 1 & 4 & 0 \\ 0 & 4 & 0 & 0 & 3 & 0 & 0 & 3 & 0 \\ 0 & 4 & 3 & 0 & 2 & 1 & 0 & 4 & 1 \\ \hline 2 & 0 & 1 & 1 & 0 & 1 & 2 & 0 & 4 \\ 0 & 2 & 1 & 0 & 1 & 2 & 0 & 2 & 2 \\ 0 & 0 & 1 & 0 & 0 & 3 & 0 & 0 & 1 \end{array}\right], \quad (4.20)$$

where $\mathbf{G}$ is defined as (4.19). Notice that the matrices of (4.20) have exactly the transpose structure of the matrices of (4.18). Hence, this structure of (4.20) is a dual solution to that of (4.18), thereby ensuring the transfer of the lowered complexity property for parity node repair.

Fig. 4.10 shows an example for exact repair of (a) systematic node 1 and (b) parity node 1. In contrast to our previous case, exact repair of parity nodes is now much simpler. In this example, by downloading only the first equation from each survivor node, we can successfully recover $\mathbf{a}'$. On the contrary, systematic node repair is more involved, with all of the projection vectors being set as $2^{-1}\mathbf{v}_1' = (1, 1, 4)^t$. Using this vector, we can achieve simultaneous interference alignment, thereby decoding the desired components of $\mathbf{a}$.

4.4.4 Generalization

Theorem 6 gives insights into generalization to $(2k, k, 2k-1)$ Exact-Repair MDS codes. The key observation is that assuming $\mathcal{M} = k(d-k+1)$, storage cost is $\alpha = \mathcal{M}/k = d-k+1 = k$ and this number is equal to the number of systematic nodes and furthermore matches the number of parity nodes. Notice that the storage size matches the size of encoding submatrices, which determines the number of linearly independent vectors of $\{\mathbf{v}\} := \{\mathbf{v}_1, \cdots\}$. In this case, therefore, we can generate k linearly independent vectors $\{\mathbf{v}\} := \{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$ and corresponding $\{\mathbf{u}\} := \{\mathbf{u}_1, \cdots, \mathbf{u}_k\}$ through the appropriate choice of $\mathbf{P}$. This immediately provides $(2k, k, 2k-1)$ Exact-Repair MDS codes.



(a) Exact repair of systematic node 1 (b) Exact repair of parity node 1

Figure 4.10: Example 2: $\mathbf{U} = \mathbf{I}$. A $(6, 3, 5)$ Exact-Repair MDS code defined over $\mathsf{GF}(5)$. Since we choose $\mathbf{U} = \mathbf{I}$, the projection vector solution for parity node repair, is much simpler. We download only the first equation from each survivor node; systematic node repair is more involved, with all of the projection vectors being set as $2^{-1}\mathbf{v}'_1 = (1, 1, 4)^t$.

Theorem 7 ($(2k, k, 2k-1)$ Exact-Repair MDS Codes)**.** *Let* $\mathbf{P}$ *be a Cauchy matrix:*

$$\mathbf{P} = \begin{bmatrix} p_1^{(1)} & p_1^{(2)} & \cdots & p_1^{(k)} \\ p_2^{(1)} & p_2^{(2)} & \cdots & p_2^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ p_k^{(1)} & p_k^{(2)} & \cdots & p_k^{(k)} \end{bmatrix},$$

where each element $p_j^{(i)} \in \mathsf{GF}(q)$*, where* $q \geq 2k$*. Suppose*

$$\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_k] \text{ is invertible and } \mathbf{U} = \kappa^{-1}\mathbf{V}'\mathbf{P}, \tag{4.21}$$

where $\mathbf{V}' = (\mathbf{V}^t)^{-1}$ *and* κ *is an arbitrary non-zero value* $\in \mathbb{F}_q$ *such that* $1 - \kappa^2 \neq 0$*. Also assume that encoding submatrices are given by*

$$\begin{aligned} \mathbf{G}_1^{(1)} = \mathbf{u}_1\mathbf{v}_1^t + p_1^{(1)}\mathbf{I}, \cdots, \mathbf{G}_k^{(1)} = \mathbf{u}_1\mathbf{v}_k^t + p_k^{(1)}\mathbf{I}, \\ \vdots \qquad\qquad \ddots \qquad\qquad \vdots \\ \mathbf{G}_1^{(k)} = \mathbf{u}_k\mathbf{v}_1^t + p_1^{(k)}\mathbf{I}, \cdots, \mathbf{G}_k^{(k)} = \mathbf{u}_k\mathbf{v}_k^t + p_k^{(k)}\mathbf{I}, \end{aligned} \tag{4.22}$$

where $\mathbf{G}_l^{(i)}$ *indicates an encoding submatrix for parity node* i*, associated with information unit* l*. Then, the repair code achieves the optimal tradeoff of (4.1).*

Proof. See Appendix 4.7.2. □

Remark 19. *Note that the minimum required alphabet size is* $2k$*. As mentioned earlier, this is because we employ a Cauchy matrix for ensuring the invertibility of any submatrices of* $\mathbf{P}$*. One may customize codes to find smaller alphabet-size codes.*

Now what if k is less than the size ($= \alpha = d - k + 1$) of encoding submatrices, i.e., $d \geq 2k-1$? Note that this case automatically implies that $n \geq 2k$, since $n \geq d+1$. The key observation in this case is that the encoding submatrix size is bigger than k, and therefore we have more degrees of freedom (a larger number of linearly independent vectors) than the number of constraints. Hence, exact repair of systematic nodes becomes transparent. This was observed in [54], where it was shown that for $d = n - 1 \geq 2k - 1$, exact repair of systematic nodes only can be guaranteed by carefully manipulating $(2k, k, 2k-1)$ codes through a pruning operation.

We propose a generalized pruning algorithm that ensures exact repair of *all* nodes for $n \geq 2k$ and $d \geq 2k - 1$. The recipe for this has two parts:

1. Constructing a target code from a larger code.

2. Showing that the resulting target code ensures exact repair of all nodes as well as the MDS-code property.

We provide detailed procedures[7] of the first part.

1(a) Using Theorem 7, construct a larger $(2n-2k, n-k, 2n-2k-1)$ code with a finite field size of $q \geq 2n-2k$.

1(b) Remove all the elements associated with the $(n-2k)$ information units (e.g., from the $(k+1)$th to the $(n-k)$th information unit). The number of nodes is then reduced by $(n-2k)$ and so are the number of information units and the number of degrees. Hence, we obtain the $(n, k, n-1)$ code.

1(c) Prune the last $(n-1-d)$ equations in each storage node and also the last $(n-1-d)$ symbols of each information unit, while keeping the number of information units and storage nodes. We can then get the (n, k, d) target code.

Indeed, based on our framework of Section 4.4, it can be shown that the resulting code described above guarantees exact repair of all nodes while retaining the MDS-code property.

Theorem 8 ($\frac{k}{n} \leq \frac{1}{2}, d \geq 2k-1$)**.** *Suppose that all of the survivor systematic nodes participate in the repair. Then, under exact repair constraints of all nodes, the optimal tradeoff of (4.1) can be attained with a deterministic scheme requiring a field size of at most* $2(n-k)$.

Proof. See Appendix 4.7.3. □

Example 1. *Fig. 4.11 illustrates how to construct an* $(n, k, d) = (5, 2, 3)$ *target code based on the above recipe. First construct the* $(2n-2k, n-k, 2n-2k-1) = (6, 3, 5)$ *code, which is larger than the* $(5, 2, 3)$ *target code, but which belongs to the category of* $n = 2k$. *For this code, we employ the example in Fig. 4.9. We now remove all the elements associated with the last* $(n-2k) = 1$ *information unit, which corresponds to* (c_1, c_2, c_3). *Next, prune the last symbol* (a_3, b_3) *of each information unit and associated elements to shrink the storage size into* 2. *We can then obtain the* $(5, 2, 3)$ *target code. Based on the proposed framework in Section 4.4, it can be shown that the resulting code guarantees exact repair of all nodes and the MDS-code property.*

4.5 Framework 2: Code Existence

We propose the second framework, which has a similar structure as that of framework 1, but which encompasses all admissible values of (n, k, d). This framework draws its inspiration from the symbol-extended interference alignment technique by Cadambe and Jafar [14], meant for the wireless interference channel. Here the symbol extension is analogous to the idea of *vector linear codes* in the network coding field.

[7]While Steps (1a) and (1b) come from the pruning technique in [54], Step (1c) is a significant distinction from that of [54].

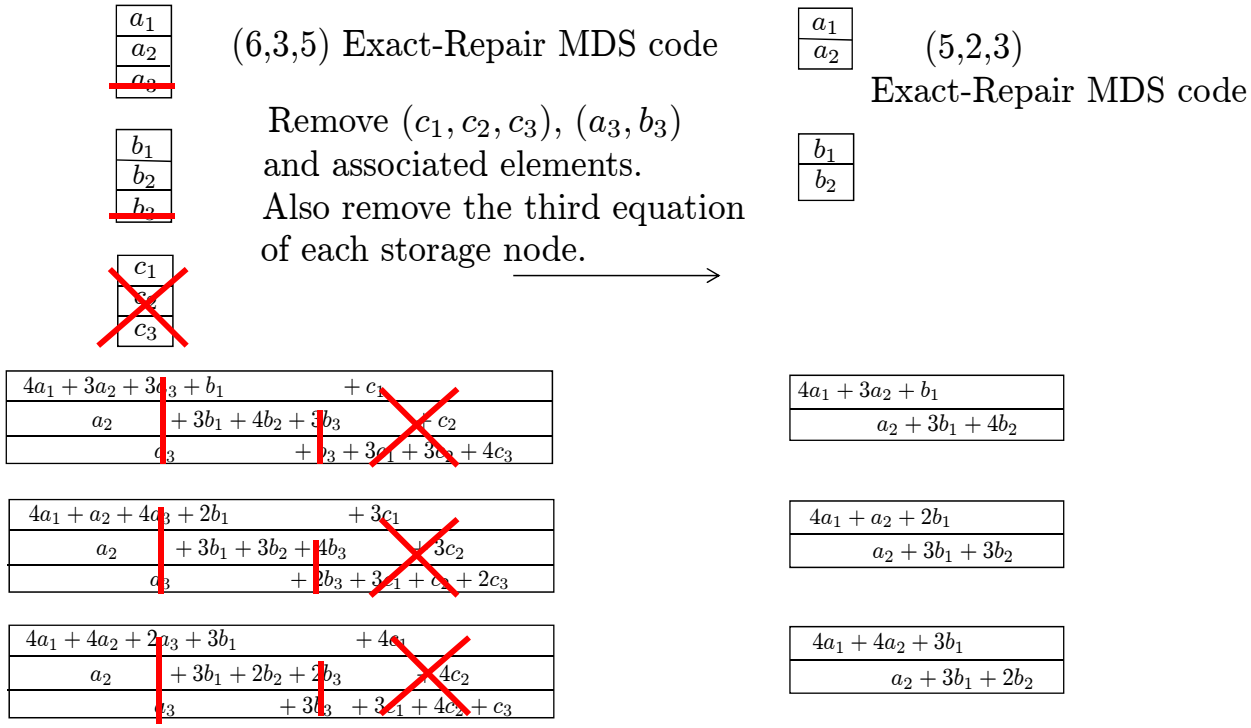


Figure 4.11: Illustration of the construction of a $(5, 2, 3)$ Exact-Repair MDS code from a $(6, 3, 5)$ Exact-Repair MDS code defined over $\mathsf{GF}(5)$. For a larger code, we adopt the $(6, 3, 5)$ code in Fig. 4.9. First, we remove all the elements associated with the last $(n - 2k) = 1$ information unit ("**c**"). Next, we prune symbols (a_3, b_3) and associated elements. Also we remove the last equation of each storage node. Finally we obtain the $(n, k, d) = (5, 2, 3)$ target code.

