ADAPTIVE TRANSMISSION AND DYNAMIC RESOURCE ALLOCATION IN COLLABORATIVE COMMUNICATION SYSTEMS

by

Mai Zhang

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

School of Electrical and Computer Engineering

West Lafayette, Indiana

August 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Borja Peleato, Chair

School of Electrical and Computer Engineering

Dr. David Love

School of Electrical and Computer Engineering

Dr. James Lehnert

School of Electrical and Computer Engineering

Dr. James Krogmeier

School of Electrical and Computer Engineering

Dr. Robert Givan

School of Electrical and Computer Engineering

Approved by:

Dr. Dimitrios Peroulis

To my parents

For sheltering me as I grow up and teaching me the right things to do in life.

ACKNOWLEDGMENTS

I am very thankful for all the help, guidance, and influence that my advisor Prof. Borja Peleato gave me. I would have never accomplished anywhere close to what I have now without all your generous support. Your wisdom, kindness, and mentoring have always inspired me. Thank you.

I would also like to thank everyone on the BAM! Wireless team, including Stephen Larew, Tomohiro Arakawa, Dennis Ogbe, among many others. We fought countless days and nights developing and debugging our software defined radio network which won prizes in the DARPA SC2 challenge. Their generous help is much appreciated. I truly learnt a lot through this experience.

I am also thankful for every professor on my advisory committee, who all provided me crucial support and advice at some point during my years at Purdue. I especially want to thank Prof. Bob Givan for the two semesters of busy yet fruitful TA experience and the wonderful discussions we had. I learnt that the best way to learn something is by teaching it to others.

Last but not least, I want to thank all the friends that I made along the way, for making my life at Purdue colorful and memorable. I will forever cherish the good times and the memories that always make me smile.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

AF	amplify-and-forward
ARQ	automatic repeat request
AWGN	additive white Gaussian noise
BER	bit error rate
BLER	block error rate
BPSK	binary phase-shift keying
CIL	CIRN interaction language
CIRN	collaborative intelligent radio network
DF	decode-and-forward
ECC	error correcting code
FER	frame error rate
GA	genetic algorithm
HARQ	hybrid ARQ
IR	incremental redundancy
LDPC	low-density parity-check
LLR	log-likelihood ratio
MDP	Markov decision process
pdf	probability density function
PSD	power spectral density
QAM	quadrature amplitude modulation
QC-LDPC	quasi-cyclic LDPC
SC2	Spectrum Collaboration Challenge
SDR	software defined radio
SINR	signal-to-interference-plus-noise ratio
SNR	signal-to-noise ratio
SRN	standard radio node
TDMA	time-division multiple access

ABSTRACT

With the ever-growing demand for higher data rate in next generation communication systems, researchers are pushing the limits of the existing architecture. Due to the stochastic nature of communication channels, most systems use some form of adaptive methods to adjust the transmitting parameters and allocation of resources in order to overcome channel variations and achieve optimal throughput. We will study four cases of adaptive transmission and dynamic resource allocation in collaborative systems that are practically significant. Firstly, we study hybrid automatic repeat request (HARQ) techniques that are widely used to handle transmission failures. We propose HARQ policies that improve system throughput and are suitable for point-to-point, two-hop relay, and multi-user broadcast systems. Secondly, we study the effect of having bits of mixed SNR qualities in finite length codewords. We prove that by grouping bits according to their reliability so that each codeword contains homogeneous bit qualities, the finite blocklength capacity of the system is increased. Thirdly, we study the routing and resource allocation problem in multiple collaborative networks. We propose an algorithm that enables collaboration between networks which needs little to no side information shared across networks, but rather infers necessary information from the transmissions. The collaboration between networks provides a significant gain in overall throughput compared to selfish networks. Lastly, we present an algorithm that allocates disjoint transmission channels for our cognitive radio network in the DARPA Spectrum Collaboration Challenge (SC2). This algorithm uses the real-time spectrogram knowledge perceived by the radios and allocates channels adaptively in a crowded spectrum shared with other collaborative networks.

1. INTRODUCTION

One of the key problems in communication systems design is the ability to handle transmission errors. If the communication channel is relatively static or varies very slowly in time, existing techniques such as beamforming, channel sounding, equalization, etc. can be used to counteract the effects of fading, multipath, and other physical channel characteristics. These techniques are widely used in present systems including LTE, 5G NR, and cable communications. However, future wireless communication systems such as millimeter wave (mmWave) communications, internet of things (IoT), vehicular communications, etc. pose difficulties in beam alignment and channel sounding, and the communication channel is expected to exhibit greater variability and less favorable conditions. A robust protocol capable of handling transmission errors and decoding failures is therefore crucial in such systems to maintain stable and continuous high speed connections.

A natural trade-off that emerges in communication systems is between having a high data rate and having low frame error probability. Using higher modulation and channel coding rate results in denser information rate in the transmission stream, but also increases the probability of decoding error at the receiver, and vice versa. Traditionally, some systems use a conservative configuration of modulation and coding scheme to guarantee a minimum frame success rate over a wide range of SNR to overcome channel variations; however, this is inefficient when the SNR is high. As a result, most modern systems use an adaptive approach to maximize data rate by utilizing some kind of feedback from the receiver.

However, such adaptive schemes cannot guarantee error-free delivery of information and are still prone to errors due to the stochastic nature of communication channels. Sudden obstacles blocking the line of sight, mobility of the users causing beam misalignment, or fast fading can all unexpectedly degrade the channel drastically. When decoding errors occur at the receiver, communication systems usually rely on automatic repeat request (ARQ) to invoke the retransmission of failed messages, assuming they are still relevant. (In latency-sensitive applications such as VoIP and live broadcast, a packet is deemed irrelevant if it fails to be delivered by a certain deadline, and in this case retransmission is unnecessary.) Traditional ARQ requires individual acknowledgement of every packet, and retransmissions are also packet-based. Hybrid ARQ (HARQ), however, allows the retransmission of a flexible number of additional parity bits, referred to as incremental redundancy (IR), whose generating matrix is pre-agreed between the transmitter and the receiver. The receiver will then be able to leverage these additional bits alongside the previously received codeword to retry decoding and to correct the errors. Such method avoids retransmitting the entire packet when only a few additional bits are sufficient for the receiver to decode the message, and hence greatly improves the overall efficiency. Clearly, there is another trade-off here: sending a larger number of IR bits reduces the risk of another decoding failure, but decreases the average data rate. This creates an opportunity to optimize and adapt the IR bits according to the channel conditions.

Another important problem that arises with communication networks is the dynamic allocation of resources. The resource allocation problem includes two aspects: sharing the time-frequency resources between multiple networks, and routing and scheduling the traffic within each network. Both aspects have their own challenges. In theory, it is possible for a network to achieve the theoretical maximum data rate with a centralized controller if it knows all the information. However, the problem of joint routing and scheduling is NP-hard, and even centralized control algorithms usually employ heuristics to simplify the computation. When multiple networks are sharing the same time-frequency resources, they need to work out a way to collaborate so that the spectrum is not overloaded with excessive interference, rendering it impossible for anyone to reliably transmit any information across. However, centralized control of multiple networks may not be feasible at all because the networks usually do not share relevant information such as the offered traffic rate, source and destination nodes, channel gains and so on. This adds to the difficulty of efficient resource allocation between networks. Practical systems typically resort to sensing-based techniques, or dividing the time and frequency into slices that are exclusively reserved for individual networks. Sensing-based approaches such as carrier-sense multiple access with collision avoidance (CSMA/CA) and DARPA XG significantly underperform theoretical limits, and slicing the spectrum into exclusive bands is not flexible enough to handle the fluctuation in network traffic, often resulting in a partition of the spectrum that correlates poorly with the actual traffic load of each network.

The recent advancement in software defined radios (SDRs) offers an extra dimension to solve this problem. Compared to traditional radios that use hardware and firmware to implement the signal processing chains, SDRs are much more flexible. Dynamic resource allocation in accordance with the traffic becomes easier to implement, and collaboration between SDR networks is possible with or without shared side information. Advanced techniques in AI may also be applied. DARPA launched the Spectrum Collaboration Challenge, aiming to address this problem. In the challenge, several SDR networks sharing the same spectrum are allowed to exchange partial information over a side channel for collaboration, and they must dynamically and autonomously allocate the resources both between and within the networks.

This thesis studies four cases of adaptive approaches used by collaborative communication systems for transmission and resource allocation, and proposes optimized schemes that can be used in each of these cases. They are outlined as follows:

Chapter 2 reviews the existing HARQ techniques, and proposes a new adaptive HARQ scheme that maximizes the overall throughput by acknowledgement bundling in a limited feedback system [1]. It models the HARQ process as a Markov decision process (MDP) and optimizes the lengths and types of IR that the transmitter should send based on the SNR and coding rate of the received codewords, and the number of decoding failures in a bundle of codewords. In addition, it studies how such techniques generalize to multi-hop relay systems and multi-user broadcast systems. It proposes a scheme for the relay to decide between amplify-and-forward (AF) and decode-and-forward (DF) based on the estimation of channel SNR, where codeword errors at the end user will be corrected by HARQ from the base station or from the relay. Finally it proposes and optimizes the HARQ policies suitable for a transmitter that broadcasts bundled codewords to multiple receivers. It shows the advantage of using bundled codewords and IR in the broadcast setting compared to individual retransmission.

Chapter 3 proposes and studies a technique for grouping the bits transmitted through a wireless channel into codewords according to their SNR quality [2]. It proves that by splitting the bits into multiple codewords and encoding the higher quality bits with higher coding rate and lower quality bits with lower rate, the throughput can be improved compared to mixing heterogeneous quality bits into fixed-rate codewords. The chapter first analyzes the pros and cons of different mappings of bits and codewords to the available time, frequency, and modulation resources. Then it describes the proposed bit splitting scheme for 16-QAM modulation, where the asymmetric mapping of bits into modulation symbols creates bits of heterogeneous qualities. Simulations show the benefits of our proposed scheme. Finally, the chapter presents a mathematical proof of the gain in feasible throughput resulting from the proposed bit splitting technique in a binary-input parallel AWGN channel with finite blocklength error correcting codes (ECC). The proposed scheme can be applied to any communications channel using ECC, but it is of particular interest for mmWave wireless communications, where the channel quality is closely monitored and high order modulations are used over wide bandwidths. The simulations suggest that modest gains in throughput can be obtained with negligible additional complexity.

Chapter 4 studies the routing and resource allocation problem in multiple collaborative networks [3]. It first proposes a method based on a genetic algorithm (GA) and dual gradient ascent to optimize the routing, scheduling, and transmit powers in a single network, assuming that centralized control is possible by a certain node that has full knowledge of the network's properties. It then extends this method to the scenario of multiple networks sharing the same time-frequency resources, but do not have a protocol to share any side information with each other. Our proposed algorithm infers certain necessary information entirely from the network nodes' transmissions, and uses that information to enable collaboration between the networks. We show that by discarding transmission links according to their rate-to-power ratios, the system of collaborative networks is able to achieve a significant gain in overall throughput compared to selfish networks that greedily assign their own transmissions and treat peer networks' transmissions as noise. Simulations also find that our proposed scheme provides a moderate gain compared to the even splitting of resources (i.e., slicing the spectrum into exclusive bands) for heterogeneous networks, especially when the offered data rate is high.

Chapter 5 presents an adaptive algorithm that allocates disjoint transmission channels in a cognitive radio network, which was used in our submission to the DARPA Spectrum Collaboration Challenge (SC2). This challenge which lasted for 3 years seeks to find a solution for the long-existing problem of radio spectrum scarcity, which is getting more and more significant with the increasingly rapid growth in the worldwide demand of data over the wireless spectrum. In SC2, the participating teams were challenged to each develop an intelligent SDR network, and the networks from different teams competed against each other to deliver as much traffic as possible within their networks. The spectrum was shared and very crowded, and so the teams had to collaborate in order to deliver data without jamming each other. The dynamic channel allocation algorithm that this thesis proposes takes into account both the spectrum availability at our own receiving nodes by reading their power spectral density (PSD) measurements, as well as the interference that a transmission channel may cause to other teams. This algorithm promotes collaboration and allows for potential inter-network spectrum reuse if the interference level is low.

2. OPTIMIZING HARQ AND RELAY STRATEGIES IN LIMITED FEEDBACK COMMUNICATION SYSTEMS

One of the key challenges for future communication systems is to deal with fast changing channels due to the mobility of users in 5G wireless networks, IoTs etc. Having a robust protocol capable of handling transmission failures in unfavorable channel conditions is crucial, but the feedback capacity may be greatly limited due to strict latency requirements. This chapter studies the hybrid automatic repeat request (HARQ) techniques involved in re-transmissions when decoding failures occur at the receiver and proposes a scheme that relies on codeword bundling and adaptive incremental redundancy (IR) to maximize the overall throughput in a limited feedback system. In addition to the traditional codeword extension IR bits, this chapter introduces a new type of IR, bundle parity bits, obtained from an erasure code across all the codewords in a bundle. The type and number of IR bits to be sent as a response to a decoding failure is optimized through a Markov Decision Process. In addition to the single link analysis, the chapter studies how the same techniques generalize to relay and multi-user broadcast systems. It proposes a scheme for the relay to decide between amplify and forward (AF) and decode and forward (DF), and optimizes the HARQ policies suitable for a broadcast transmitter using bundled codewords. Simulation results show that the proposed schemes can provide a significant increase in throughput over traditional HARQ techniques.

2.1 Introduction

Communication systems are naturally prone to varying channel conditions. Before the recent information explosion, it was common for systems to use conservative configurations which allowed them to operate in a wide range of conditions, but this came at the expense of performance. In order to accommodate the ever growing traffic requirements of next generation communication devices, researchers are now using adaptive schemes to maximize bandwidth efficiency and squeeze as much throughput as possible in every situation. A significant amount of work has been devoted to designing algorithms for adapting physical layer parameters such as the transmit power, modulation and coding rate based on the channel state information [4]–[8], but there is not as much literature on adaptive retransmissions when failures occur despite it has been shown that they can provide significant gains in terms of both throughput [9]–[11] and outage probability [12], [13].

Traditional automatic repeat request (ARQ) forces the receiver to send an ACK back to the transmitter for every packet it successfully decodes, and a NACK otherwise. If the transmitter does not receive an ACK before the timeout expires, the entire packet will be resent, assuming that it is still within the latency limit. Retransmitting the whole packet is justified when the previous one has been completely lost, but in many cases the received packet can be partially recovered, and it still contains useful information for the decoder, even if it cannot be entirely decoded. In those cases, it is more efficient if the receiver can recover the whole packet with the help of a few additional bits sent from the transmitter, referred to as incremental redundancy (IR). This is commonly known as Type-II hybrid automatic repeat request (Type-II HARQ) [14], and it is the focus of this chapter.

The achievable data rate (throughput) with Type-II HARQ has been upper-bounded under the assumption of unlimited single bit IR and perfect feedback [9], [15] and several methods have been proposed to construct IR bits [16], [17] or optimize their block lengths under a finite number of retransmissions [18], [19]. However, most of these works have focused on extending idealized error correcting codes (ECC) in known channels with either infinite or single bit feedback. Some works have proposed more realistic models accounting for system-level constraints [20]–[22], bundling multiple packets in one resource block [11], [23] and imperfect channel information [24]. The first part of this chapter takes one step further in this direction by introducing a new type of IR bits and proposing frameworks to optimize the number and type of IR bits to be sent in scenarios with imperfect ECC, limited feedback, packet bundling, and overhead costs for each round of incremental redundancy. It models the problem as a Markov decision process (MDP) which minimizes the average cost per information bit delivered, relying on a code-specific Gaussian model for the probability of decoding failure as a function of SNR and code rate. By adjusting the relative costs associated to decoding and retransmissions, this method can be used to model practical constraints such as latency and hardware.

HARQ has been included and widely deployed in recent cellular networks such as LTE [25], [26] and 5G NR [27], and there are studies that evaluate its performance [28]. However, most standards proposed the use of fixed-length IRs due to practical constraints. LTE generally assigns one bit feedback per transport block, equivalent to one bit per codeword. The 5G NR standard includes multiple types of operations and is a little more flexible, but does not rise to the level that we propose in this chapter. It still uses ACK or NACK and pre-fixed IR lengths. This chapter shows that our proposed HARQ strategies can potentially achieve higher throughput by allowing even more flexibility than 5G NR in the types and lengths of IR retransmissions. In terms of channel coding, the punctured turbo codes in LTE have been replaced with low-density parity-check (LDPC) codes in 5G NR. Among other advantages, LDPC codes provide more flexible puncturing and rate adaptation, allowing for a nearly continuous number of IR bits. The ideas in this chapter can be applied to any family of channel codes, but assume the use of LDPC codes by default.

Upcoming millimeter wave (mmWave) systems are likely to deploy dense networks of access points acting as relays between a base station and the end users, requiring HARQ strategies suitable for multi-hop architectures [29]–[31]. These relay nodes have to decide between using amplify and forward (AF) or decode and forward (DF) when passing on information. AF amplifies and retransmits incoming packets as they are received, signal and noise. It is faster and less complex but noise accumulates over multiple hops until the packet can become unrecoverable. DF decodes the received signal and reencodes it before retransmission. This provides noise reduction and early detection of failures, but the required processing increases latency, complexity, and power consumption. Previous literature has shown that DF generally has higher channel capacity than AF [32] and lower frame error rate (FER) with several HARQ protocols [33]. However, some of the practical benefits of AF, such as simpler hardware and lower latency, were not considered in those studies. Hybrid schemes between AF and DF, such as transcoding [34] or compress and forward [35], have been shown to reduce the latency in modern 5G relay systems. Those schemes address how information is processed by the relay in each transmission, but do not address how to proceed when decoding failures occur. The second part of this chapter shows how the MDP framework initially proposed for a single link scenario can be easily extended to account for the complexities of a relay system, including optimizing the decision between AF and DF.

Finally, the third part of this chapter addresses another very relevant scenario for modern and future networks: multi-user systems where a single base station is communicating with multiple recipients. Even if each recipient is only interested in some of the information, it makes sense for the base station to bundle several packets and broadcast the bundle to all the users. If multiple users suffer a small number of decoding failures, the base station does not need to send individual IR to everyone; instead, it can broadcast one additional piece of information – for example the XOR of the packets in the bundle – to help multiple users decode their failed codewords. This idea, commonly known as network coded (H)ARQ, dates back to the 1980s [36], but it has recently experienced a renewed interest from the research community due to its potential uses in the Internet of Things (IoT). Its maximal achievable throughput under idealized conditions was characterized in [37], and [38] extended that work with a deeper study of the practical overheads associated with various implementations. It showed that using general linear codes requires significantly more overhead than binary codes, since the transmitter not only needs to specify which packets are included in each linear combination, but also their coefficients. Hence, this chapter only considers binary XOR packet combinations. The choice of packets to include is then a special case of the well known index coding problem [39], [40], but our framework also requires optimizing the number of bits to be sent, which further complicates the problem. Still, we show that it can be formulated in a relatively simple convex form. Numerical convex optimization algorithms can then be applied to solve for a good approximation to the optimum.

The main contributions of the chapter can be summarized as follows. It introduces a new type of IR bits, bundle parity bits, computed across a bundle of codewords. It proposes a MDP model for the HARQ process over a point-to-point link, optimizing the type and number of IR bits to be sent when failures occur. It then shows how such HARQ scheme can be generalized and adapted to suit a two-hop relay network, where the relay node can be optimized to choose between AF and DF based on the channel state information. Finally, it considers a multi-user broadcast scenario and shows that the optimization of the HARQ can be formulated in a convex form. Numerical simulations verify the derivations, and show

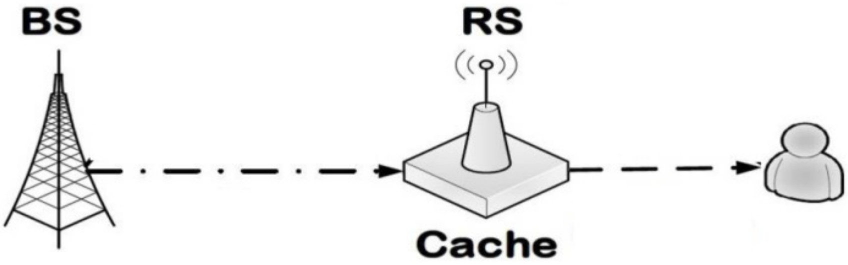


Figure 2.1. Relay system model

that the proposed methods achieve modest improvements against traditional schemes in all three scenarios.

The rest of the chapter is organized as follows. Section 2.2 explains the system model and some notation to be used throughout the chapter. Section 2.3 introduces the different types of IR bits being considered and how they can help in the decoding of a given bundle of codewords. Section 2.4 builds a single-link decision engine optimizing the type and number of IR bits to be sent as a function of the channel SNR, coding rate, and number of failed codewords in the bundle. Section 2.5 derives the decision engine for the relay, which decides between AF and DF relay strategies as a function of the SNRs on the two links and the code rate on the first link. Section 2.6 addresses the case of multi-user systems, proposing a combinatorial optimization algorithm for deciding how the failed codewords should be grouped for the generation of IR when failures occur. Finally, section 2.7 illustrates the performance of our proposed policies through numerical simulations, and section 2.8 concludes the chapter.

2.2 System Model

This section introduces the system models used throughout the chapter. It first presents a single link scenario (with a direct channel between the transmitter and receiver) describing the channel, modulation, ECC, and HARQ schemes. Then it extends this scenario to the dual-hop relay system depicted in Fig. 2.1, where the base station (BS) can only reach the end user through an intermediate relay station (RS), and to a multi-user scenario where a single transmitter (possibly the relay) is communicating with multiple recipients. All the links in the relay and multi-user scenarios follow the same model as that in the single link scenario.

2.2.1 Channel and Modulation

Modern communication systems estimate the channel by periodically sending pilot signals, and use those estimates to adjust the modulation and coding schemes so as to maintain a certain frame error rate (FER). However, the unpredictable nature of channels and the blind period between channel sounding cycles make it impossible to achieve optimal adaptation for all codewords.

In this chapter, channels are modeled as interference-free AWGN. We assume that multiple codewords (or packets[1]) are bundled together into a single block, experiencing the same (often unknown) SNR at the receiver. All the IR bits requested in the same round also experience the same SNR, but this SNR is independent from that for the bundle and for previous rounds of IR (if any). This assumption is made in light of the fact that there is typically a delay between the transmissions of the original bundle and the IR, during which channel conditions could have changed.

In order to increase throughput, the transmitter uses high-order modulations with multiple bits per symbol for all but the noisiest channel conditions. Encoding these modulation symbols directly would increase the error correction capabilities [41], but would complicate significantly the encoding and decoding. Treating the bits in a modulation symbol as independent and using binary error correcting codes is significantly simpler computationally, specially in the case of LDPC codes, but the performance is slightly worse than with non-binary error correction codes. Still, it is the most common approach in practice. Therefore, this chapter assumes the use of binary encoders and decoders which operate as if the bits came from independent BPSK modulations with constant SNR throughout each codeword, even if higher order modulations are actually being used [2].

[1] ↑For simplicity, we assume that each packet consists of a single codeword and we refer to packets or codewords indistinctly. In a scenario where each packet consists of multiple codewords, we can either acknowledge codewords independently or treat each packet as a single unit which can either succeed or fail to decode.

2.2.2 Error Correction

Several works (*e.g.* [18], [42]) have shown that the FER of a finite length code can be well approximated by

$$P_e(R, SNR) = Q\left(\frac{\mu - R}{\sigma}\right), \tag{2.1}$$

where R represents the code rate (*i.e.* number of information bits divided by codeword length) and μ and σ are code-specific parameters that depend on the SNR. We model such dependency as

$$\mu = a_\mu \cdot SNR^{-c_\mu} + b_\mu, \tag{2.2}$$

$$\sigma = a_\sigma \cdot SNR^{-c_\sigma} + b_\sigma. \tag{2.3}$$

The techniques proposed in this chapter could be applied to any code by adjusting the parameters (a_μ, b_μ, c_μ, a_σ, b_σ, c_σ), but the numerical simulations in this chapter will focus on the binary QC-LDPC code of length $n = 648$ and $k = 432$ (rate 2/3) proposed in the 3GPP standard for 802.11n [43], for illustrative purposes. Our prior work [44], [45] showed that

$$a_\mu = -0.2 \qquad b_\mu = 0.86 \qquad c_\mu = 1.74 \tag{2.4}$$

$$a_\sigma = 0.12 \qquad b_\sigma = -0.08 \qquad c_\sigma = 0.42 \tag{2.5}$$

provide a good fit to this code when $SNR \in [0.5, 2]$.