Similar to the previous framework, it consists of four components: (1) developing a code structure for exact repair of systematic nodes; (2) drawing a dual structure between the systematic and parity node repair; (3) guaranteeing the MDS-code property; (4) providing a probabilistic guarantee of the existence of the code for a large enough alphabet size. In particular, the diagonal structure of single-antenna wireless channels (exploited in [14]) forms the basis of the structure of encoding submatrices of our codes.

The framework covers all admissible values of (n, k, d). This contrasts the scalar-linear code based framework which covers a subset of all admissible values, but which provides a deterministic code construction with small alphabet size and guaranteed zero error. In contrast, here we target only the existence of exact-repair codes without specifying constructions. This allows for a simpler characterization of the solution space for the entire range of admissible repair code parameters. In order to convey the concepts in a clear and concise manner, we first focus on the simplest example which does not belong to the previous framework: $(6, 3, 4)$ Exact-Repair MDS codes. This example is a representative of the general case of $k < n$ and $k \leq d \leq n-1$, with the generalization following in a straightforward way from this example. This will be discussed in Section 4.5.4.

4.5.1 Systematic Node Repair

In order to address the challenge of this simultaneous interference alignment, we invoke the idea of symbol extension introduced in [14], which is equivalent to the concept of vector linear codes in the storage repair problem. Fig. 4.12 illustrates the idea of vector linear codes through storage node 1 in the $(6, 3, 4)$ code example. While scalar linear codes do not allow symbol splitting, vector linear codes permit the splitting of symbols into arbitrarily small subsymbols. In this example, each node stores $\alpha = 2$ symbols, each of which has unit capacity. In vector linear codes, this unit-capacity symbol is allowed to be split into subsymbols with arbitrarily small fractional capacity. In this example, we split each symbol into B number of subsymbols, so each subsymbol has $\frac{1}{B}$ capacity.

This idea of vector linear codes is the key to interference alignment for the storage repair problem. Fig. 4.13 illustrates exact repair of systematic node 1 for $(6, 3, 4)$ Exact-Repair MDS codes. Using vector linear codes, we split each symbol into $B = m^N$ number of subsymbols, where m is an arbitrarily large positive integer and the exponent N is carefully chosen depending on code parameters. Specifically,

$$N = (k-1)(d-k+1). \tag{4.23}$$

This choice of N and the form of $B = m^N$ are closely related to the scheme to be described in the sequel. In this example, $N = 4$. The maximum file size (based on the cutset bound (4.1)) is $\mathcal{M} = 6$ units, inducing a storage cost $\alpha = 2$ units. Since each subsymbol has $\frac{1}{m^4}$ capacity, each storage node contains $\alpha m^4 (= 2m^4)$ number of subsymbols, e.g., $\mathbf{a}^t = (a_1, \cdots, a_{2m^4})$, where a_i indicates a subsymbol. Note that the size of encoding submatrices $(\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i)$

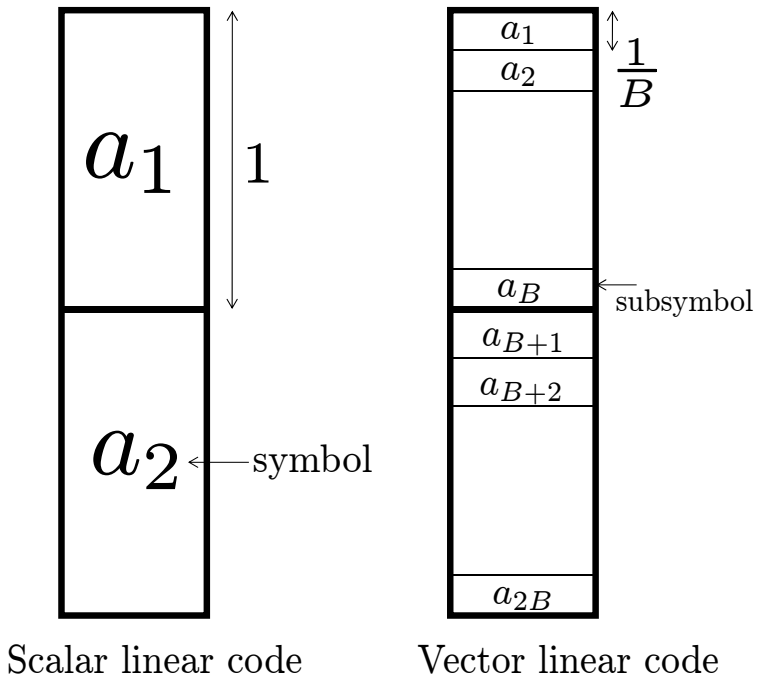


Figure 4.12: Illustration of the idea of vector linear codes through storage node 1 in the $(6, 3, 4)$ code example. In scalar linear codes, symbols are not allowed to be split. On the other hand, vector linear codes allow to split symbols into arbitrarily small subsymbols. In this example, node 1 stores $\alpha = 2$ symbols, each of which has unit capacity. In vector linear codes, this unit-capacity symbol can be split into subsymbols with arbitrarily small fractional capacity. For example, we can split each symbol into B number of subsymbols, so each subsymbol has $\frac{1}{B}$ capacity.

is $2m^4$-by-$2m^4$. We consider *diagonal* encoding submatrices. As pointed out in [14], the diagonal matrix structure ensures a *commutative* property which provides the key to the interference alignment scheme (to be described shortly):

$$\mathbf{A}_i = \begin{bmatrix} \alpha_{i,1} & 0 & \cdots & 0 \\ 0 & \alpha_{i,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \alpha_{i,2m^4} \end{bmatrix} (\textit{commutative} \text{ property holds}). \tag{4.24}$$

A failed node 1 is exactly repaired through the following steps. Assume that survivor nodes $(2, 3, 4, 5)$ participate in exact repair of node 1, i.e., $k - 1 = 2$ systematic nodes and $d - k + 1 = 2$ parity nodes. One can alternatively use 1 systematic node and 3 parity nodes for repair instead. This does not fundamentally alter the analysis, and will be covered in Section 20. For the time being, assume the above configuration for the connection: $(k - 1)$ systematic nodes and $(d - k + 1)$ parity nodes. The parity survivor nodes project their data using the following *projection matrix*:

$$\mathbf{V} := [\mathbf{v}_1, \cdots, \mathbf{v}_{m^4}] \in \mathbb{F}_q^{2m^4 \times m^4}, \tag{4.25}$$

where $\mathbf{v}_i \in \mathcal{V}$. The set $\mathcal{V}$ is defined as:

$$\mathcal{V} := \left\{ (\mathbf{B}_1^{e_1} \mathbf{B}_2^{e_2} \mathbf{C}_1^{e_3} \mathbf{C}_2^{e_4})\, \mathbf{w} : e_1, e_2, e_3, e_4 \in \{0, \cdots, m-1\} \right\}, \tag{4.26}$$

where $\mathbf{w} = [1, \cdots, 1]^t$. Note that $|\mathcal{V}| \leq m^4$. The vector $\mathbf{v}_i$ maps to a different sequence of (e_1, e_2, e_3, e_4). For example, we can map:

$$\begin{aligned} &\mathbf{v}_1 = \mathbf{w},\ \mathbf{v}_2 = \mathbf{C}_2\mathbf{w},\ \mathbf{v}_3 = \mathbf{C}_2^2\mathbf{w},\ \cdots, \\ &\mathbf{v}_{m^4-2} = \mathbf{B}_1^{m-1}\mathbf{B}_2^{m-1}\mathbf{C}_1^{m-1}\mathbf{C}_2^{m-3}\mathbf{w}, \\ &\mathbf{v}_{m^4-1} = \mathbf{B}_1^{m-1}\mathbf{B}_2^{m-1}\mathbf{C}_1^{m-1}\mathbf{C}_2^{m-2}\mathbf{w}, \\ &\mathbf{v}_{m^4} = \mathbf{B}_1^{m-1}\mathbf{B}_2^{m-1}\mathbf{C}_1^{m-1}\mathbf{C}_2^{m-1}\mathbf{w}. \end{aligned} \tag{4.27}$$

Consider the equations downloaded from parity nodes 1 and 2 (nodes 4 and 5):

$$\begin{aligned} &\text{From parity node 1: } \mathbf{a}^t(\mathbf{A}_1\mathbf{V}) + \mathbf{b}^t(\mathbf{B}_1\mathbf{V}) + \mathbf{c}^t(\mathbf{C}_1\mathbf{V}); \\ &\text{From parity node 2: } \mathbf{a}^t(\mathbf{A}_2\mathbf{V}) + \mathbf{b}^t(\mathbf{B}_2\mathbf{V}) + \mathbf{c}^t(\mathbf{C}_2\mathbf{V}). \end{aligned} \tag{4.28}$$

Note that $\mathbf{B}_1\mathbf{V}$ contains the following column vectors:

$$\begin{aligned} &\mathbf{B}_1\mathbf{v}_1 = \mathbf{B}_1\mathbf{w},\ \mathbf{B}_1\mathbf{v}_2 = \mathbf{B}_1\mathbf{C}_2\mathbf{w},\ \mathbf{B}_1\mathbf{v}_3 = \mathbf{B}_1\mathbf{C}_2^2\mathbf{w},\ \cdots, \\ &\mathbf{B}_1\mathbf{v}_{m^4-2} = \mathbf{B}_1^{m}\mathbf{B}_2^{m-1}\mathbf{C}_1^{m-1}\mathbf{C}_2^{m-3}\mathbf{w}, \\ &\mathbf{B}_1\mathbf{v}_{m^4-1} = \mathbf{B}_1^{m}\mathbf{B}_2^{m-1}\mathbf{C}_1^{m-1}\mathbf{C}_2^{m-2}\mathbf{w}, \\ &\mathbf{B}_1\mathbf{v}_{m^4} = \mathbf{B}_1^{m}\mathbf{B}_2^{m-1}\mathbf{C}_1^{m-1}\mathbf{C}_2^{m-1}\mathbf{w}. \end{aligned}$$