Binary QC-LDPC codes offer very efficient encoding and decoding using parallel shift registers [46], [47]. This has made them the preferred option in 5G NR, over turbo codes such as the ones proposed in the LTE standard. Additionally, QC-LDPC codes can be flexibly punctured and extended for nearly continuous rate adaptation. A QC-LDPC code is uniquely defined by a sparse parity check matrix $H \in \{0,1\}^{(n-k)\times n}$, such that $Hx = \mathbf{0}$

for all codewords x. Received channel values (*i.e.* matched filter outputs) are processed to obtain a log-likelihood ratio (LLR) for each individual bit b as

$$\ell = \mathrm{LLR}(b|r) = \log \frac{p(b=0|r)}{p(b=1|r)}, \tag{2.6}$$

where $p(0|r)$ and $p(1|r)$ represent the conditional probability of $b = 0$ and $b = 1$, respectively, given the received value r. It is not hard to prove that for an AWGN channel with equiprobable and symmetric inputs, the LLR values are given by

$$\mathrm{LLR}(b|r) = 2 \cdot \mathit{SNR} \cdot r. \tag{2.7}$$

The decoding of LDPC codes is typically done through message-passing algorithms, which refine these LLR values iteratively until convergence or until a prefixed maximum number of iterations is reached. When the algorithm does converge, it is almost always to the right codeword. We thus assume that a codeword error occurs if and only if the LDPC decoder fails to converge.

2.2.3 Single Link System: Hybrid ARQ

This chapter focuses on the optimization of HARQ protocols, abstracting some of the other practical complexities that are present in real world communication networks. For example, the chapter assumes perfect synchronization between all the nodes and error-free, albeit limited-capacity, feedback links. Feedback links are assumed to offer no more than one bit of feedback per packet, allowing for 256 possible responses to a bundle of 8 packets, for instance. However, most of the proposed HARQ strategies do not require that many feedback messages, so the required number of feedback bits can be lower.

It is also assumed that the receiver can request as many rounds of incremental redundancy as needed until the whole bundle is successfully decoded. Each round is penalized with an adjustable overhead cost of c_R per link plus a decoding cost of c_D for each codeword for which decoding is attempted.

2.2.4 Relay System: Amplify or Decode?

In the relay scenario, the intermediate node needs to decide whether to adopt an AF or DF strategy for each incoming bundle. It will base this decision on the channel SNR estimates and the code rate of the bundle. With DF, the system is equivalent to two separate links, which could be independently optimized using the same HARQ protocol as for the single link scenario. With AF, the HARQ problem is slightly more complex. When a bundle arrives, the relay will forward it without any processing, but we assume that it caches the LLR values temporarily. If the end user is successful in decoding the whole bundle, these LLR values can be discarded, but if the end user suffers any decoding failures, the relay reverts to DF. It decodes the bundle using its cached LLR values (employing HARQ if needed) and only after having succeeded it sends IR to the end user.

When employing AF, we assume unit transmit power at the base station and that the relay amplifies its received signal to invert the attenuation of the first channel. In other words, if the relay receives

$$y_1 = g_1 x + n_1,$$

where g_1 is the channel gain on the first link, x is the signal with $E[x^2] = 1$ and n_1 is Gaussian noise with variance σ_1^2, the relay amplifies y_1 by a factor $1/g_1$ before forwarding it. Then, the received signal at the end user is

$$\begin{aligned} y_2 &= g_2 \frac{1}{g_1} y_1 + n_2 && (2.8) \\ &= g_2 \Big(x + \frac{n_1}{g_1} + \frac{n_2}{g_2} \Big), && (2.9) \end{aligned}$$

where g_2 is the gain over the second link. Since the noise components n_1 and n_2 are independent, the SNR at the end user with AF is

$$SNR_{AF} = \frac{E[x^2]}{\mathrm{Var}\left[\frac{n_1}{g_1} + \frac{n_2}{g_2}\right]} = (SNR_1^{-1} + SNR_2^{-1})^{-1}, \tag{2.10}$$

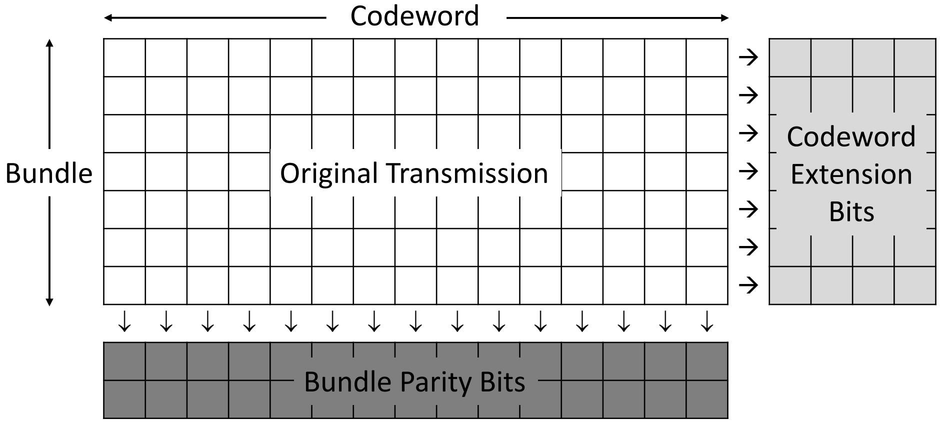


Figure 2.2. Types of incremental redundancy.

where $E[\,\cdot\,]$ and $\mathrm{Var}\,[\,\cdot\,]$ denote expectation and variance respectively, and $SNR_j = g_j^2/\sigma_j^2$ $(j = 1, 2)$ is the SNR on the j-th link. Note that SNR_{AF} is always lower than the SNR on either link.

2.2.5 Multi-user Systems

The last scenario studied in this chapter is that of a single transmitter communicating with multiple recipients. Each recipient is only interested in a subset of the information being transmitted, but can overhear everything. Each receiver has its own data and feedback channel, with independent SNR and decoding process. When a receiver is unable to decode its desired information, it reports the failures to the transmitter and requests IR. The transmitter compiles the failure reports from all the receivers and uses the proposed algorithm to optimize the set of IR bits that should be broadcast in order to ensure that none of them suffers a probability of error above a pre-fixed value γ. This optimization is formulated as a convex optimization problem, albeit with the number of variables increasing exponentially with the number of failures reported. In any case, if the number of failures is too large, it is usually better to re-transmit the whole bundle anyway.

2.3 Incremental Redundancy

This chapter uses the term “Incremental Redundancy” (IR) to denote all the bits transmitted with the objective of aiding in the recovery of one or more codewords whose decoding had previously failed. Fig. 2.2 shows three different types of IR:

1. Chase Combining [48]: the sequence of IR bits is identical to a subset of the bits previously sent. It is simple and computationally efficient, since the transmitter does not need to generate new bits and the decoder can just refine the previous LLRs using maximal ratio combining. However, some of the information transmitted might be redundant to the receiver, so it is a suboptimal approach.

2. Bundle parity bits: the sequence of IR bits consists of a bit-wise erasure code over the previously transmitted codewords [49]. This chapter uses the XOR of the codewords in a bundle, unless stated otherwise.

3. Codeword parity (or extension) bits: the sequence of IR bits extends each of the previously transmitted codewords with either previously punctured bits or with completely new parity found by adding new rows and columns to the parity check matrix H.

We assume that the decoder can handle the decoding of a (possibly extended) codeword, but does not have enough memory to jointly decode all the codewords in a bundle. Each codeword is therefore decoded independently, although Chase Combining and bundle parity bits can be used to refine its LLR values.

We now study the effect that each of these types of IR bits has on the codewords. In a nutshell, Chase Combining and bundle parity bits increase the SNR for some bits in the failed codewords, and extension bits reduce the rate of the codeword. These improvements in SNR and rate can be translated into a lower probability of error using Eq. (2.1).

2.3.1 Chase Combining

Let $r^{(0)} = b + n^{(0)}$ and $r^{(1)} = b + n^{(1)}$ be the received values corresponding to two transmissions of the same bit b with different SNR_0 and SNR_1, respectively. With Chase

Combining, the receiver can combine both values into $r^{(0)} + r^{(1)} = 2b + n^{(0)} + n^{(1)}$ resulting in an effective SNR of

$$SNR_{\mathrm{CC}} = \frac{4}{\frac{1}{SNR_0} + \frac{1}{SNR_{\mathrm{IR}}}}, \tag{2.11}$$

for the retransmitted bits. Since $p(1|r^{(0)}, r^{(1)})$ is proportional to $p(1|r^{(0)})p(0|r^{(1)})$ (the same applies for $b = 0$), the decoder can just add the LLRs from the individual transmissions.

2.3.2 Bundle parity

Similarly to Chase Combining, bundle parity bits can be used to increase the SNR for some of the bits in the failed codewords. Assume that a vector $\mathbf{b} = [b_1, b_2, \cdots, b_n]$ of n bits from from different codewords is transmitted through an AWGN channel and that their XOR $x = b_1 \oplus \cdots \oplus b_n$ is transmitted through another AWGN channel with possibly different SNR. Denoting the received values for $\mathbf{b}$ and x as $\mathbf{r}$ and r_x, respectively, the probability of a specific bit b_k being 0 conditioned on these received values can be found as

$$p_k(0|\mathbf{r}, r_x) = \sum_{\substack{\mathbf{d} \in \{0,1\}^n \\ d_k = 0}} \frac{\left(\prod_{j=1}^{n} p_j(d_j|r_j)\right) p_x(\bigoplus \mathbf{d}|r_x)}{\sum_{\mathbf{v} \in \{0,1\}^n} \left(\prod_{j=1}^{n} p_j(v_j|r_j)\right) p_x(\bigoplus \mathbf{v}|r_x)}, \tag{2.12}$$

where $\bigoplus$ represents the XOR operator and $p_x(\bigoplus \mathbf{v}|r_x)$ denotes the probability that $x = v_1 \oplus \cdots \oplus v_n$ given the received value r_x. Eq. (2.12) provides the exact probabilities required for the computation of the LLR values, but it is impractical to evaluate for large bundle sizes because the number of terms increases exponentially. Hence, we adopt a similar approximation to that used in Min-Sum LDPC decoders [50] and calculate the updated LLR for bit b_k as

$$\ell_k^{\mathrm{new}} = \ell_k + \left(\prod_{\substack{i=1 \\ i \neq k}}^{n+1} \mathrm{sign}(\ell_i)\right) \min_{\substack{i=1\ldots n+1 \\ i \neq k}} \left|\ell_i\right|, \tag{2.13}$$

where ℓ_{n+1} denotes the LLR value for $x = \bigoplus \mathbf{b}$. The effect of this update can be modelled as an increase in the SNR of the bits using Eq. (2.7). Specifically,

$$SNR_{\text{new}} = \frac{E[\ell^{\text{new}}]^2}{\text{Var}\left[\ell^{\text{new}}\right]}, \tag{2.14}$$

where ℓ^{new} corresponds to the LLRs conditioned on $b = 0$ being transmitted[2]. The two terms in Eq. (2.13) are independent, so the moments of ℓ^{new} can be found by adding their corresponding moments. Characterizing the mean and variance of the minimum value among a set of Gaussians is possible, but requires tedious equations that add little value to this chapter. Instead, Fig. 2.3 illustrates the SNR_{new} as a function of the number of failed codewords and the SNR of the original bits, assuming a SNR of 0 dB for the IR. In a practical setting, that table would be computed offline and saved in memory to be used in the optimizations described in subsequent sections.

LDPC decoders can occasionally fail to converge, but when they converge to a feasible codeword it is almost always the right one. Therefore, when the decoder fails to decode some of the codewords in a bundle, the receiver can set the LLR values for successfully decoded codewords to have infinite magnitude and update those for the failed codewords according to Eq. (2.13) before attempting another decoding. If it succeeds in decoding any previously failed codewords, their LLRs can be scaled to have infinite magnitude and those for failed codewords can be updated again.

2.3.3 Codeword Extension

Finally, codeword extension bits reduce the rate of the code. The probability of a successful decoding with these extension bits is highly dependent on the specific code being used. The code specifications often characterize this probability, but only under the assumption that the original codeword and the extension bits are received with the same SNR. Unfortunately, this is generally not the case in practice.

[2] ↑The same formula would hold if $b = 1$ is being transmitted.

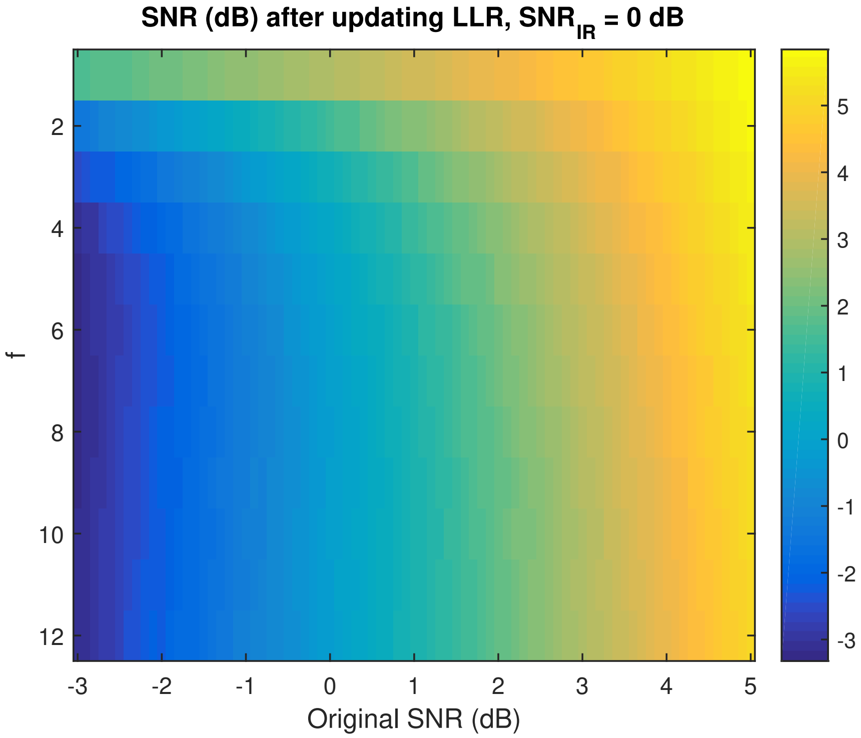


Figure 2.3. SNR after updating the LLRs of f bits based on a transmission of their XOR with $SNR_{IR} = 0$ dB.