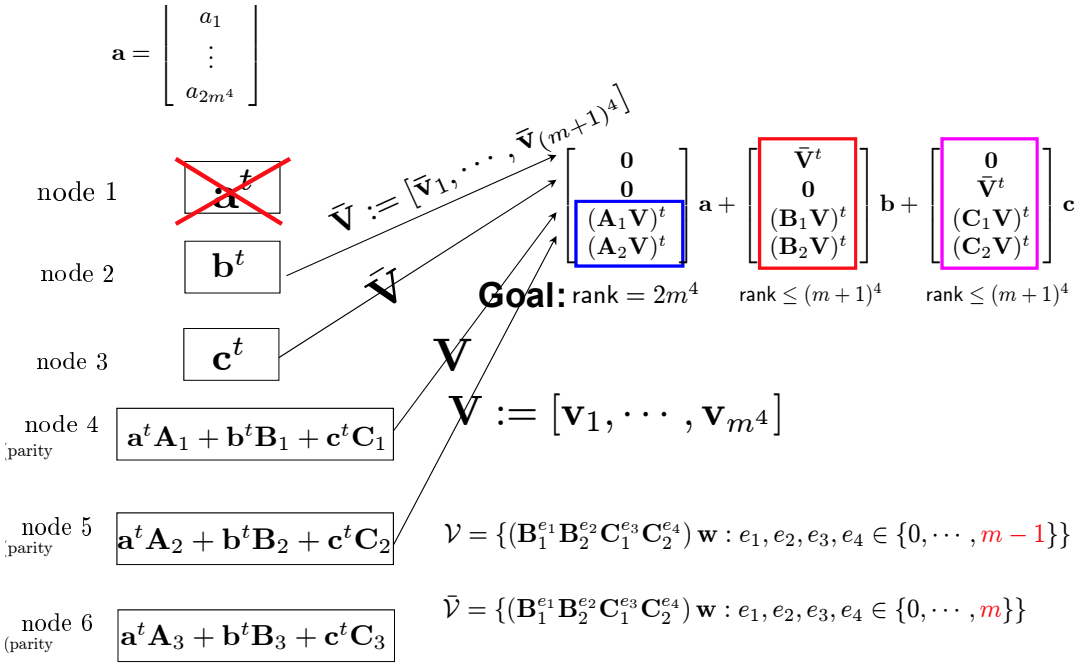


Figure 4.13: Illustration of exact repair of systematic node 1 for a $(6, 3, 4)$ Exact-Repair MDS code. We split each symbol into $B = m^N$ number of subsymbols, where m is an arbitrarily large positive integer and the exponent N is equal to 4 and is carefully chosen depending on code parameters, specifically $N = (k-1)(d-k+1) = 4$. This corresponds to the total number of encoding submatrices involved in the connection except for those associated with desired signals. Note that each subsymbol has $\frac{1}{m^4}$ capacity. The maximum file size (based on the optimal tradeoff of (4.1)) is $\mathcal{M} = 6$ units, inducing a storage cost $\alpha = 2$ units. Hence, each storage contains $2m^4$ number of subsymbols and the size of encoding submatrices is $2m^4$-by-$2m^4$. We consider diagonal encoding submatrices. A failed node is exactly repaired by having systematic and parity survivor nodes project their data onto linear subspaces spanned by column vectors of $\bar{\mathbf{V}} := [\bar{\mathbf{v}}_1, \cdots, \bar{\mathbf{v}}_{(m+1)^4}]$ and $\mathbf{V} := [\mathbf{v}_1, \cdots, \mathbf{v}_{m^4}]$, respectively. Here $\bar{\mathbf{v}}_i \in \bar{\mathcal{V}}$ and $\mathbf{v}_i \in \mathcal{V}$. Notice that $\mathbf{B}_1\mathbf{v}_i, \mathbf{B}_2\mathbf{v}_i, \mathbf{C}_1\mathbf{v}_i, \mathbf{C}_2\mathbf{v}_i \in \bar{\mathcal{V}}, \forall i = 1, \cdots, m^4$. Hence, the matrix associated with interference $\mathbf{b}$ has rank of at most $(m+1)^4$ instead of $2m^4$. Similarly the matrix associated with interference $\mathbf{c}$ has rank of at most $(m+1)^4$. This enables simultaneous interference alignment as $m \to \infty$. On the other hand, $\mathsf{rank}[\mathbf{A}_1\mathbf{V}, \mathbf{A}_2\mathbf{V}] = 2m^4$ with probability 1, providing a probabilistic guarantee of decodability of desired signals. Finally, notice that the total repair bandwidth $\gamma = 2\frac{(m+1)^4}{m^4} + 2 \cdot 1$ approaches the cutset lower bound of 4 units as m goes to infinity. Therefore, we can ensure exact repair of systematic node 1 with minimum repair bandwidth matching the cutset lower bound.

An important observation is that any column vector $\mathbf{B}_1\mathbf{v}_i$ is an element of $\bar{\mathcal{V}}$ defined as:

$$\bar{\mathcal{V}} := \{(\mathbf{B}_1^{e_1}\mathbf{B}_2^{e_2}\mathbf{C}_1^{e_3}\mathbf{C}_2^{e_4})\,\mathbf{w} : e_1, e_2, e_3, e_4 \in \{0, \cdots, m\}\}\,. \tag{4.29}$$

Similarly any column vector in $\mathbf{B}_2\mathbf{V}$, $\mathbf{C}_1\mathbf{V}$ or $\mathbf{C}_2\mathbf{V}$ is an element of $\bar{\mathcal{V}}$. This implies that $[\mathbf{B}_1\mathbf{V}, \mathbf{B}_2\mathbf{V}] \in \mathbb{F}_q^{2m^4 \times 2m^4}$ is a rank-deficient matrix, i.e., $\mathsf{rank}[\mathbf{B}_1\mathbf{V}, \mathbf{B}_2\mathbf{V}] \leq (m+1)^4$. Similarly $\mathsf{rank}[\mathbf{C}_1\mathbf{V}, \mathbf{C}_2\mathbf{V}] \leq (m+1)^4$. This allows for simultaneous interference alignment although the same projection matrix $\mathbf{V}$ is used for $\mathbf{b}$ and $\mathbf{c}$. This observation motivates the systematic survivor nodes to project their data using the following projection matrix:

$$\bar{\mathbf{V}} := [\bar{\mathbf{v}}_1, \cdots, \bar{\mathbf{v}}_{(m+1)^4}] \in \mathbb{F}_q^{2m^4 \times (m+1)^4}, \tag{4.30}$$

where $\bar{\mathbf{v}}_i \in \bar{\mathcal{V}}$ and is mapped to a difference sequence of (e_1, e_2, e_3, e_4) as in (4.27). We can then guarantee that:

$$\begin{aligned} \mathsf{colspan}[\mathbf{B}_1\mathbf{V}, \mathbf{B}_2\mathbf{V}] &\subset \mathsf{colspan}[\bar{\mathbf{V}}] \\ \mathsf{colspan}[\mathbf{C}_1\mathbf{V}, \mathbf{C}_2\mathbf{V}] &\subset \mathsf{colspan}[\bar{\mathbf{V}}]. \end{aligned} \tag{4.31}$$

Hence, using $\mathbf{b}^t\bar{\mathbf{V}}$ and $\mathbf{c}^t\bar{\mathbf{V}}$ (downloaded from systematic survivor nodes), we can completely remove any interference $(\mathbf{b}^t(\mathbf{B}_1\mathbf{V}), \mathbf{b}^t(\mathbf{B}_2\mathbf{V}), \mathbf{c}^t(\mathbf{C}_1\mathbf{V}), \mathbf{c}^t(\mathbf{C}_2\mathbf{V}))$ from (4.28), thereby obtaining $\mathbf{a}^t[\mathbf{A}_1\mathbf{V}, \mathbf{A}_2\mathbf{V}]$. To successfully recover $\mathbf{a}$, we need:

$$\mathsf{rank}[\mathbf{A}_1\mathbf{V}, \mathbf{A}_2\mathbf{V}] = 2m^4. \tag{4.32}$$

In other words, $[\mathbf{A}_1\mathbf{V}, \mathbf{A}_2\mathbf{V}]$ must have full rank. The proof of equation (4.32) is the existence proof stemming from the Schwartz-Zippel Lemma [46]. Specifically, we show that there exist diagonal encoding submatrices $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ so that this is satisfied. The argument is as follows.

Consider equation (4.32). In the matrix on the left hand side, notice that the design of $\mathbf{V}$ in (4.26) does not depend on $\mathbf{A}_1$ and $\mathbf{A}_2$. Therefore, it can be noted that each entry of the matrix is a different monomial in the diagonal entries of the encoding submatrices $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$. Based on this observation, it can be shown that the determinant of the matrix in (4.32) is a non-zero polynomial in the diagonal entries of $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i, i = 1, 2, 3$. Let us denote this polynomial by $g(\cdot)$. Note that for (4.32) to be satisfied, we need $g(\cdot)$ to evaluate to a non-zero value in the field. Therefore, it suffices to show that there exists a realization of diagonal entries for the coding submatrices so that the polynomial $g(\cdot)$ evaluate to a non-zero value in the field. To do so, we invoke the Schwartz-Zippel Lemma. Over a sufficiently large field, the lemma guarantees, via a probabilistic argument, the existence of diagonal matrices $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ so that this polynomial evaluates to some non-zero value.

We now validate that total repair bandwidth approaches $\gamma = 4$ units as the number of subsymbols goes to infinity:

$$\begin{aligned} \gamma &= (k-1)\frac{(m+1)^4}{m^4} + (d-k+1)\cdot\frac{m^4}{m^4} \\ &= 2\frac{(m+1)^4}{m^4} + 2\cdot 1 \\ &\longrightarrow 4 \text{ units}, \;\; m \to \infty. \end{aligned} \tag{4.33}$$

The first equality is because each subsymbol has capacity of $\frac{1}{m^4}$ and we use projection matrix $\bar{\mathbf{V}} \in \mathbb{F}_q^{2m^4 \times (m+1)^4}$ and $\mathbf{V} \in \mathbb{F}_q^{2m^4 \times m^4}$ when connecting to systematic nodes and parity nodes respectively. Note that as m goes to infinity, the total repair bandwidth approaches minimum repair bandwidth matching the cutset lower bound (4.1).