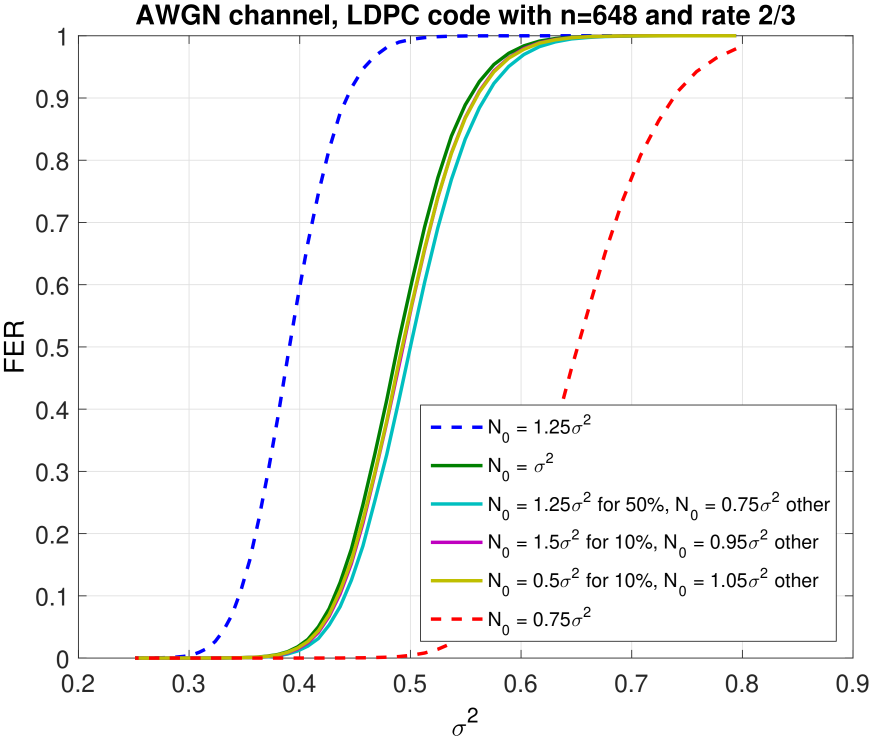


Figure 2.4. Probability of decoding failure for a signal with $E_b = 1$ and variable noise variance. The four solid curves, which correspond to combinations with the same SNR_{eff}, are nearly identical.

In order to simplify our derivations, we define the effective SNR of a codeword as

$$SNR_{\text{eff}} = \left(E\left[\frac{1}{SNR}\right]\right)^{-1}, \tag{2.15}$$

where the expectation is taken over the bits in the codeword. When all the bits in the codeword have the same energy E_b, SNR_{eff} is equivalent to dividing E_b by the average noise power. Fig. 2.4 illustrates the probability of decoding failure for different noise powers and distributions of signal strength within a codeword. Solid curves, which correspond to different distributions with the same SNR_{eff} are nearly identical, while dashed curves show the effect of a 25% variation in SNR_{eff}. We therefore assume that the probability of failure mostly depends on SNR_{eff}, not on the SNR variance within the codeword.

2.4 Decision Engine for Single Link

This section considers a point-to-point link, and proposes an optimization method where the requested number and type of IR bits can be chosen to minimize a cost function. We discretize the coding rate R and the SNR into a finite set of values so that practical numerical methods can be applied to the optimization problem. Since the feedback channel has limited capacity and only offers a few bits for each IR request, we constrain the number of IR bits to be requested to a small set of pre-defined values. A Markov Decision Process (MDP) can then be established to model the HARQ protocol as follows:

- State: $s = (f, \mathit{SNR}, R)$, where f denotes the number of decoding failures in the bundle, SNR their effective SNR, and R their coding rate.
- Action: $A(s) = (\alpha, \beta)$, where α and β respectively represent the requested number of extension bits for every codeword in the bundle and the requested number of bundle parity bits. Chase Combining bits will not be used because for typical values of SNR and code rate, their performance is inferior compared to extension bits [48].
- Cost: $C = b\alpha + \beta + f c_D + c_R$, where b denotes the bundle size (*i.e.* number of codewords per bundle). Assuming that transmitting one bit costs one unit, c_D denotes the cost to decode a single codeword, and c_R denotes the overhead cost due to each round of IR accounting for hardware complexity, increased latency, feedback bits, etc. One possible interpretation for this cost is latency. In that case, c_D would be the time required to decode a codeword and c_R the time between retransmissions.

The objective is to find the actions that minimize the total expected cost until all codewords in the bundle are successfully decoded, *i.e.*

$$A(s) = \arg\min_{(\alpha,\beta)} E\{\text{Total cost}|s, \alpha, \beta\} \tag{2.16}$$

for all s. By sending IR bits, α reduces the code rate and β increases the SNR, transitioning from $s_0 = (f_0, \mathit{SNR}_0, R_0)$ to a new state $s_1 = (f_1, \mathit{SNR}_1, R_1)$, where SNR_1 and R_1 are deterministic and $f_1 \leq f_0$ follows a binomial distribution. They can be determined by the following equations:

$$SNR_1 = \left[\left(\frac{\alpha}{SNR_{IR}} + \frac{\beta}{SNR_{\text{new}}} + \frac{k/R_0 - \beta}{SNR_0}\right)\frac{1}{k/R_0 + \alpha}\right]^{-1} \tag{2.17}$$

$$R_1 = \frac{k}{k/R_0 + \alpha} \tag{2.18}$$

$$P(f_1|s_0, \alpha, \beta) = \binom{f_0}{f_1} p^{f_1}(1-p)^{f_0 - f_1} \tag{2.19}$$

where k denotes the number of information bits per codeword and SNR_{new} denotes the increased SNR of the bits that participated in the bundle parity IR, as given by Eq. (2.14) and illustrated in Fig. 2.3. The formula for SNR_1 is obtained from Eq. (2.15) by observing that every codeword in a bundle can be partitioned into three sections according to the SNR: the α bits of codeword extension have SNR_{IR}, the β bits of overlapping part with bundle parity IR have SNR_{new} after updating their LLRs, and the remaining $k/R_0 - \beta$ bits keep the same SNR_0 as before receiving the IR.The probability p in Eq. (2.19) denotes the conditional probability that a codeword fails in state s_1 given that it failed in s_0, and can be computed using Eq. (2.1) as

$$p = \frac{P_e(R_1, SNR_1)}{P_e(R_0, SNR_0)}. \tag{2.20}$$

For any state s and SNR_{IR}, the total expected future cost V and the optimal action A can be expressed recursively as follows:

$$\begin{aligned} V(s, SNR_{IR}) &= E[\text{Total cost}|s, \alpha, \beta] \\ &= b\alpha + \beta + fc_D + c_R + \sum_{s'} P(s'|s, \alpha, \beta) V(s', SNR_{IR}) \end{aligned} \tag{2.21}$$

$$A(s, SNR_{IR}) = \arg\min_{(\alpha,\beta)} V(s, SNR_{IR}), \tag{2.22}$$

where the summation is taken over all possible states s' to which s can transition according to Eq. (2.17)-(2.19) given that (α, β) IR bits are sent. $P(s'|s, \alpha, \beta)$ denotes the state transition probability.

If we discretize the states and actions to take values from a finite set, the value iteration algorithm [51] can then be used to numerically find $V(s, \mathit{SNR}_{IR})$ and $A(s, \mathit{SNR}_{IR})$ for all s and SNR_{IR}. Essentially, value iteration initializes V with random values, and alternates between finding the optimal actions A according to Eq. (2.22) and updating the value V according to Eq. (2.21), until convergence. At that point $A(s, \mathit{SNR}_{IR})$ stores the optimal policy to follow when the HARQ process is at state s expecting SNR_{IR} for the IR, while $V(s, \mathit{SNR}_{IR})$ stores the total expected future cost until successfully decoding all codewords in the bundle at the receiver.

The single link scenario decision engine is specified by the policy A, and it can be readily extended to individual links in a multi-hop scenario as well. The receiver can estimate its state by computing the bundle's relevant statistics when decoding failures occur, and it then follows A to request a combination of (α, β) IR bits from the transmitter.

2.5 Decision Engine for Relay

This section extends the framework described in Section 2.4 to the two-hop scenario illustrated in Fig. 2.1. On top of optimizing the type and number of IR bits to be transmitted, the intermediate station also has to decide between using an amplify and forward (AF) or decode and forward (DF) relay strategy. In order to compare both strategies we propose a parametric cost model for each of them and a decision engine to minimize the average cost per successfully delivered information bit. Specifically, we model the cost of AF and DF (c_{AF} and c_{DF}) as functions of the SNR on both links (SNR_1 and SNR_2) and the code rate in the first link (R_1). As in the single link decision engine, the decoding cost c_D and the overhead cost c_R are normalized by the cost of transmitting 1 bit of information over one link.

2.5.1 Cost of DF

With a DF relaying strategy, both links can be treated as independent. Thus, the cost of DF is decomposed as

$$c_{DF} = c_1 + c_2, \tag{2.23}$$

where c_j is the expected cost on the j-th link ($j = 1, 2$). We further decompose each c_j as the sum of three terms: the number of bits sent on the j-th link, the cost of decoding the b codewords in the bundle, and the expected future cost in the case of decoding failures. Thus,

$$c_j = \frac{bk}{R_j} + bc_D + \sum_{i=1}^{b} P_B(b, p_j, i)\delta_j(i), \tag{2.24}$$

where $P_B(b, p_j, i) := \binom{b}{i} p_j^i (1 - p_j)^{b-i}$ represents the probability of suffering i failures in the bundle and $\delta_j(i)$ represents the expected future cost on the j-th link when that happens. The probability of failure $p_j = P_e(R_j, SNR_j)$ is obtained from Eq. (2.1) and

$$\delta_j(i) = V((i, SNR_j, R_j), SNR_{IR,j}) \tag{2.25}$$

is given by Eq. (2.21) from the single link scenario. The code rate on the second link R_2 should be chosen such that c_2 is minimized. For the sake of simplicity, we assume that the IR experiences the same SNR as the original codewords in the relay scenario, hence $SNR_{IR,j} = SNR_j$ for both links $j = 1, 2$.

2.5.2 Cost of AF

With an AF strategy, the relay is assumed to keep the code rate unchanged, *i.e.* $R_2 = R_1$, so the same number of bits is sent over both links in the first transmission. Decoding the bundle at the end user costs bc_D plus any cost associated to IR if failures occur. Thus, the cost of AF is decomposed as

$$c_{AF} = 2 \cdot \frac{bk}{R_1} + bc_D + \sum_{i=1}^{b} P_B(b, p_{AF}, i)\delta_{AF}(i), \tag{2.26}$$

where $p_{AF} = P_e(R_1, SNR_{AF})$, SNR_{AF} is taken from Eq. (2.10), and $\delta_{AF}(i)$ denotes the expected future cost when i failures are present at the end user. If decoding failures do occur, the end user will request IR from the relay. We assume that the relay always reverts back to a DF strategy in this case, decoding the cached bundle with IR from the base station if necessary. If there are j failed codewords at the relay, decoding the entire bundle will cost

$V((j, SNR_1, R_1), SNR_1)$. Once the relay has succeeded at decoding the whole bundle, it can generate and transmit the IR that the end user requested. This step costs another $V((i, SNR_{AF}, R_2), SNR_2)$.

With AF, the noise accumulates over the two links. It is therefore very unlikely for a codeword that could not be decoded at the relay to be correctly decoded at the end user. Similarly, if a codeword was correctly decoded by the end user we assume that it will also be successfully decoded by the relay. The number of failures at the relay then follows a binomial distribution with i representing the number of failures at the end user and p_R representing the conditional probability that a codeword fails at the relay conditioned on it failing at the end user. Hence,

$$\begin{aligned}\delta_{AF}(i) = bc_D &+ V((i, SNR_{AF}, R_2), SNR_2) \\ &+ \sum_{j=1}^{i} P_B(i, p_R, j) V((j, SNR_1, R_1), SNR_1), \end{aligned} \tag{2.27}$$

where

$$p_R = \frac{P_e(R_1, SNR_1)}{P_e(R_1, SNR_{AF})}$$

follows from Bayes' rule.

The values of c_{DF} and c_{AF} can now be computed for all discretized values of SNR_1, SNR_2, and R_1 using Eqs. (2.23) and (2.26). A decision map is then generated specifying whether AF or DF provides lower expected cost. According to this decision map, the relay can make the AF or DF decision by estimating the SNR on the two links and finding the rate of the received codeword in a practical situation.

2.6 Decision Engine for Multi-User Systems

This section addresses a system where a single server (or base station) uses a broadcast link to deliver content to multiple users. The channels from the base station (BS) to each user experience different and independent SNR, so when the BS broadcasts a bundle of codewords, each user is able to decode some of them but not others. If all users are interested in decoding all codewords the problem is similar to that of a single link: it makes sense to focus on the user with the most failures and broadcast the corresponding IR bits, that all other users are also able to hear and use in their own data recovery. However, we analyze the more interesting case in which each user is only interested in a subset of the codewords but can overhear and attempt to decode those meant for other users. Furthermore, we assume that users can report the specific codewords that they succeeded in decoding. In this case, the BS can leverage that information and use network coding schemes to optimize the IR [38], [39], [52], [53]. Since not all users are interested in all codewords, extension IR bits for any given codeword would only benefit a subset of the users, possibly a single one. Bundle parity bits obtained by taking the XOR of multiple codewords, however, have the potential to help multiple users decode their desired information. This section focuses on optimizing the choice of codewords in such combinations and the number of bundle parity bits to be sent for each of them.

Consider a bundle of codewords being broadcast to multiple users, so that user i is only interested in codeword i but overhears all the others. If user i can successfully decode codeword i, then it is done and does not require any IR. Our goal is to minimize the total number of IR bits sent while ensuring a minimal probability of success for all the users who failed to do so. Let b denote the number of users that fail to decode their corresponding codewords, and consider all the possible subsets of $\{1, \ldots, b\}$, indexed with numbers between 0 and $2^b - 1$. A simple way of doing this would be to use the binary representation of the elements included in the subset. Let $\Omega_k \subseteq \{1, \ldots, b\}$ represent the k-th such subset for $k = 0, \ldots, 2^b - 1$, so that $j \in \Omega_k$ if and only if $\lfloor k/2^{j-1} \rfloor$ is odd. Let β_k represent the number of IR bits to be sent obtained from the XOR of the codewords in Ω_k. Then the problem that we are trying to solve is

$$\text{minimize} \quad \sum_{k=0}^{2^b-1} \beta_k \tag{2.28}$$

$$\text{subject to} \quad P_e^{(i)} \leq \gamma, \qquad i = 1, \ldots, b \tag{2.29}$$

$$\beta_k \geq 0, \qquad k = 0, \ldots, 2^b - 1 \tag{2.30}$$

where $P_e^{(i)}$ represents the probability of user i failing to decode after receiving the IR, conditioned on having failed without it, and γ represents the highest such probability that we are willing to tolerate. The failure probability $P_e^{(i)}$ depends on $SNR_0^{(i)}$, the SNR of the original codeword, and on $SNR_{\text{eff}}^{(i)}$, the effective SNR after IR defined in Eq. (2.15). The latter is itself a function of $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_{2^b-1})$, as described below.

Let $\chi_i \in \{1, \ldots, b\}$ represent the indices of the codewords that user i failed to decode and assume that $i \in \chi_i$ (otherwise the user has received its information and is out of the picture). Receiving β_k bits from the XOR of codewords in Ω_k would help user i increase the SNR in β_k of the bits from codeword i. Fig. 2.3 provides the new SNR for those bits, denoted SNR_{new}, as a function of $SNR_0^{(i)}$ and the number of failed codewords in the XOR, denoted $f_k^{(i)} = |\chi_i \cap \Omega_k|$. Assuming that the IR updates do not overlap, the effective SNR for user i after IR would be

$$SNR_{\text{eff}}^{(i)}(\boldsymbol{\beta}) = \left[\frac{1}{SNR_0^{(i)}} + \frac{1}{n}\sum_{\{k:i\in\Omega_k\}} \beta_k \left(\frac{1}{SNR_{\text{new}}(f_k^{(i)}, SNR_0^{(i)})} - \frac{1}{SNR_0^{(i)}}\right)\right]^{-1}. \tag{2.31}$$

Eqs. (2.1)-(2.3) and (2.20) can be used to rewrite the error constraints in (2.29) as

$$\lambda_i a_\sigma z_i(\beta)^{c_\sigma} - a_\mu z_i(\beta)^{c_\mu} \leq \theta_i \tag{2.32}$$

for $i = 1, \ldots, b$, where

$$\lambda_i := Q^{-1}\left(\gamma P_e(R, SNR_0^{(i)})\right), \tag{2.33}$$

$$\theta_i := b_\mu - R - \lambda_i b_\sigma, \tag{2.34}$$

$$z_i(\boldsymbol{\beta}) := \left(SNR_{\text{eff}}^{(i)}(\boldsymbol{\beta})\right)^{-1}. \tag{2.35}$$

In the above definitions $P_e(R, SNR_0^{(i)})$ denotes the probability of error before IR and R the coding rate, assumed to be identical for all codewords for the sake of simplicity. Using the numerical values in Eqs. (2.4) and (2.5), problem (2.28) becomes

$$\begin{aligned} &\text{minimize} && \sum_{k=0}^{2^b-1} \beta_k \\ &\text{subject to} && 0.2 z_i(\boldsymbol{\beta})^{1.74} + 0.12 \lambda_i z_i(\boldsymbol{\beta})^{0.42} \leq \theta_i, \qquad i = 1, \ldots, b \\ & && \beta_k \geq 0, \qquad k = 0, \ldots, 2^b - 1. \end{aligned} \tag{2.36}$$

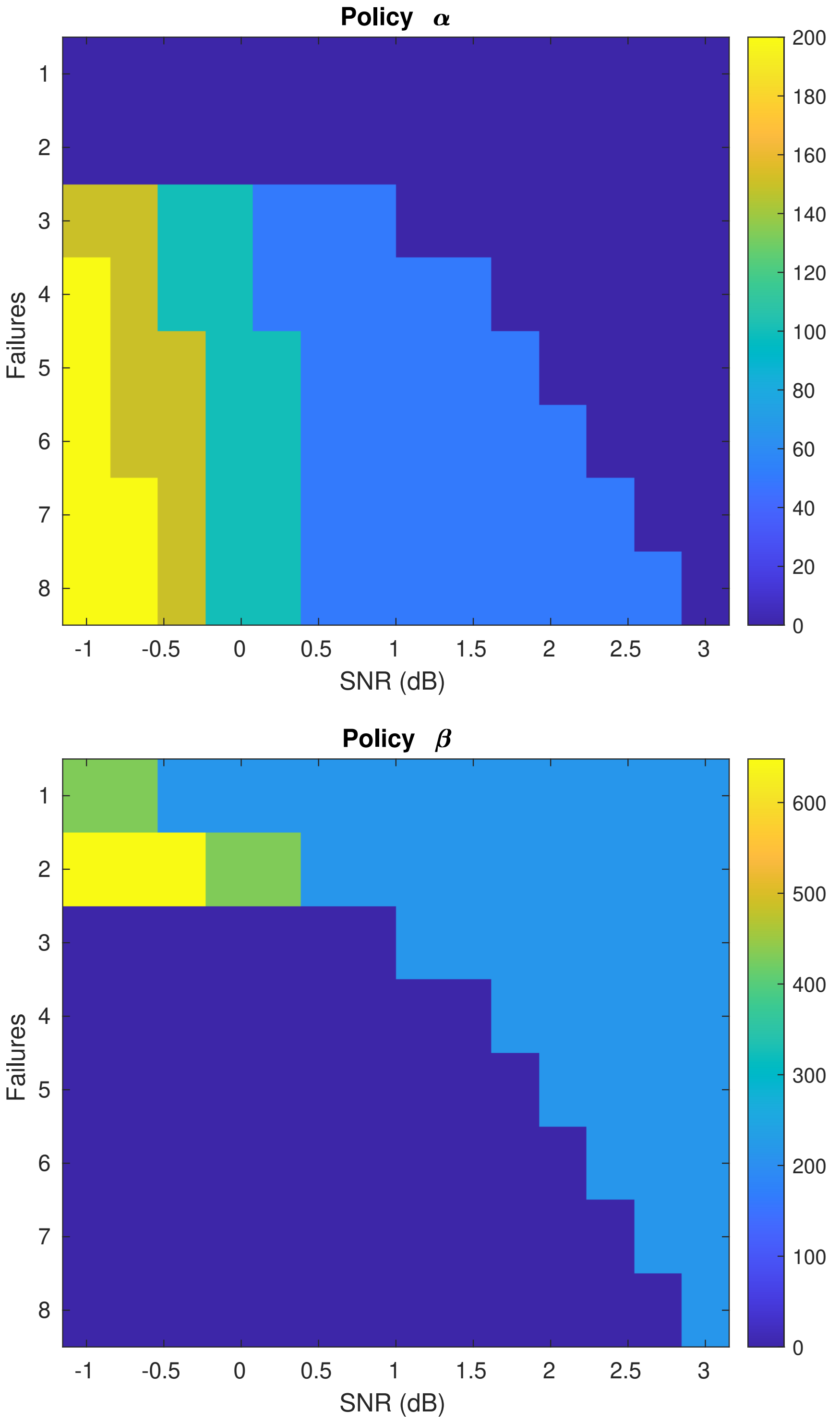
Observe that $z_i(\boldsymbol{\beta})$ is a linear function of $\boldsymbol{\beta}$, as shown in Eq. (2.31). Assuming that $z_i(\boldsymbol{\beta}) \geq 0.5$, since the model in Eq. (2.1) is not valid outside of that range, the above problem is convex for $\gamma \geq \frac{2.3}{P_e(R, SNR_0)} \cdot 10^{-4}$ and can be solved by any of the many existing convex optimization methods [54]. The solution $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{2^b-1})$, with all values rounded to the nearest integer, provides a good approximation to the optimal combination of IR bits to be sent so as to guarantee a probability of success above $1 - \gamma$ for all users.

2.7 Numerical Results

We now simulate the proposed methods and show numerical results to evaluate their performance. All simulations assume a bundle size of $b = 8$ codewords obtained from the QC-LDPC code of length $n = 648$ and $k = 432$ (rate 2/3) specified in [43]. Decoding and retransmission overhead costs are set to $c_D = 300$ and $c_R = 100$.