4.5.2 Parity Node Repair

The code $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ constructed here can also be used to create an optimal repair strategy for a failed parity node in the same manner. The key idea is the following. In an MDS code, any k nodes are information equivalent to the original information in a system and therefore can be interpreted as k systematic nodes. The data stored in the remaining $(n-k)$ nodes are functions of these k nodes and can therefore be interpreted as parity nodes. Hence, through a remapping of the nodes and an appropriate transformation, a parity node of a code can be interpreted as a systematic node of a virtual alternate code - a parity node failure can therefore be interpreted as a systematic node failure under a virtual alternate code. Specifically, for linear MDS codes, by using a change of basis, a parity node in the original code can be virtually interpreted as a systematic node of a virtual alternate code. As long as the alternate code shares properties similar to the original code (diagonal encoding submatrices etc.), the ideas of systematic node repair can be applied to parity node repair as well. Let us crystallize this idea in the context of an example. Suppose that a parity node, say node 6, fails. We can now remap the nodes so that this failed node is systematic node $\mathbf{c}'^t$. Therefore, in this alternate virtual code, we have three systematic nodes $\mathbf{a}', \mathbf{b}', \mathbf{c}'$:

$$\begin{aligned} &\text{Node 1: } \mathbf{a}'^t = \mathbf{a}^t \\ &\text{Node 2: } \mathbf{b}'^t = \mathbf{b}^t \\ &\text{Node 3: } \mathbf{c}'^t = \mathbf{a}^t\mathbf{A}_3 + \mathbf{b}^t\mathbf{B}_3 + \mathbf{c}^t\mathbf{C}_3. \end{aligned} \tag{4.34}$$

With this remaping, $\mathbf{c}^t$ is now a parity node. The three parity nodes can be expressed as

$$\begin{aligned} &\text{Node 4: } \mathbf{a}'^t\{\mathbf{A}_1 + \mathbf{A}_3(\mathbf{C}_3)^{-1}\mathbf{C}_1\} + \mathbf{b}'^t\{\mathbf{B}_1 + \mathbf{B}_3(\mathbf{C}_3)^{-1}\mathbf{C}_1\} + \mathbf{c}'^t(\mathbf{C}_3)^{-1}\mathbf{C}_1 \\ &\text{Node 5: } \mathbf{a}'^t\{\mathbf{A}_2 + \mathbf{A}_3(\mathbf{C}_3)^{-1}\mathbf{C}_2\} + \mathbf{b}'^t\{\mathbf{B}_2 + \mathbf{B}_3(\mathbf{C}_3)^{-1}\mathbf{C}_2\} + \mathbf{c}'^t(\mathbf{C}_3)^{-1}\mathbf{C}_2. \\ &\text{Node 6: } \mathbf{c}^t = \mathbf{a}'^t\mathbf{A}_3(\mathbf{C}_3)^{-1} + \mathbf{b}'^t\mathbf{B}_3(\mathbf{C}_3)^{-1} + \mathbf{c}'^t(\mathbf{C}_3)^{-1}. \end{aligned} \tag{4.35}$$

Let us denote the ith parity node (i.e., node $i+k = i+3$) as $\mathbf{a}'^t\mathbf{A}'_i + \mathbf{b}'^t\mathbf{B}'_i + \mathbf{c}'^t\mathbf{C}'_i$ so that for example $\mathbf{A}'_1 = \mathbf{A}_1 + \mathbf{A}_3(\mathbf{C}_3)^{-1}\mathbf{C}_1$ and so on. From the above expressions, all the encoding submatrices $\mathbf{A}'_i, \mathbf{B}'_i, \mathbf{C}'_i$ are diagonal. This is because the sum, product and inverse of two diagonal matrices are diagonal. The diagonal property ensures that the encoding submatrices commute even in this virtual code. This means that by picking the repair vectors in a manner analogous to (4.26) and (4.29) aligns interference so that an equation analogous to (4.31) is satisfied. Using an argument similar to the previous section, it can be

shown that the desired signal can also be completely recovered as well. The detailed proof is omitted here to avoid tedious notation.

Remark 20 (Participation of Arbitrary d Nodes for Exact Repair)**.** *We have considered a somewhat restrictive connection configuration for exact repair: namely connecting to surviving* $(k-1)$ *systematic nodes and to other* $(d-k+1)$ *parity nodes. We now consider more general connection configurations. For example, consider the case when node 1 fails. Suppose we connect to nodes* $(2,4,5,6)$ *for exact repair of node 1: 1 systematic node and 3 parity nodes. We use the idea similar to the idea of parity node repair. We remap one parity node to make it look like a systematic node. We then virtually connect to 2 systematic and to 2 parity nodes. Specifically we can remap node 6 with* $\mathbf{c}'^t$ *and perform conversions similar to (4.34) and (4.35). Then applying the same procedures as before, we can guarantee the exact repair of* $\mathbf{a}$*.*

4.5.3 MDS-Code Property

The MDS property means that the code must be able to tolerate the failure of any three storage nodes in the system. Equivalently, any set of three nodes in the system, when interpreted as equations in $\mathbf{a}, \mathbf{b}, \mathbf{c}$ must have full rank of $\mathcal{M} = 6m^4$ and hence the matrix representing these equations, must have a non-zero determinant. Note that there are $\binom{6}{3}$ possible sets of three nodes in the storage system. The MDS property is therefore equivalent to showing that $\binom{6}{3} = 20$ determinants are all non-zero. Note that each determinant is a polynomial in the entries of the encoding submatrices. In the next section, we will show in the more general context of arbitrary n and k that even with diagonal coding submatrices chosen here, all these polynomials are non-zero. To summarize, we show that the MDS property corresponds to 20 non-zero polynomials in the entries of the diagonal elements of $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ each evaluating to a non-zero value. We will denote these polynomials by $f_1, f_2, \ldots, f_{20}$. Again using the Schwartz-Zippel Lemma, one can show that there exists a realization of diagonal entries for the coding submatrices so that the product of the polynomials $f_1(\cdot) \cdots f_2(\cdot)$ evaluates to a non-zero value in the field.

As mentioned, for a sufficiently large field size, a random construction for encoding submatrices suffices to guarantee exact repair of all nodes and MDS-code property with probability 1. Hence, we obtain the following theorem.

Theorem 9 ($(6,3,4)$ Exact-Repair MDS Codes)**.** *There exist vector linear Exact-Repair MDS codes that achieve the minimum repair bandwidth corresponding to the cutset bound (4.1), allowing for any failed node to be exactly repaired with access to any arbitrary* $d = 4$ *survivor nodes, provided storage symbols can be split into a sufficiently large number of subsymbols, and the field size can be made sufficiently large.*

4.5.4 Generalization

The interference alignment technique described in the previous sections can be generalized to all admissible values of (n, k, d), i.e., $k < n$ and $k \leq d \leq n-1$.

Theorem 10 ((n, k, d) Exact-Repair MDS Codes)**.** *There exist vector linear Exact-Repair MDS codes that achieve the minimum repair bandwidth corresponding to the cutset bound (4.1), allowing for any failed node to be exactly repaired with access to any arbitrary d survivor nodes, where $k \leq d \leq n-1$, provided storage symbols can be split into a sufficiently large number of subsymbols, and the field size can be made sufficiently large.*

Proof. In the interests of conceptual simplicity, and to parallel the analysis of the $(6, 4, 3)$ example described earlier, we provide only a sketch of the proof for the general case. This can be formalized to be precise at the cost of much heavier notational clutter, which we consciously avoid.

Systematic Node Repair: Let $\mathbf{G}_l^{(i)}$ indicate an encoding submatrix for parity node i, associated with information packet l, where $1 \leq i \leq n-k$ and $1 \leq l \leq k$. Let $\mathbf{w}_l$ be lth information packet vector. Without loss of generality, consider exact repair of systematic node 1. Using vector linear codes, we split each symbol into $B = m^N$ number of subsymbols, where m is an arbitrarily large positive integer and the exponent N is given by

$$N = (k-1)(d-k+1). \tag{4.36}$$

The maximum file size (based on the cutset bound (4.1)) is $\mathcal{M} = k(d-k+1)$ units, inducing a storage cost $\alpha = d-k+1$ units. Since each subsymbol has $\frac{1}{m^N}$ capacity, each storage contains αm^N number of subsymbols. Note that the size of encoding submatrices is αm^N-by-αm^4.

A failed node 1 is exactly repaired through the following steps. Suppose without loss of generality that we connect to $(k-1)$ systematic nodes and to first $(d-k+1)$ parity nodes[8]. The parity survivor nodes project their data using the following projection matrix:

$$\mathbf{V} := [\mathbf{v}_1, \cdots, \mathbf{v}_{m^N}] \in \mathbb{F}_q^{\alpha m^N \times m^N}, \tag{4.37}$$

where $\mathbf{v}_i \in \mathcal{V}$. The set $\mathcal{V}$ is defined as:

$$\mathcal{V} := \left\{ \prod_{i=1,\cdots,d-k+1, l=2,\cdots,k} \left[\mathbf{G}_l^{(i)}\right]^{e_{i,l}} \mathbf{w} : e_{i,l} \in \{0, \cdots, m-1\} \right\}, \tag{4.38}$$

where $\mathbf{w} = [1, \cdots, 1]^t$. Note that $|\mathcal{V}| \leq m^N$.

[8]As mentioned earlier, we can convert the other connection configurations into this particular configuration with the remapping technique.

Let us consider the equations downloaded from parity nodes:

$$\begin{aligned}
&\mathbf{w}_1^t(\mathbf{G}_1^{(1)}\mathbf{V}) + \mathbf{w}_2^t(\mathbf{G}_2^{(1)}\mathbf{V}) + \cdots + \mathbf{w}_k^t(\mathbf{G}_k^{(1)}\mathbf{V});\\
&\qquad\vdots\\
&\mathbf{w}_1^t(\mathbf{G}_1^{(d-k+1)}\mathbf{V}) + \mathbf{w}_2^t(\mathbf{G}_2^{(d-k+1)}\mathbf{V}) + \cdots + \mathbf{w}_k^t(\mathbf{G}_k^{(d-k+1)}\mathbf{V}).
\end{aligned} \tag{4.39}$$

Note that by (4.38), for $l \neq 1$, any column vector in $[\mathbf{G}_l^{(1)}\mathbf{V}, \cdots, \mathbf{G}_l^{(d-k+1)}\mathbf{V}]$ is an element of $\bar{\mathcal{V}}$ defined as:

$$\bar{\mathcal{V}} := \left\{ \prod_{i=1,\cdots,d-k+1, l=2,\cdots,k} \left[\mathbf{G}_l^{(i)}\right]^{e_{i,l}} \mathbf{w} : e_{i,l} \in \{0, \cdots, m\} \right\}, \tag{4.40}$$

This implies that for $l \neq 1$, $\mathsf{rank}[\mathbf{G}_l^{(1)}\mathbf{V}, \cdots, \mathbf{G}_l^{(d-k+1)}\mathbf{V}] \leq (m+1)^N$. This allows for simultaneous interference alignment. The systematic survivor nodes project their data using the following *projection matrix*:

$$\bar{\mathbf{V}} := [\bar{\mathbf{v}}_1, \cdots, \bar{\mathbf{v}}_{(m+1)^N}] \in \mathbb{F}_q^{\alpha m^N \times (m+1)^N}, \tag{4.41}$$

where $\bar{\mathbf{v}}_i \in \bar{\mathcal{V}}$. We can then guarantee that for $l \neq 1$:

$$\mathsf{span}[\mathbf{G}_l^{(1)}\mathbf{V}, \cdots, \mathbf{G}_l^{(d-k+1)}\mathbf{V}] \subset \mathsf{span}[\bar{\mathbf{V}}]. \tag{4.42}$$

Hence, using $\mathbf{w}_i^t\bar{\mathbf{V}}$ $(i \neq 1)$ obtained from systematic survivor nodes, we can clean out any interference of $\mathbf{w}_i^t(\mathbf{G}_i\mathbf{V})$ $(i \neq 1)$ from (4.39). Now let us consider the decodability of desired signals. To successfully recover $\mathbf{w}_1$, we need:

$$\mathsf{rank}[\mathbf{G}_1^{(1)}\mathbf{V}, \cdots, \mathbf{G}_1^{(d-k+1)}\mathbf{V}] = (d-k+1)m^N = \alpha m^N. \tag{4.43}$$

Using the same argument based on Schwartz-Zippel lemma, we can ensure (4.43) with probability 1 for a sufficiently large field size.

Finally we validate that total repair bandwidth is:

$$\begin{aligned}
\gamma &= (k-1)\frac{(m+1)^N}{m^N} + (d-k+1)\cdot\frac{m^N}{m^N}\\
&\longrightarrow d \text{ units.}
\end{aligned} \tag{4.44}$$

Note that as m goes to infinity, the total repair bandwidth approaches minimum repair bandwidth matching the cutset lower bound (4.1).

Parity Node Repair: As discussed in the previous sections, we can draw a dual structure by remapping parity nodes with primed new notations. The key observation is that newly mapped encoding submatrices are still diagonal matrices. Hence, we can apply the same technique used in systematic node repair.

MDS-Code Property: We check the invertibility of a composite matrix when a Data Collector connects to i systematic nodes and to $(k-i)$ parity nodes for $i = 0, \cdots, k$. As mentioned earlier, for a sufficiently large field size, the composite matrix has non-zero determinant with probability 1. □

4.6 Summary

Unlike wireless communication problems, our storage repair problems have more flexibility in designing encoding matrices which correspond to wireless channel coefficients (provided by nature) in communication problems. Exploiting this fact, we developed the common-eigenvector-based interference alignment technique to provide a constructive code framework for optimal exact repair codes for the case of $\frac{k}{n} \leq \frac{1}{2}$ and $d \geq 2k-1$. This framework provides insights into a dual relationship between the systematic and parity node repair, as well as opens up a larger constructive design space of solutions.

Leveraging the strong connection between the wireless interference channel problem and the storage repair problem, we could make use of Cadambe-Jafar's interference alignment scheme to develop optimal exact-repair MDS codes for all admissible values of (n, k, d). This code requires an infinite file size to achieve the minimum repair bandwidth. Exploring whether or not a *finite* file size is sufficient to achieve the minimum repair bandwidth is an interesting direction of future work.

4.7 Appendices

4.7.1 Proof of Lemma 3

It suffices to show that

$$\begin{bmatrix} \mathbf{A}'_1 & \mathbf{A}'_2 & \mathbf{A}'_3 \\ \mathbf{B}'_1 & \mathbf{B}'_2 & \mathbf{B}'_3 \\ \mathbf{C}'_1 & \mathbf{C}'_2 & \mathbf{C}'_3 \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \mathbf{A}_3 \\ \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 \\ \mathbf{C}_1 & \mathbf{C}_2 & \mathbf{C}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}.$$

Using (4.6) and (4.13), we compute:

$$\begin{aligned}
&(1-\kappa^2)(\mathbf{A}'_1\mathbf{A}_1 + \mathbf{A}'_2\mathbf{B}_1 + \mathbf{A}'_3\mathbf{C}_1) \\
&= \left(\mathbf{v}'_1\mathbf{u}''^t_1 - \kappa^2\alpha'_1\mathbf{I}\right)(\mathbf{u}_1\mathbf{v}_1^t + \alpha_1\mathbf{I}) + \left(\mathbf{v}'_2\mathbf{u}''^t_1 - \kappa^2\beta'_1\mathbf{I}\right)(\mathbf{u}_1\mathbf{v}_2^t + \beta_1\mathbf{I}) + \left(\mathbf{v}'_3\mathbf{u}''^t_1 - \kappa^2\gamma'_1\mathbf{I}\right)(\mathbf{u}_1\mathbf{v}_3^t + \gamma_1\mathbf{I}) \\
&\overset{(a)}{=} (\mathbf{v}'_1\mathbf{v}_1^t + \mathbf{v}'_2\mathbf{v}_2^t + \mathbf{v}'_3\mathbf{v}_3^t) + (\alpha_1\mathbf{v}'_1 + \beta_1\mathbf{v}'_2 + \gamma_1\mathbf{v}'_3)\mathbf{u}''^t_1 - \kappa^2\mathbf{u}_1(\alpha'_1\mathbf{v}_1 + \beta'_1\mathbf{v}_2 + \gamma'_1\mathbf{v}_3)^t - \kappa^2\mathbf{I} \\
&\overset{(b)}{=} (\mathbf{v}'_1\mathbf{v}_1^t + \mathbf{v}'_2\mathbf{v}_2^t + \mathbf{v}'_3\mathbf{v}_3^t) + \kappa\mathbf{u}_1\mathbf{u}''^t_1 - \kappa^2\mathbf{u}_1(\alpha'_1\mathbf{v}_1 + \beta'_1\mathbf{v}_2 + \gamma'_1\mathbf{v}_3)^t - \kappa^2\mathbf{I} \\
&\overset{(c)}{=} (\mathbf{v}'_1\mathbf{v}_1^t + \mathbf{v}'_2\mathbf{v}_2^t + \mathbf{v}'_3\mathbf{v}_3^t) - \kappa^2\mathbf{I} \\
&\overset{(d)}{=} (1-\kappa^2)\mathbf{I}
\end{aligned}$$

where (a) follows from $\alpha_1\alpha'_1 + \beta_1\beta'_1 + \gamma_1\gamma'_1 = 1$ due to (4.11); (b) follows from (4.12); (c) follows from $\mathbf{u}'_1 = \kappa(\alpha'_1\mathbf{v}_1 + \beta'_1\mathbf{v}_2 + \gamma'_1\mathbf{v}_3)$ (See Claim 6); and (d) follows from the fact that $\mathbf{v}'_1\mathbf{v}_1^t + \mathbf{v}'_2\mathbf{v}_2^t + \mathbf{v}'_3\mathbf{v}_3^t = \mathbf{I}$, since $(\mathbf{v}'_1, \mathbf{v}'_2, \mathbf{v}'_3)$ are dual basis vectors.

Similarly, one can check that $\mathbf{B}_1'\mathbf{A}_2 + \mathbf{B}_2'\mathbf{B}_2 + \mathbf{B}_3'\mathbf{C}_2 = \mathbf{I}$ and $\mathbf{C}_1'\mathbf{A}_3 + \mathbf{C}_2'\mathbf{B}_3 + \mathbf{C}_3'\mathbf{C}_3 = \mathbf{I}$. Now let us compute one of the cross terms:

$$
\begin{aligned}
&(1-\kappa^2)(\mathbf{A}_1'\mathbf{A}_2 + \mathbf{A}_2'\mathbf{B}_2 + \mathbf{A}_3'\mathbf{C}_2) \\
&= \left(\mathbf{v}_1'\mathbf{u}_1'^t - \kappa^2\alpha_1'\mathbf{I}\right)(\mathbf{u}_2\mathbf{v}_1^t + \alpha_2\mathbf{I}) + \left(\mathbf{v}_2'\mathbf{u}_1'^t - \kappa^2\beta_1'\mathbf{I}\right)(\mathbf{u}_2\mathbf{v}_2^t + \beta_2\mathbf{I}) + \left(\mathbf{v}_3'\mathbf{u}_1'^t - \kappa^2\gamma_1'\mathbf{I}\right)(\mathbf{u}_2\mathbf{v}_3^t + \gamma_2\mathbf{I}) \\
&\overset{(a)}{=} (\alpha_2\mathbf{v}_1' + \beta_2\mathbf{v}_2' + \gamma_2\mathbf{v}_3')\mathbf{u}_1'^t - \kappa^2\mathbf{u}_2(\alpha_1'\mathbf{v}_1 + \beta_1'\mathbf{v}_2 + \gamma_1'\mathbf{v}_3)^t \\
&\overset{(b)}{=} 0
\end{aligned}
$$

where (a) follows from $\mathbf{u}_i'^t\mathbf{u}_j = \delta(i-j)$ and $< (\alpha_1', \beta_1', \gamma_1'), (\alpha_2, \beta_2, \gamma_2) >= 0$; (b) follows from (4.12) and Claim 6. Similarly, we can check that the other cross terms are zero matrices. This completes the proof.

Claim 6. *For all i, $\mathbf{u}_i' = \kappa(\alpha_i'\mathbf{v}_1 + \beta_i'\mathbf{v}_2 + \gamma_i'\mathbf{v}_3)$.*

Proof. By (4.12), we can rewrite

$$
[\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3] = \frac{1}{\kappa}[\mathbf{v}_1', \mathbf{v}_2', \mathbf{v}_3'] \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}.
$$

Using the fact that $(\mathbf{u}_1', \mathbf{u}_2', \mathbf{u}_3')$ are dual basis vectors, we get

$$
\begin{bmatrix} \mathbf{u}_1'^t \\ \mathbf{u}_2'^t \\ \mathbf{u}_3'^t \end{bmatrix} = \kappa \begin{bmatrix} \alpha_1' & \beta_1' & \gamma_1' \\ \alpha_2' & \beta_2' & \gamma_2' \\ \alpha_3' & \beta_3' & \gamma_3' \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^t \\ \mathbf{v}_2^t \\ \mathbf{v}_3^t \end{bmatrix}.
$$

This completes the proof. □

4.7.2 Proof of Theorem 7

For generalization, we are forced to use some heavy notation but only for this section and the related appendices. Let $\mathbf{w}_j \in \mathbb{F}_q^k$ be a message vector for information unit j. Let $\mathbf{G}_j^{(i)} \in \mathbb{F}_q^{k\times k}$ be an encoding submatrix for parity node i, associated with the jth information unit.

Exact Repair of Systematic Nodes: By symmetry, we consider only systematic node 1. We have each survivor node project their data with projection vector $\mathbf{v}_1'$, which is the first column vector of $\mathbf{V}' = (\mathbf{V}^t)^{-1}$. We then get:

$$
\begin{aligned}
&\text{From systematic node } j\text{: } \mathbf{w}_j^t\mathbf{v}_1', \\
&\text{From parity node } i\text{: } \mathbf{w}_1^t(\mathbf{u}_i + p_1^{(i)}\mathbf{v}_1') + \underbrace{\sum_{j=2}^{k} p_j^{(i)}(\mathbf{w}_j^t\mathbf{v}_1')}_{interference},
\end{aligned}
$$

where $2 \leq j \leq k$ and $1 \leq i \leq k$. Note that we can achieve simultaneous interference alignment for non-intended signals. Since $\mathbf{u}_i$'s are linearly independent, we can decode desired signals $\mathbf{w}_1$, thus ensuring exact repair.