2.7.1 Single Link

This subsection simulates the method described in Section 2.4. As a reminder, the goal was to optimize the number and type of IR bits to be sent when there is a direct link between the transmitter and the receiver. Value iteration was applied to Eqs. (2.21) and (2.22) to yield a policy $A(f, SNR, R, SNR_{IR})$ specifying the number of extension bits (α) and bundle parity bits (β) to be requested as a function of the number of failed codewords remaining in the bundle f and their effective SNR. We restrict the range of α and β to be $[0, 216]$ and $[0, 648]$ respectively, so that the set of actions is finite. Fig. 2.5 shows a slice of the policy for


Policy α

Policy β

Figure 2.5. α and β decision for $R = 2/3$ and $SNR_{IR} = SNR$

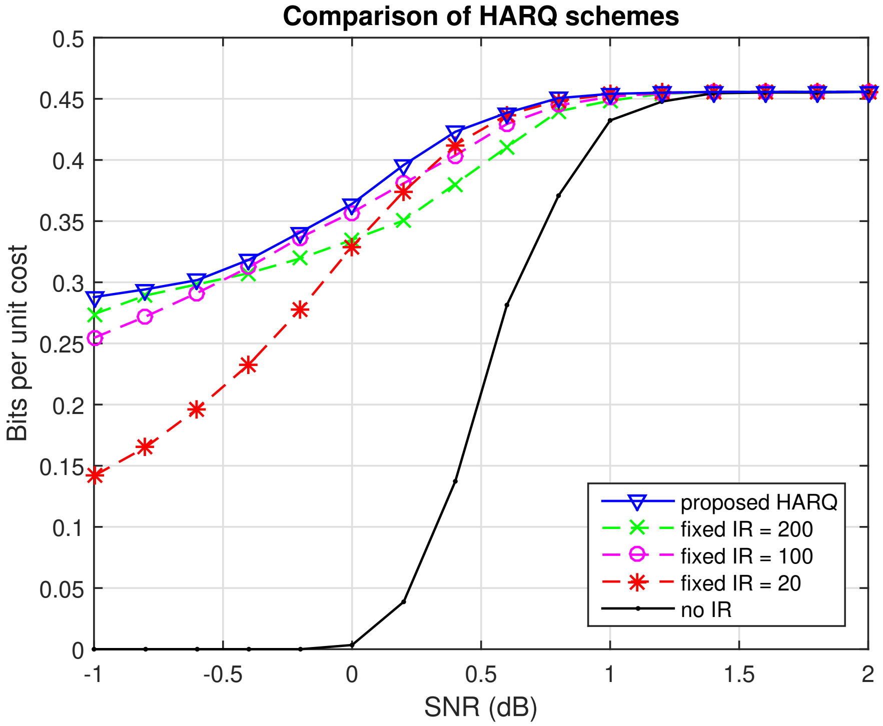


Figure 2.6. Throughput of different IR schemes for a single link.

code rate $R = 2/3$ and the IR having the same SNR as the original bundle, $(SNR_{IR} = SNR)$. It can be seen that the sum of α and β increases as the SNR decreases. This is because more IR bits are required to recover a highly corrupted bundle. In addition, our policy suggests that bundle parity bits are preferred over extension bits when there is a small number of failed codewords. This is worth noticing, since bundle parity is equivalent to Chase Combining when there is a single failure and extension bits generally offer better performance than Chase Combining [48]. However, the feedback limitations in our system prevent the receiver from conveying to the transmitter the specific codewords that failed; if extension bits were requested, the transmitter would have to send them for every codeword in the bundle, even for those that have already been successfully decoded. The policy illustrated in Fig. 2.5 has less than 16 possible combinations of (α, β), so it suffices to use 4 feedback bits to specify the request. This translates to only 1 bit of feedback per 2 codewords, which is half as much feedback as traditional fixed-length IR schemes with individual acknowledgements.

Fig. 2.6 compares the number of information bits delivered per unit cost for different IR schemes. Each point in the plot is the result of averaging Monte Carlo simulations for 1000 bundles and unlimited rounds of IR until success. If we interpret the cost as delay, then the number of information bits delivered divided by the cost will be the throughput. It can be observed that our HARQ policy provides modest gains over those with a fixed IR length, regardless of what this fixed number is and the SNR of the channel. These gains would be even larger in a scenario with variable SNR where, unlike fixed IR schemes, the proposed HARQ protocol would be able to adapt the IR length to each individual bundle.

2.7.2 Relay

This subsection simulates the method described in Section 2.5. As a reminder, a relay between the transmitter and receiver has to decide between AF and DF, using the same policies as in the single link scenario when failures occurred. The costs c_{DF} and c_{AF}, defined in Eqs. (2.23) and (2.26), are computed offline and compared to obtain the decision map. The relay estimates the SNR of both channels, finds the code rate of the received bundle, and looks up the decision map for whether or not to decode it. Fig. 2.7 shows the decision map for $R_1 = 2/3$. It can be observed that AF is preferred when both SNR_1 and SNR_2 are high enough, since the resulting SNR_{AF} is high and so AF removes the decoding cost at the relay, offsetting the small additional risk of decoding failure at the end user. Especially when $SNR_2 > 4.5$ dB, AF is the better choice regardless of SNR_1. The simulations also show that the AF region shifts to the right as the code rate R_1 increases. This makes sense because for higher code rates, the SNR must be increased correspondingly so that the risk of decoding failure is maintained at a low level for AF to prevail as discussed earlier.

Monte Carlo simulations also verify that the proposed relay HARQ strategy provides higher throughput than existing ones. Again, we could interpret the cost as delay, and so the information bits delivered per unit cost would measure the average throughput. The simulations first use an AWGN channel with deterministic gain to show that the relay decision map in Fig. 2.7 indeed chooses the forwarding scheme with a higher throughput. We then introduce stochastic channel gains to simulate a more practical scenario. Although the relay

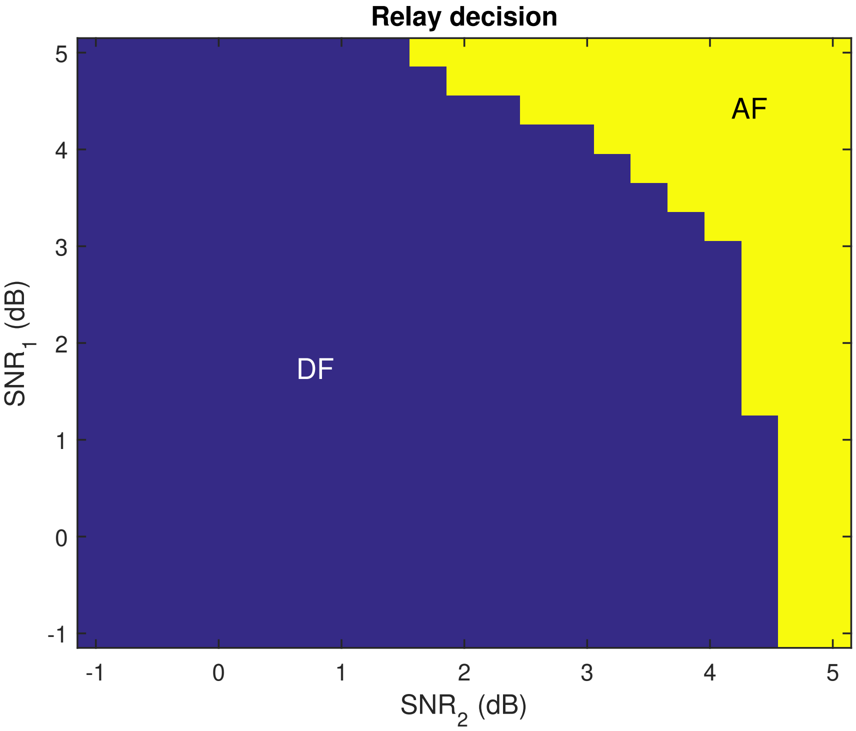


Figure 2.7. Relay decision map, shown for $R_1 = 2/3$.

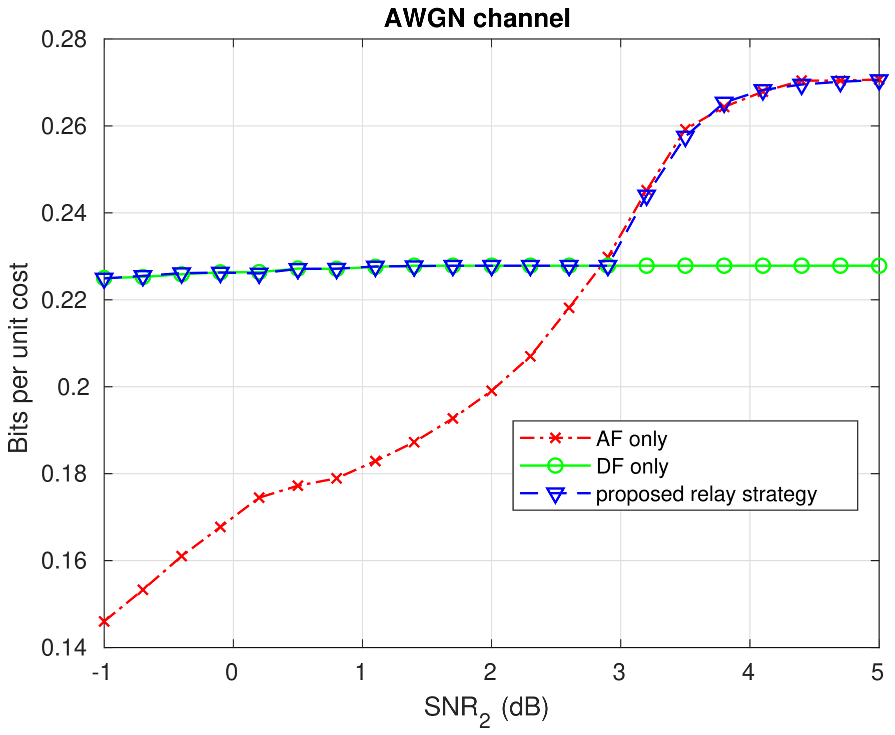


Figure 2.8. Average throughput of different relay strategies in AWGN channel.

decision engine was derived based on the assumption of AWGN channels, we show that the smart relay using our proposed policy based on the measured CSI (channel side information) is also suitable in this scenario and outperforms a fixed AF or DF relay.

In order to perform a fair comparison all relays use the same HARQ strategy described earlier when it comes to the single-link regime. Fig. 2.8 shows the average throughput using different relay strategies as a function of SNR_2, given a fixed $SNR_1 = 4$ dB and $R_1 = 2/3$. The relay decision map in Fig. 2.7 predicts that AF is the better choice if $SNR_2 > 3$ dB, and indeed we see in the figure that AF results in higher throughput than DF when $SNR_1 > 3$ dB. The smart relay is programmed to take the strategy with higher throughput.

Using the decision map should provide an advantage against channel variations because the relay can measure the SNR of its received signal and adopt the appropriate strategy accordingly, whereas a relay with fixed forwarding scheme will fail to adapt to the time-varying channel. The received signal is modeled as $y = gx + n$ where we assume unit transmit power ($E[x^2] = 1$) and additive Gaussian noise $n \sim \mathcal{N}(0, \sigma^2)$. The channel gain g

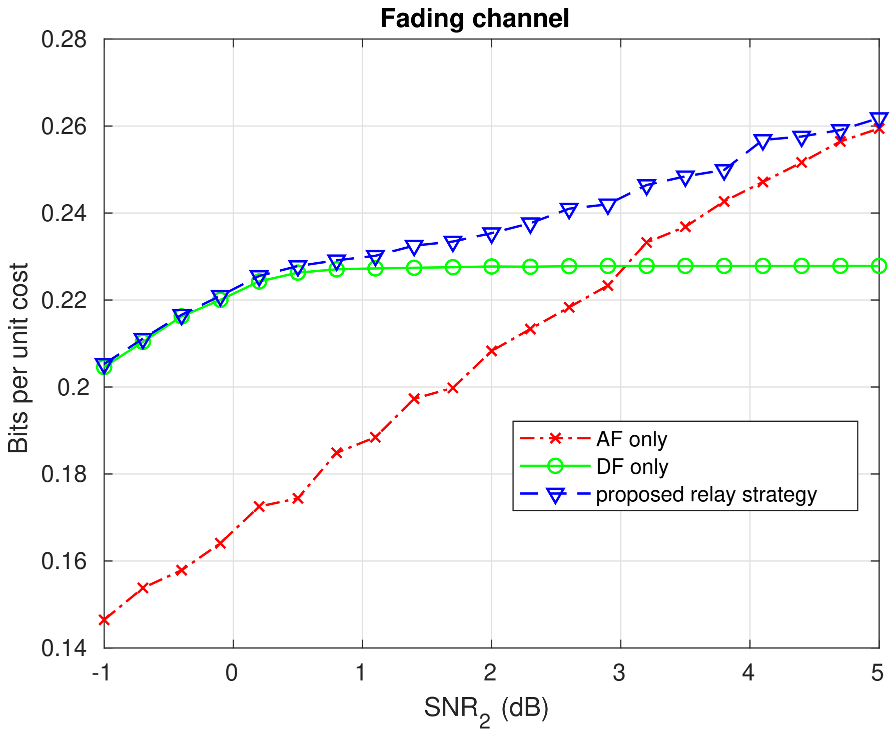


Figure 2.9. Average throughput of different relay strategies in fading channel.

is uniformly distributed over the range $[0.75, 1.25]$, remaining constant within each bundle but changing across different bundles and links. Fig. 2.9 shows the average throughput of the different relay strategies in the fading scenario as a function of SNR_2 for fixed $SNR_1 = 4$ dB and $R_1 = 2/3$. The smart relay exhibits a noticeable gain in throughput compared to AF or DF only relays. The gain is especially prominent in the region where AF and DF have similar performance, because our proposed hybrid relay strategy combines the advantages of both when neither of them significantly dominates the other.

2.7.3 Multi-user systems:

This subsection simulates the method described in Section 2.6. As a reminder, a single transmitter is delivering content to multiple receivers using a broadcast link. Each receiver is only interested in a subset of the codewords, but can overhear the others. Our goal is to minimize the number of IR bits to be broadcast in order to guarantee a certain probability of success for all users who suffered failures in decoding their desired information.