Exact Repair of Parity Nodes: The idea is the same as that of Theorem 6. First we remap parity nodes with new variables:

$$\begin{bmatrix} \mathbf{w}'_1 \\ \mathbf{w}'_2 \\ \vdots \\ \mathbf{w}'_k \end{bmatrix} := \begin{bmatrix} \mathbf{G}_1^{(1)t} & \mathbf{G}_2^{(1)t} & \cdots & \mathbf{G}_k^{(1)t} \\ \mathbf{G}_1^{(2)t} & \mathbf{G}_2^{(2)t} & \cdots & \mathbf{G}_k^{(2)t} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}_1^{(k)t} & \mathbf{G}_2^{(k)t} & \cdots & \mathbf{G}_k^{(k)t} \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_k \end{bmatrix}.$$

Define the newly remapped encoding submatrices as:

$$\begin{bmatrix} \mathbf{G}_1^{\prime(1)} & \mathbf{G}_1^{\prime(2)} & \cdots & \mathbf{G}_1^{\prime(k)} \\ \mathbf{G}_2^{\prime(1)} & \mathbf{G}_2^{\prime(2)} & \cdots & \mathbf{G}_2^{\prime(k)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}_k^{\prime(1)} & \mathbf{G}_k^{\prime(2)} & \cdots & \mathbf{G}_k^{\prime(k)} \end{bmatrix} := \begin{bmatrix} \mathbf{G}_1^{(1)} & \mathbf{G}_1^{(2)} & \cdots & \mathbf{G}_1^{(k)} \\ \mathbf{G}_2^{(1)} & \mathbf{G}_2^{(2)} & \cdots & \mathbf{G}_2^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}_k^{(1)} & \mathbf{G}_k^{(2)} & \cdots & \mathbf{G}_k^{(k)} \end{bmatrix}^{-1}. \tag{4.45}$$

We can now apply the generalization of Lemma 3 to obtain the dual structure:

$$\begin{cases} \mathbf{G}_1^{\prime(1)} = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_1\mathbf{u}_1^{\prime t} - \kappa^2 p_1^{\prime(1)}\mathbf{I}\right), \\ \vdots \\ \mathbf{G}_k^{\prime(1)} = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_1\mathbf{u}_k^{\prime t} - \kappa^2 p_1^{\prime(k)}\mathbf{I}\right), \end{cases} \cdots \begin{cases} \mathbf{G}_1^{\prime(k)} = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_k\mathbf{u}_1^{\prime t} - \kappa^2 p_k^{\prime(1)}\mathbf{I}\right), \\ \vdots \\ \mathbf{G}_k^{\prime(k)} = \frac{1}{1-\kappa^2}\left(\mathbf{v}'_k\mathbf{u}_k^{\prime t} - \kappa^2 p_k^{\prime(k)}\mathbf{I}\right), \end{cases}$$

where the dual basis vectors are defined as:

$$\begin{bmatrix} p_1^{\prime(1)} & p_2^{\prime(1)} & \cdots & p_k^{\prime(1)} \\ p_1^{\prime(2)} & p_2^{\prime(2)} & \cdots & p_k^{\prime(2)} \\ \vdots & \vdots & \ddots & \vdots \\ p_1^{\prime(k)} & p_2^{\prime(k)} & \cdots & p_k^{\prime(k)} \end{bmatrix} := \begin{bmatrix} p_1^{(1)} & p_1^{(2)} & \cdots & p_1^{(k)} \\ p_2^{(1)} & p_2^{(2)} & \cdots & p_2^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ p_k^{(1)} & p_k^{(2)} & \cdots & p_k^{(k)} \end{bmatrix}^{-1}.$$

By symmetry, we consider only parity node 1. Choosing the projection vector $\mathbf{u}_1$, we get:

From systematic node j: $\dfrac{\mathbf{w}_1^{\prime t}}{1-\kappa^2}(\mathbf{u}'_j - \kappa^2 p_1^{\prime(j)}\mathbf{v}'_1) - \dfrac{\kappa^2}{1-\kappa^2}\displaystyle\sum_{i=2}^{k} p_i^{\prime(j)}(\mathbf{w}_i^{\prime t}\mathbf{u}_1)$,

From parity node i : $\mathbf{w}_i^{\prime t}\mathbf{u}_1$,

where $1 \leq j \leq k$ and $2 \leq i \leq k$. Note that we can achieve simultaneous interference alignment for non-intended signals. Since $\mathbf{u}'_i$'s are linearly independent, we can decode desired signals $\mathbf{w}'_1$, thus ensuring exact repair of parity node 1.

The MDS-Code Property: We check the invertibility of a composite encoding submatrix when a Data Collector connects to i systematic nodes and $(k-i)$ parity nodes for $i = 0, \cdots, k$. The main idea is to use a Gaussian elimination method as we did in Section 4.4.3. The verification is tedious and therefore details are omitted.

Minimum Required Finite-Field Size: Note that the dimension of a Cauchy matrix $\mathbf{P}$ is k-by-k. Therefore, the minimum finite-field size required to generate the Cauchy matrix is $2k$, i.e., $q \geq 2k$.

4.7.3 Proof of Theorem 8

According to the proposed pruning algorithm, we start with an larger $(2n-2k, n-k, 2n-2k-1)$ code which has encoding submatrices as follows:

$$\begin{cases} \mathbf{G}_1^{(1)} = \mathbf{u}_1\mathbf{v}_1^t + p_1^{(1)}\mathbf{I}, \\ \vdots \\ \mathbf{G}_{n-k}^{(1)} = \mathbf{u}_1\mathbf{v}_{n-k}^t + p_{n-k}^{(1)}\mathbf{I}, \end{cases} \cdots \begin{cases} \mathbf{G}_1^{(n-k)} = \mathbf{u}_{n-k}\mathbf{v}_1^t + p_1^{(n-k)}\mathbf{I}, \\ \vdots \\ \mathbf{G}_{n-k}^{(n-k)} = \mathbf{u}_{n-k}\mathbf{v}_{n-k}^t + p_{n-k}^{(n-k)}\mathbf{I}, \end{cases} \tag{4.46}$$

where $\mathbf{G}_j^{(i)} \in \mathbb{F}_q^{(n-k)\times(n-k)}$ indicates an encoding submatrix for parity node i, associated with the jth information unit. We use an invertible matrix for $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_{n-k}]$ and set

$$\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_{n-k}] = \kappa^{-1}\mathbf{V}'\mathbf{P}, \tag{4.47}$$

where $\mathbf{V}' = (\mathbf{V}^t)^{-1}$ and $\kappa \in \mathbb{F}_q$ is an arbitrary non-zero value such that $1-\kappa^2 \neq 0$. We use a Cauchy matrix $\mathbf{P}$ and let $p_j^{(i)}$ be the (j,i) element of $\mathbf{P}$. Notice that we have $(n-k)$ information units $\mathbf{w}_j \in \mathbb{F}_q^{n-k}$, $1 \leq j \leq n-k$.

Next we remove the last $(n-2k)$ information units and associated elements to obtain the $(n, k, n-1)$ code. This code has information units $(\mathbf{w}_1, \cdots, \mathbf{w}_k)$ and encoding submatrices $\mathbf{G}_j^{(i)}$ for $1 \leq j \leq k$ and $1 \leq i \leq n-k$. Lastly, we prune the last $(n-1-d)$ equations in each storage node and also the last $(n-1-d)$ symbols of each information unit. We then obtain the (n, k, d) target code which has encoding submatrices:

$$\begin{cases} \bar{\mathbf{G}}_1^{(1)} = \bar{\mathbf{u}}_1\bar{\mathbf{v}}_1^t + p_1^{(1)}\mathbf{I}, \\ \vdots \\ \bar{\mathbf{G}}_k^{(1)} = \bar{\mathbf{u}}_1\bar{\mathbf{v}}_k^t + p_k^{(1)}\mathbf{I}, \end{cases} \cdots \begin{cases} \bar{\mathbf{G}}_1^{(n-k)} = \bar{\mathbf{u}}_{n-k}\bar{\mathbf{v}}_1^t + p_1^{(n-k)}\mathbf{I}, \\ \vdots \\ \bar{\mathbf{G}}_k^{(n-k)} = \bar{\mathbf{u}}_{n-k}\bar{\mathbf{v}}_k^t + p_k^{(n-k)}\mathbf{I}, \end{cases} \tag{4.48}$$

where $\bar{\mathbf{u}}_i, \bar{\mathbf{v}}_j \in \mathbf{F}_q^{d-k+1}$ indicate the top $(d-k+1)$ symbols of $\mathbf{u}_i, \mathbf{v}_j \in \mathbf{F}_q^{n-k}$, respectively. Here the size of an identity matrix $\mathbf{I}$ is $(d-k+1)$. For simplicity, we use the same notation for a different dimension of an identity matrix. It can be easily differentiated from the context.

Let us now prove that the resulting code ensures exact repair of all nodes and MDS-code property. We will provide the detailed proof for a simple case of $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_{n-k}] = \mathbf{I}$.

Exact Repair of Systematic Nodes: By symmetry, we consider only systematic node 1. We connect to $(k-1)$ systematic nodes and $(d-k+1)$ parity nodes. Without loss of generality, we consider parity nodes from 1 to $d-l$. As for a projection vector, we use $\mathbf{e}_1 = [1, 0, \cdots, 0]^t$. We then get:

$$\text{From systematic node } j\text{: } \bar{\mathbf{w}}_j^t \mathbf{e}_1,$$

$$\text{From parity node } i\text{: } \bar{\mathbf{w}}_1^t(\bar{\mathbf{u}}_i + p_1^{(i)}\mathbf{e}_1) + \sum_{j=2}^{k} p_j^{(i)}(\bar{\mathbf{w}}_j^t \mathbf{e}_1)$$

where $2 \le j \le k$ and $1 \le i \le d-k+1$. Note that we can achieve simultaneous interference alignment for non-intended signals. The interference term can be canceled with side information obtained from systematic nodes. After cancelation, we rewrite $(d-k+1)$ equations obtained from parity nodes:

$$\bar{\mathbf{w}}_1^t \left[\begin{array}{c} \bar{\mathbf{u}}_1 + p_1^{(1)}\mathbf{e}_1, \bar{\mathbf{u}}_2 + p_1^{(2)}\mathbf{e}_1, \cdots, \bar{\mathbf{u}}_{d-k+1} + p_1^{(d-k+1)}\mathbf{e}_1 \end{array} \right]$$

By (4.47), $\bar{\mathbf{u}}_i = \kappa^{-1}(p_1^{(i)}, \cdots, p_{d-k+1}^{(i)})^t$, $\forall i = 1, \cdots, n-k$. Using the fact that any submatrix of $\mathbf{P}$ is invertible, it can be shown that the matrix in the right-hand-side is invertible. This guarantees the decodability of the desired message vector $\bar{\mathbf{w}}_1$.