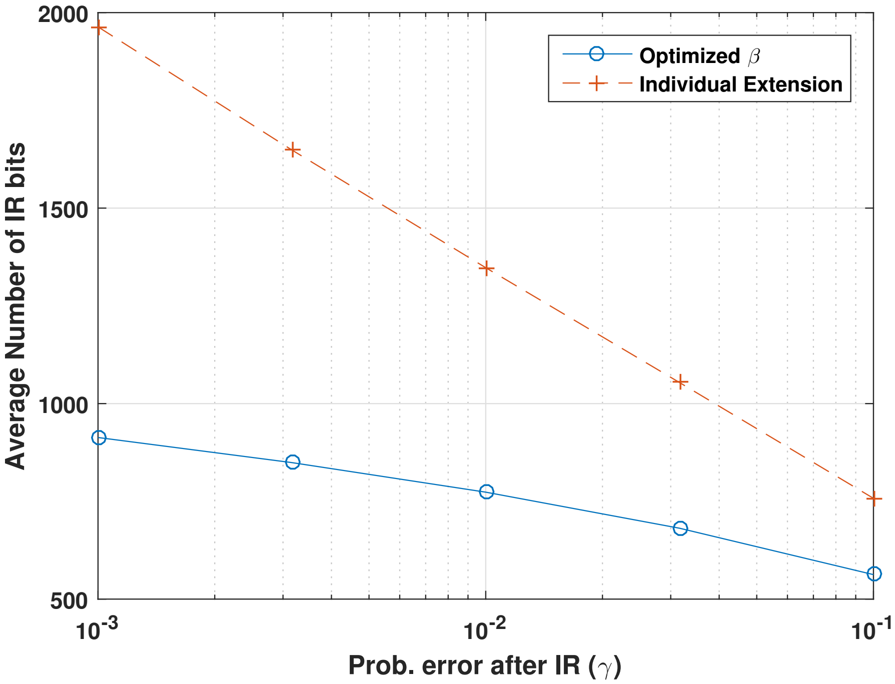


Figure 2.10. Average number of IR bits required to guarantee that the probability of decoding failure after IR is below γ for all users.

Fig. 2.10 compares the average number of incremental redundancy bits resulting from Eq. (2.36) with that required if we were to send extension bits for every codeword that failed decoding at its desired receiver. Each point in the plot is the result of 100 Monte Carlo simulations with rate $R = 0.5$ broadcast to 8 users experiencing random SNR uniformly distributed between -2 dB and -1 dB. According to Eq. (2.1), that yields a probability of decoding failure between 0.1 and 0.9 per codeword at each user. We used a logarithmic barrier method coupled with Newton descent to solve problem (2.36) and plotted the average $\|\boldsymbol{\beta}\|_1$ (number of IR bits) for different values of γ (probability of error after receiving the IR). Then, we used Eq. (2.1) to derive the number of extension bits that would be required to guarantee the same probability of error for all users. As it can be seen, our proposed method requires significantly fewer bits regardless of γ.

In most practical instances, the solution to problem (2.36) is not unique. There is a whole subspace of optimal values for $\boldsymbol{\beta}$. In order to obtain a sparse solution, we introduced a small random perturbation in the objective, minimizing $\sum_{k=0}^{2^b-1}(1+\epsilon_k)\beta_k$ instead of $\sum_{k=0}^{2^b-1}\beta_k$, where ϵ_k are random noise variables distributed between 0 and 10^{-2}. The result was that, in most cases, the number of non-zero entries in $\boldsymbol{\beta}$ was lower than the number of failed codewords. This means that, on top of requiring fewer IR bits, our method is also able to group them into fewer types than a pure extension approach, reducing the amount of overhead.

2.8 Conclusion

This chapter addresses the problem of error correction in single link, relay, and broadcast systems. Specifically, it proposes techniques for optimizing the incremental redundancy (IR) bits sent by an HARQ protocol under the assumption that the feedback channel can only support a few bits of feedback per bundle of codewords (or packets). Apart from the traditional extension IR bits, consisting of a few additional bits for each codeword, this chapter also considers bundle IR, consisting of encoded IR bits which the receiver can use to refine the LLRs in multiple codewords.

The allocation of IR bits in a single link is modelled as a Markov Decision Process seeking to minimize a pre-determined cost function. The chapter describes how the problem should be formulated and solved, resulting in a set of policies parameterized by the number of failures per codeword bundle, effective SNR of the received codewords, and coding rate. It then extends this single link framework to a relay scenario, where an intermediate node has to decide whether to decode (DF) or just amplify (AF) incoming bundles before forwarding them on. Finally, the chapter studies a multiuser scenario where a single source broadcasts information to multiple receivers with different interests. It proposes transmitting encoded IR bits that benefit multiple receivers and formulates a convex problem to optimize their number and encoding.

Numerical simulations show that the proposed methods provide a modest increase in throughput compared to traditional HARQ schemes with fixed-length codeword extension. The proposed policy for the relay outperforms fixed forwarding strategies and the proposed strategy for broadcast systems significantly reduces the total number of IR bits needed to guarantee a given probability of success, compared to sending individual extension bits for each codeword. The increased flexibility in requesting different numbers and types of IR bits and the ability to make decisions based on the measurement of the received signals display significant advantages in limited feedback communication systems.

3. INCREASING THROUGHPUT IN WIRELESS COMMUNICATIONS BY GROUPING SIMILAR QUALITY BITS

This chapter proposes and studies a technique for grouping the bits transmitted through a wireless channel into codewords according to their quality (SNR). It proves that splitting the bits into multiple codewords of different rates provides a higher throughput than mixing heterogeneous quality bits into fixed-rate codewords. The chapter first analyzes the pros and cons of different mappings of bits and codewords to the available time, frequency, and modulation resources. Then it describes the proposed scheme for a 16-QAM modulation and illustrates its benefits through simulations. Finally, it provides a mathematical proof of its superiority in a binary-input parallel AWGN channel with finite length error correcting codes (ECC). The proposed scheme can be applied to any communications channel using ECC, but it is of particular interest for millimeter-wave (mmWave) wireless communications, where the channel quality is closely monitored and high order modulations are used over wide bandwidths. The simulations suggest that modest gains in throughput can be obtained with negligible additional complexity.

3.1 Introduction

Typical wireless systems subdivide their available time and frequency bands into smaller resource blocks that can be independently processed at the receiver. Within each of these resource blocks, the transmitter can compensate for link gain fluctuations by periodically adjusting the size of the modulation and the rate of the error correcting codes (ECC), attempting to ensure a somewhat uniform probability of failure at the receiver. Several other factors, such as latency, reliability and decoder complexity are also taken into account but, ultimately, the overall goal is to maximize throughput, understood as the number of information bits correctly received per second.

The Shannon coding theorem states that if the blocklength of the channel code goes to infinity, it is possible for the throughput to match the channel capacity. However, [42] has shown a much tighter upper bound in the finite blocklength regime by introducing the channel dispersion parameter V. Specifically, the maximum cardinality $M^*(n, \epsilon)$ of a blocklength n codebook that can be decoded with block error rate (BLER) at most ϵ satisfies

$$\log M^*(n, \epsilon) = nC - \sqrt{nV} Q^{-1}(\epsilon) + O(\log n), \tag{3.1}$$

where C and V are the Shannon capacity and the dispersion of the channel, respectively.

This chapter will propose a technique for mapping information bits into finite length codewords and those codewords into resource blocks with the goal of maximizing throughput. The proposed technique can be applied to any communications channel with heterogeneous bit qualities but, for the sake of simplicity, we will assume a wireless system using QAM modulations and an AWGN channel [55]. Our simulations will employ the binary Low Density Parity Check (LDPC) codes of length $n = 648$ proposed in the 3GPP standard for 802.11n [43]. Binary LDPC codes are the most prevalent ECC in modern systems and have been included in the mmWave standard [56].

In a binary LDPC decoder, received channel values are processed to obtain a log-likelihood ratio (LLR) for each individual bit b_i as $\ell_i = \log \frac{p_i(0|\mathbf{y}_i)}{p_i(1|\mathbf{y}_i)}$, where $p_i(0|\mathbf{y}_i)$ and $p_i(1|\mathbf{y}_i)$ respectively represent the probability of b_i being 0 or 1, given the corresponding received value $\mathbf{y}_i$. These LLRs are then progressively refined by iterative message passing.

Summarizing, this chapter explores multiple practical aspects of modulation and coding that may be relevant in the research, design, and standardization of mmWave for fifth generation (5G) and beyond networks. The chapter will be organized as follows. Section 3.2 studies different mappings of information bits to modulation symbols, and codewords to time-frequency and constellation resources. Simulations show practical benefits from our proposed bit mapping strategies in a 16-QAM modulation and suggest that interleaving can actually increase the decoding failure rate if not designed carefully. Section 3.3 studies the achievable channel coding rate of a parallel AWGN channel for finite blocklength codes. It provides a mathematical proof that grouping the bits into codewords with homogeneous SNR allows for a higher data rate than mixing them into the same codeword. Section 3.4 concludes the chapter.

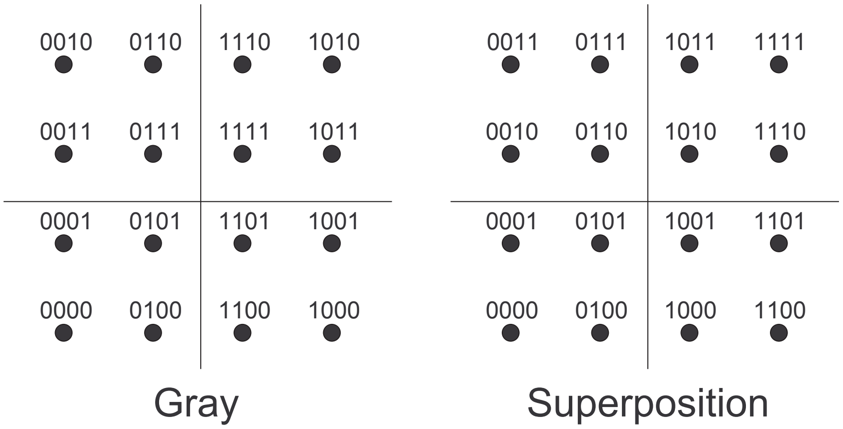


Figure 3.1. Mapping of bits to 16-QAM constellation symbols.

3.2 Codeword Mapping and Modulation

To the best of our knowledge, there does not exist a commonly agreed standard for the modulation and frame structure of mmWave communications [57]. Additionally, the number of codewords transmitted in each frame will depend on the order of the modulation, code length, etc. Information bits are encoded and then mapped into modulation symbols to be transmitted using the time and frequency resource blocks. Hence, a modulation needs to specify both how to map the bits into symbols and how to distribute the codewords across the resource blocks.

3.2.1 Mapping bits to modulation symbols

Modulations with multiple bits per symbol can use different mappings, as shown in Fig. 3.1 for 16-QAM. Gray mapping offers the lowest bit error rate (BER) since adjacent constellation symbols differ in a single bit, but it is not linear. Superposition mapping can be understood as the algebraic sum of independent BPSK modulations with different power. The transmitted symbols can therefore be constructed by scaling and adding multiple independent streams of binary data, which facilitates its implementation in multi-user scenarios [58].

Another factor to consider in the mapping of bits to modulation symbols is that when there are more than two bits per symbol, not all the bits have the same error probability [55]. Regardless of whether Gray or superposition labeling is used, the second and fourth bit in Fig. 3.1 are more prone to error than the first and third. Furthermore, the marginal distributions of the noise experienced by each bit are highly correlated.

Ideally, the system should use a non-binary LDPC code with a decoder capable of accounting for these asymetries and correlations. Non-binary LDPC codes significantly outperform their binary counterparts in terms of failure rate, but their increased decoding complexity is a problem for the high speeds expected in mmWave communications [59]. Therefore, most practical systems use binary LDPC codes and operate as if each bit suffered independent Gaussian noise. The following subsection will propose a refinement to this approach.

3.2.2 Mapping codewords to resource blocks

There are multiple ways in which the LDPC codewords can be mapped to the resource blocks. As depicted in Fig. 3.2, codewords can be assigned different time slots, frequency subcarriers, or bits in a modulation symbol. Each codeword mapping approach has its own advantages and drawbacks[1]:

Split across time

Codewords are transmitted sequentially in time, using all available subcarriers. This mapping minimizes latency, since codewords can start being decoded as soon as they are received. Additionally, full-duplex receivers are able to report decoding failures immediately, allowing the transmitter to adjust the modulation size or code rate for subsequent codewords. On the negative side, if the subcarriers use different modulations or there is frequency selective fading, the reliability of the bits in a codeword could vary widely. Furthermore, it is

[1] ↑In practice, bits from different codewords are often interleaved to break error bursts. Still, each codeword can either be distributed across many subcarriers during a short time, or over one subcarrier for a longer period, etc.