Exact Repair of Parity Nodes: By symmetry, it suffices to consider parity node 1. We connect to k systematic nodes and $(d-k)$ parity nodes. Without loss of generality, we consider parity nodes from 2 to $d-k+1$. As for a projection vector, we use $\bar{\mathbf{u}}_1 = \kappa^{-1}(p_1^{(1)}, \cdots, p_{d-k+1}^{(1)})^t$. We then get:

$$\text{From systematic node } j\text{: } \bar{\mathbf{w}}_j^t \bar{\mathbf{u}}_1,$$

$$\text{From parity node } i\text{: } \sum_{j=1}^{k} \bar{\mathbf{w}}_j^t \left(\bar{\mathbf{u}}_i \mathbf{e}_j^t + p_j^{(i)} \mathbf{I} \right) \bar{\mathbf{u}}_1 = \frac{1}{\kappa} \left(\sum_{j=1}^{k} p_j^{(1)} \bar{\mathbf{w}}_j^t \right) \bar{\mathbf{u}}_i + \sum_{j=1}^{k} p_j^{(i)} \left(\bar{\mathbf{w}}_j^t \bar{\mathbf{u}}_1 \right),$$

where $1 \le j \le k$ and $2 \le i \le d-k+1$. Here the equality follows from the fact that $\mathbf{e}_j^t \bar{\mathbf{u}}_1 = \kappa^{-1} p_j^{(1)}$. Note that the second term in the parity node equation can be canceled with side information $(\bar{\mathbf{w}}_1^t \bar{\mathbf{u}}_1, \cdots, \bar{\mathbf{w}}_k^t \bar{\mathbf{u}}_1)$ obtained from systematic nodes. After cancelation, we rewrite $(d-k)$ equations obtained from parity nodes:

$$\left[\sum_{j=1}^{k} p_j^{(1)} \bar{\mathbf{w}}_j^t \right] \left[\begin{array}{c} \frac{1}{\kappa}\bar{\mathbf{u}}_2, \frac{1}{\kappa}\bar{\mathbf{u}}_3, \cdots, \frac{1}{\kappa}\bar{\mathbf{u}}_{d-k+1} \end{array} \right].$$

Since we know $\bar{\mathbf{w}}_j^t \bar{\mathbf{u}}_1$ (side-information obtained from systematic nodes), we can construct $\frac{1}{\kappa} \sum_{j=1}^{k} p_j^{(1)} \bar{\mathbf{w}}_j^t \bar{\mathbf{u}}_1$. Adding this value to the above, we get:

$$\left[\sum_{j=1}^{k} p_j^{(1)} \bar{\mathbf{w}}_j^t \right] \left[\begin{array}{cccc} \frac{1}{\kappa}\bar{\mathbf{u}}_1, & \frac{1}{\kappa}\bar{\mathbf{u}}_2, & \frac{1}{\kappa}\bar{\mathbf{u}}_3, & \cdots, \quad \frac{1}{\kappa}\bar{\mathbf{u}}_{d-k+1} \end{array} \right].$$

Using the fact that any submatrix of $\mathbf{P}$ is invertible, we can show that the right-hand-side matrix is invertible. This enables to decode the left-hand-side vector, thus obtaining:

$$\bar{\mathbf{w}}_1^t \bar{\mathbf{u}}_1,\ \cdots,\ \bar{\mathbf{w}}_k^t \bar{\mathbf{u}}_1,\ \sum_{j=1}^{k} p_j^{(1)} \bar{\mathbf{w}}_j^t, \tag{4.49}$$

Using this information, we can now regenerate

$$\sum_{j=1}^{k} \left(\bar{\mathbf{w}}_j^t \bar{\mathbf{u}}_1\right) \mathbf{e}_j^t + \sum_{j=1}^{k} p_j^{(1)} \bar{\mathbf{w}}_j^t = \sum_{j=1}^{k} \bar{\mathbf{w}}_j^t \left(\bar{\mathbf{u}}_1 \mathbf{e}_j^t + p_j^{(1)} \mathbf{I}\right) = \sum_{j=1}^{k} \bar{\mathbf{w}}_j^t \bar{\mathbf{G}}_j^{(1)}.$$

This matches the content of parity node 1, thus ensuring exact repair of the parity node.

The MDS-Code Property: We check the invertibility of a composite encoding submatrix when a Data Collector connects to i systematic nodes and $(k-i)$ parity nodes for $i = 0, \cdots, k$. The main idea is to use a Gaussian elimination method as we did in Section 4.4.3. The verification is tedious and therefore details are omitted.

Minimum Required Finite-Field Size: Note that the dimension of a Cauchy matrix $\mathbf{P}$ is $(n-k)$-by-$(n-k)$. Therefore, the minimum finite-field size required to generate the Cauchy matrix is $2(n-k)$, i.e., $q \geq 2(n-k)$.

Chapter 5

Conclusion

In this dissertation, we have made progress on addressing the following two questions: (1) What is the fundamental role of feedback in interference networks? (2) How should multiple links code their information to mitigate the interference they cause to each other?

Role of Feedback: The first part of the dissertation addressed the first question. Specifically, we showed that feedback has a significant role to mitigate interference, thus improving the non-feedback capacity. To show this, we have established the feedback capacity region to within 2 bits/s/Hz/user and the symmetric capacity to within 1 bit/s/Hz/user universally for the two-user Gaussian IC with feedback. We develop both new achievable scheme and outer bound to provide an approximate characterization of the capacity region. As a side-generalization, we have characterized the exact feedback capacity region of El Gamal-Costa deterministic IC. We also develop two interpretations as to how feedback can provide significant gain. One interpretation is that feedback maximizes resource utilization by filling up all the resource holes under-utilized in the non-feedback case. The other interpretation is that feedback can exploit received signals as side information to increase capacity. Interestingly, the latter interpretation leads us to make a connection to other problems.

Our feedback result considers a case where feedback is given for free from each receiver to its corresponding transmitter. One natural question that arises is: What if feedback cost is taken into account, i.e., can 1 bit of feedback provide more than 1 bit of a capacity increase? Surprisingly, our recent preliminary result gives us a positive answer. Specifically, we consider the two-way interference channel where feedback can be provided through the backward interference channel. We show that 1 bit of feedback can provide a capacity increase of an arbitrarily large number of bits. The next step is to extend this to more general network settings.

We believe this research topic may provide insights into developing a new communication network infrastructure for smart grids. This belief is motivated by the observation that smart grids are facing a very frequent information exchange between smart meters at home, Independent System Operators and power generators, thus making information flows highly interactive. Existing infrastructures such as Wi-Fi networks and cellular networks may not

be efficient to deal with highly interactive information flows.

Interference Alignment: Exploiting a recent breakthrough, the concept of interference alignment, we develop an interference alignment technique for cellular networks. Our IA technique shows significant performance especially when the power of the dominant interferer is much greater than the power of the remaining aggregate interference. We also proposed subspace IA scheme in order to mitigate the interference from multiple dominant interferers, which achieves almost interference-free dof even for more-than-two cell cases. Of practical importance is the fact that our downlink IA scheme can be implemented with small changes to an existing cellular system supporting multi-user MIMO, as it requires only a localized *within-a-cell* feedback mechanism. This technique can be extended to asymmetric antenna configurations and scenarios with more than one dominant interferer.

As mentioned earlier, our IA scheme can provide huge gain especially when interference from the dominant interferer (the nearby base-station) is much stronger than residual interference from many other base-stations. This naturally leads us to believe that our scheme has great potential to heterogeneous networks that merge a multitude of wireless networks, such as femto-cells, pico-cells, relays and Wi-Fi networks, into macro cellular networks. Notice, for example, that in macro-pico cellular networks, a user connected to a pico base-station may see significant interference from the nearby macro base-station.

In Chapter 4, we found the universality of the IA principle developed in the context of wireless networks. Deep understanding on the IA principle has enabled us to find the interdisciplinary nature of this principle. Connecting wireless networks to distributed storage networks, we developed a new class of MDS codes that significantly reduces the repair cost over the conventional MDS codes and also achieves information-theoretic optimal bound on the repair cost for all admissible code parameters. Specifically, under scalar-linear codes, we have constructed Exact-Repair MDS codes that achieve the cutset lower bound on repair band for the case of $\frac{k}{n} \leq \frac{1}{2}$; and $d \geq 2k-1$. Furthermore, we have shown the existence of vector-linear Exact-Repair MDS codes that are optimal in repair bandwidth, for all admissible values of (n, k, d).

This result comes from applying the principle developed in communication networks into the other field of networks, storage networks. Now what if we go back to communication networks with this result? Our recent observation finds an intimate connection between the storage repair problem and a class of multiple unicast flow problems (an open problem for decades in network coding fields). We now intend to exploit the principles learned from storage networks to address the class of multiple unicast problems.

We believe that ultimately this research will give insights into cracking general class of multiple unicast problems. Also this research - exploiting the interference alignment principle in the context of *wireline* networks - will have much greater impact in practice, as compared to *wireless* networks where the principle originates. In *wireless* networks, many of the IA techniques are faced with significant challenges in implementation: they require global channel state information and high signal-to-noise ratio. In *wireline* networks, on the other hand, these challenges disappear. Since the wireless channels - that may need to be

fed back frequently due to variation over time - are mapped to network coding coefficients in wireline networks, much less frequent feedback is required in the wireline networks. Also there is no concept of signal-to-noise ratio in wireline networks.

Bibliography

[1] *3GPP TR 25.814 V7.1.0, "Physical Layer Aspects for Evolved Universal Terrestrial Radio Access"*. October 2007.

[2] *IEEE, 802.16m-08/0004r2, "IEEE802.16m Evaluation Methodology Document (EMD)"*. July 2008.

[3] *3GPP TR 36.814, "Further advancements for E-UTRA physical layer aspects"*. March 2010.

[4] *Qualcomm Incorporated, "LTE Advanced: Heterogeneous Networks"*. February 2010.

[5] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, July 2000.

[6] S. M. Alamouti. A simple transmit diversity technique for wireless communication. *IEEE Journal on Selected Areas in Communications*, 16:1451–1458, October 1998.

[7] Salman Avestimehr, Suhas Diggavi, and David Tse. A deterministic approach to wireless relay networks. *Proceedings of Allerton Conference on Communication, Control, and computing.*, September 2007.

[8] Ziv Bar-Yossef, Yitzhak Birk, T.S.Jayram, and Tomer Kol. Index coding with side information. *Foundations of Computer Science (FOCS)*, pages 197–206, October 2006.

[9] Dennis S. Bernstein. *Matrix mathematics: Theory, facts, and formulas with application to linear systems theory*. Princeton University Press, 2005.

[10] Guy Bresler and David Tse. The two-user Gaussian interference channel: a deterministic view. *European Transactions on Telecommunications*, June 2008.

[11] A. A. Bubrulle. Work notes on elementary matrices. *Tech. Rep. HPL-93-69, Hewlett-Packard Laboratory*, 1993.

[12] S. Butman. A general formulation of linear feedback communication systems with solutions. *IEEE Transactions on Information Theory*, 15:392–400, May 1969.

[13] Viveck R. Cadambe, Cheng Huang, Syed A. Jafar, and Jin Li. Optimal repair of MDS codes in distributed storage via subspace interference alignment. *Proceedings of the IEEE International Symposium on Information Theory, Saint Petersburg, Russia (arXiv:1106.1250)*, July 2011.

[14] Viveck R. Cadambe and Syed A. Jafar. Interference alignment and the degree of freedom for the k user interference channel. *IEEE Transactions on Information Theory*, 54(8):3425–3441, August 2008.

[15] Viveck R. Cadambe, Syed A. Jafar, and Hamed Maleki. Distributed data storage with minimum storage regenerating codes - exact and functional repair are asymptotically equally efficient. *arXiv:1004.4229*, April 2010.

[16] T. M. Cover and Abbas A. El-Gamal. Capacity theorems for the relay channel. *IEEE Transactions on Information Theory*, 25:572–584, September 1979.

[17] T. M. Cover and C. S. K. Leung. An achievable rate region for the multiple-access channel with feedback. *IEEE Transactions on Information Theory*, 27:292–298, May 1981.

[18] T. M. Cover and Sandeep Pombra. Gaussian feedback capacity. *IEEE Transactions on Information Theory*, 35:37–43, January 1989.

[19] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. New York Wiley, 2th edition, July 2006.

[20] Daniel Cullina, A. G. Dimakis, and Tracey Ho. Searching for minimum storage regenerating codes. *Allerton Conference on Control, Computing and Communication*, September 2009.

[21] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. Wainwright, and Kannan Ramchandran. Network coding for distributed storage systems. *Proceedings of the IEEE INFOCOM, Anchorage, Alaska*, May 2007.

[22] A. G. Dimakis, Kannan Ramchandran, Yunnan Wu, and Changho Suh. A survey on network codes for distributed storage. *Proceedings of the IEEE*, 99(3):476–489, March 2011.

[23] Randall Dougherty, Christopher Freiling, and Kenneth Zeger. Insufficiency of linear network codes. *IEEE Transactions on Information Theory*, 51(8):2745–2759, August 2005.

[24] A. El-Gamal and M. H. Costa. The capacity region of a class of deterministic interference channels. *IEEE Transactions on Information Theory*, 28(2):343–346, March 1982.

[25] Raul Etkin, David Tse, and Hua Wang. Gaussian interference channel capacity to within one bit. *IEEE Transactions on Information Theory*, 54:5534–5562, December 2008.

[26] N. T. Gaarder and J. K. Wolf. The capacity region of a multiple-access discrete memoryless channel can increase with feedback. *IEEE Transactions on Information Theory*, January 1975.

[27] Michael Gastpar and Gerhard Kramer. On noisy feedback for interference channels. *In Proc. Asilomar Conference on Signals, Systems, and Computers*, October 2006.

[28] Leonidas Georgiadis and Leandros Tassiulas. Broadcast erasure channel with feedback - capacity and algorithms. *2009 Workshop on Network Coding, Theory and Applications (NetCod)*, June 2009.

[29] David Gesbert, Saad Ghazanfar Kiani, Anders Gjendemsjø, and Geir Egil Øien. Adaptation, coordination, and distributed resource allocation in interference-limited wireless networks. *Proceedings of the IEEE*, 95(12):2393–2409, December 2007.

[30] David Gesbert, Marios Kountouris, Robert W. Heath, Chan-Byoung Chae, and T. Salzer. From single user to multiuser communications: Shifting the MIMO paradigm. *IEEE Signal Processing Magazine*, 24(5), October 2007.

[31] Krishna Gomadam, Viveck R. Cadambe, and Syed A. Jafar. Approaching the capacity of wireless networks through distrubted interference alignment. *Proceedings of the IEEE GLOBECOM, New Orleans, USA*, December 2008.

[32] T. S. Han and K. Kobayashi. A new achievable rate region for the interference channel. *IEEE Transactions on Information Theory*, 27:49–60, January 1981.

[33] Andries P. Hekstra and Frans M. J. Willems. Dependence balance bounds for single-output two-way channels. *IEEE Transactions on Information Theory*, 35:44–53, January 1989.

[34] Tracey Ho, Ralf Koetter, Muriel Médard, Michelle Effros, J. Shi, and D. Karger. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52(10):4413–4430, October 2006.

[35] A. S. Householder. *The Theory of Matrices in Numerical Analysis*. Dover, Toronto, Cananda, 1974.

[36] Jinhua Jiang, Yan Xin, and Hari Krishna Garg. Discrete memoryless interference channels with feedback. *CISS 41st Annual Conference*, pages 581–584, March 2007.

[37] Nihar Jindal, Sriram Vishwanath, and Andrea Goldsmith. On the duality of Gaussian multiple-access and broadcast channels. *IEEE Transactions on Information Theory*, 50(5):768–783, May 2004.

[38] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft. XORs in the air: Practical wireless network coding. *ACM SIGCOMM Computer Communication Review*, 36:243–254, October 2006.

[39] Young-Han Kim. Feedback capacity of the first-order moving average Gaussian channel. *IEEE Transactions on Information Theory*, 52(7):3063–3079, July 2006.

[40] R. Koetter and M. Médard. An algebraic approach to network coding. *IEEE/ACM Transactions on Networking*, 11(5):782–795, October 2003.

[41] Gerhard Kramer. Feedback strategies for white Gaussian interference networks. *IEEE Transactions on Information Theory*, 48:1423–1438, June 2002.

[42] Gerhard Kramer. Correction to "Feedback strategies for white Gaussian interference networks", and a capacity theorem for Gaussian interference channels with feedback. *IEEE Transactions on Information Theory*, 50, June 2004.

[43] J. Nicholas Laneman and Gregory W. Wornell. Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks. *IEEE Transactions on Information Theory*, 49:2415–2425, October 2003.

[44] M. A. Maddah-Ali, S. A. Motahari, and Amir K. Khandani. Communication over MIMO X channels: Interference alignment, decomposition, and performance analysis. *IEEE Transactions on Information Theory*, 54:3457–3470, August 2008.

[45] Mohammad Maddah-Ali and David Tse. Completely stale transmitter channel state information is still very useful. *Proceedings of Allerton Conference on Communication, Control, and computing.*, September 2010.

[46] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[47] L. H. Ozarow. The capacity of the white Gaussian multiple access channel with feedback. *IEEE Transactions on Information Theory*, July 1984.

[48] Zhengang Pan, Kai-Kit Wong, and Tung-Sang Ng. Generalized multiuser orthogonal space-division multiplexing. *IEEE Transactions on on Wireless Communications*, 3(6):1969–1973, November 2004.

[49] Sameer Pawar, S. ElRouayheb, and Kannan Ramchandran. On secure distributed data storage under repair dynamics. *Proceedings of the IEEE International Symposium on Information Theory, Austin, USA*, June 2010.

[50] S. W. Peters and R. W. Heath. Interference alignment via alternating minimization. *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing, Taipei, Taiwan*, April 2009.

[51] Vinod Prabhakaran and Pramod Viswanath. Interference channels with source cooperation. *IEEE Transactions on Information Theory*, 57(1):156–186, January 2011.

[52] K. V. Rashmi, N. B. Shah, P. V. Kumar, and K. Ramchandran. Explicit construction of optimal exact regenerating codes for distributed storage. *Allerton Conference on Control, Computing and Communication*, September 2009.

[53] J. P. M. Schalkwijk and T. Kailath. A coding scheme for additive noise channels with feedback - part I: No bandwith constraint. *IEEE Transactions on Information Theory*, April 1966.

[54] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran. Explicit codes minimizing repair bandwidth for distributed storage. *Proceedings of the IEEE Information Theory Workshop, Cairo, Egypt*, January 2010.

[55] C. E. Shannon. The zero error capacity of a noisy channel. *IRE Transactions on Information Theory*, September 1956.

[56] C. E. Shannon. Coding theorems for a discrete source with a fidelity criterion. *IRE Nat. Conv. Rec.*, 1959.

[57] Changho Suh, Minnie Ho, and David Tse. Downlink interference alignment. *IEEE Transactions on Communications*, 59:2616–2626, September 2011.

[58] Changho Suh and Kannan Ramchandran. On the existence of optimal exact-repair MDS codes for distributed storage. *Submitted to the IEEE Transactions on Information Theory (arXiv:1004.4663)*, April 2010.

[59] Changho Suh and Kannan Ramchandran. Exact-repair MDS code construction using interference alignment. *IEEE Transactions on Information Theory*, 57(3):1425–1442, March 2011.

[60] Changho Suh and David Tse. Interference alignment for cellular networks. *Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing*, September 2008.

[61] Changho Suh and David Tse. Feedback capacity of the Gaussian interference channel to within 1.7075 bits: the symmetric case. *arXiv:0901.3580v1*, January 2009.

[62] Changho Suh and David Tse. Feedback capacity of the Gaussian interference channel to within 2 bits. *IEEE Transactions on Information Theory*, 57(5):2667–2685, May 2011.

[63] Ravi Tandon and Sennur Ulukus. Dependence balance based outer bounds for Gaussian networks with cooperation and feedback. *submitted to the IEEE Transactions on Information Theory*, December 2008.

[64] Emre Telatar and David Tse. Bounds on the capacity region of a class of interference channels. *IEEE International Symposium on Information Theory*, June 2007.

[65] David Tse and Pramod Viswanath. *Fundamentals of Wireless Communication*. Cambridge, 1 edition, 2005.

[66] Daniela Tuninetti. On InterFerence Channel with Generalized Feedback (IFC-GF). *IEEE International Symposium on Information Theory*, June 2007.

[67] Pramod Viswanath and David Tse. Sum capacity of the multiple-antenna Gaussian broadcast channel and uplink-downlink duality. *IEEE Transactions on Information Theory*, 49(8):1912–1921, August 2003.

[68] Frans M. J. Willems. The feedback capacity region of a class of discrete memoryless multiple access channels. *IEEE Transactions on Information Theory*, 28:93–95, January 1982.

[69] Frans M. J. Willems and E. C. van der Meulen. The discrete memoryless multiple-access channel with cribbing encoders. *IEEE Transactions on Information Theory*, 31:313–327, May 1985.

[70] Y. Wu. A construction of systematic MDS codes with minimum repair bandwidth. *arXiv:0910.2486*, October 2009.

[71] Y. Wu, P. A. Chou, and S. Y. Kung. Information exchange in wireless networks with network coding and physical-layer broadcast. *CISS 39th Annual Conference*, March 2005.

[72] Y. Wu and A. G. Dimakis. Reducing repair traffic for erasure coding-based storage via interference alignment. *Proceeding of the IEEE International Symposium on Information Theory*, 2009.

[73] Y. Wu, A. G. Dimakis, and Kannan Ramchandran. Deterministic regenerating codes for distributed storage. *Allerton Conference on Control, Computing and Communication*, September 2007.

[74] Raymond W. Yeung. *Information Theory and Network Coding*. Springer, 2008.