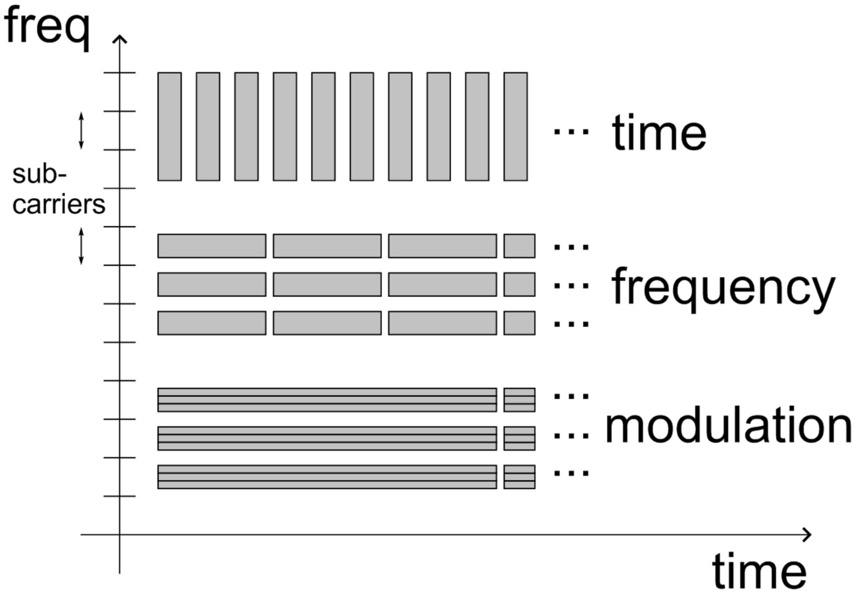


Figure 3.2. Mapping of codewords to resources.

very hard for binary decoders to exploit the correlation in the noise suffered by the bits in each modulation symbol.

Split across frequency

Codewords are assigned to different subcarriers, using as many time slots as necessary. Frequency selective fading can be addressed by adjusting the modulation and coding rate within each codeword, and it is easy to multiplex users in frequency by assigning them different subcarriers. Unfortunately, latency and memory requirements would be worse than in the previous case. The receiver needs to wait until the first batch of codewords has been received before starting to decode them all in parallel. The noise correlation between bits in the same modulation symbol is still hard to exploit, but there exist multi-edge type decoders that can do it if the modulation remains constant throughout the codeword [60].

Split across modulation

For modulations with more than one bit per symbol, have each bit correspond to a different codeword [44]. This mapping increases the number of subcarriers and/or timeslots required to transmit each codeword, thereby increasing latency, but it simplifies the exploitation of the correlation and asymmetries between the different bits in a symbol. Specifically, when one codeword is successfully decoded, the known bits can be used to refine the LLRs from the other bits in the same symbols, helping decode the remaining codewords. Fig. 3.3 shows how the distribution of LLR values for the first bit in a Gray-coded 16-QAM constellation changes when the second bit is known.

Bits with higher probability of error should be encoded at a lower rate than those with lower BER. The maximal coding rate achievable with finite blocklength decreases with the channel dispersion (variance of the information density) [42]. Grouping the bits into codewords according to their reliability will create several parallel channels with smaller dispersion than the original channel, thereby increasing the overall finite length capacity. We will present a proof of this result in a relaxed scenario in section 3.3.

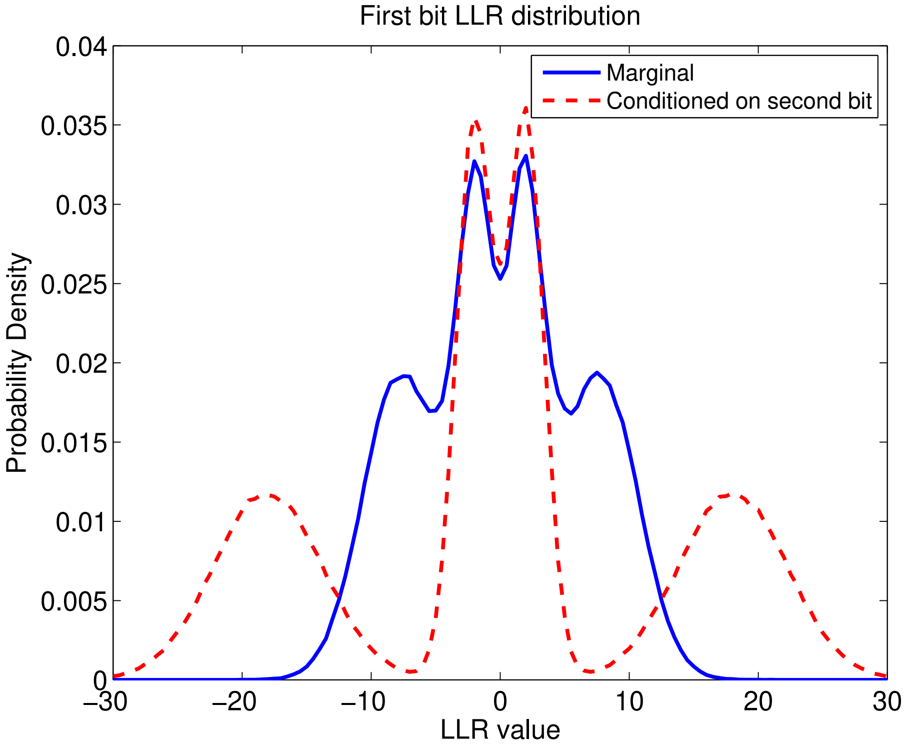


Figure 3.3. Pdf of the LLR values for the first bit in a 16-QAM constellation using Gray mapping (see Fig. 3.1) and SNR = 10 dB.

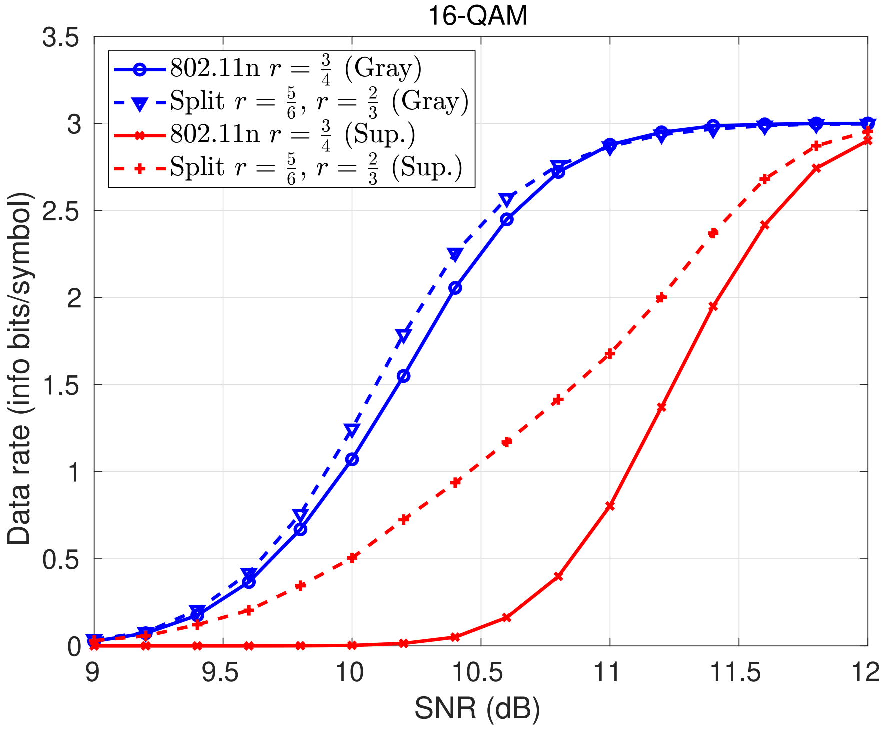


Figure 3.4. Average throughput per 16-QAM symbol. Solid curves are encoding all bits with an LDPC code of rate $\frac{3}{4}$ and dashed curves are encoding the most reliable (first and third) bits with rate $\frac{5}{6}$, the least reliable (second and fourth) with rate $\frac{2}{3}$, and exploiting the correlation between codewords.

3.2.3 Simulations

The splitting across time or frequency is a relatively standard concept but, to the extent of our knowledge, the idea of assigning the bits in a modulation symbol to codewords of different rates is new. Other attempts have been made at using bit mapping to improve the overall decoding performance [61], but they are significantly more complex to implement. One of the main advantages of our method is that it does not require any knowledge about the channel or decoder because the asymmetry in bit qualities is solely due to the QAM constellation. Fig. 3.4 compares the throughput achieved with a 16-QAM modulation when all the bits in each symbol are part of the same codeword (solid curves) with that achieved when they are grouped into codewords of different rates according to their BER and the correlation between the codewords can be exploited (dashed curves). It can be observed that the latter increases the rate for both Gray and superposition bit mappings.

An interesting observation that arises from Fig. 3.4 is that interleaving can be detrimental to the data rate if it is done across symbols with different reliability. However, interleaving is necessary in practice to break error bursts and smooth channel variations. Rather than avoiding it completely, we propose having multiple parallel encoder-interleaver chains producing the different bits in a modulation symbol, as shown in Fig. 3.5. Bits prone to higher probability of error would be interleaved among themselves and encoded with lower rates than those subject to smaller probabilities of error. Interleaving is done across codewords of the same rate.

3.3 Coding Rate of Mixed SNR Channels

We have not been able to derive a mathematical characterization for the gain obtained by splitting the bits in a 16-QAM constellation, mainly because of the difficulty in deriving the finite blocklength capacity of such a system. However, this section provides such characterization for a simplified channel.

We consider a binary-input parallel AWGN channel where a proportion $\alpha \in [0, 1]$ of the bits experience SNR of γ_1, and the remaining $\bar{\alpha} := 1 - \alpha$ experience SNR of γ_2. The transmitter knows which bits will experience a higher or lower SNR, and can therefore decide

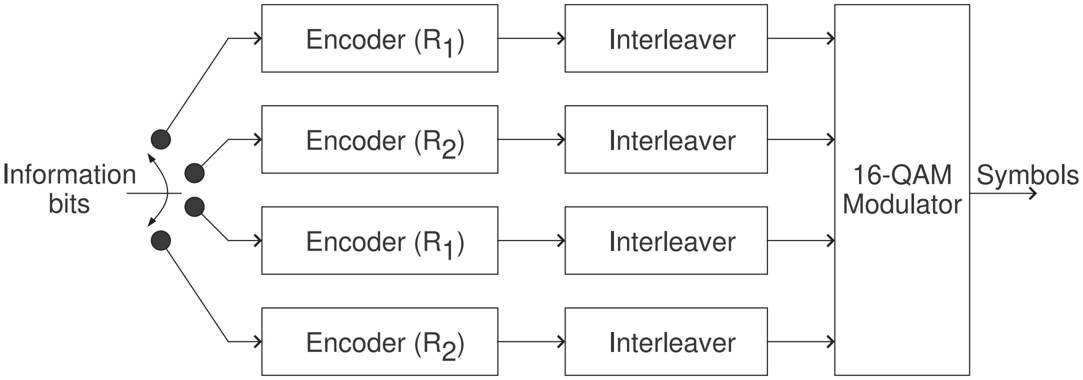


Figure 3.5. Mapping codewords to bits in a modulation symbol.

to mix them in the same codeword or group them into codewords with homogeneous SNR. This assumption arises naturally in a 16-QAM channel where some bits are known to suffer higher BER than others; here we relax the model to BPSK and parallel AWGN. We will prove that grouping the bits into codewords according to their SNR allows for a higher channel coding rate with finite length codewords.

Theorem 3.3.1. *When seeking to guarantee BLER $\leq \epsilon$ with finite blocklength codes, grouping the bits into SNR-homogeneous codewords will achieve a higher channel coding rate than mixing bits of different SNR in the same codeword.*

Proof. With bit grouping, we effectively have two independent AWGN channels with different SNR. If we have L codewords of length n bits, αL codewords will experience SNR γ_1, and $\bar{\alpha} L$ codewords will experience γ_2. (L can be chosen large enough such that these are integers.) According to Eq. (4.6), the maximum feasible number of information bits subject to block error rate below ϵ can be expressed as

$$\begin{aligned} k^*_{\text{split}} = \alpha L \left(nC_1(\gamma_1) - \sqrt{nV_1(\gamma_1)} Q^{-1}(\epsilon) + \rho_n \right) \\ + \bar{\alpha} L \left(nC_1(\gamma_2) - \sqrt{nV_1(\gamma_2)} Q^{-1}(\epsilon) + \rho_n \right) \end{aligned} \tag{3.2}$$

where $C_1(\gamma) = \frac{1}{2}\log(1+\gamma)$ and $V_1(\gamma) = \frac{\gamma(\gamma+2)}{2(\gamma+1)^2}$ are the channel capacity and dispersion for an AWGN channel [62]; $\rho_n = O(\log n)$ is the error term.

Without bit grouping, we effectively have an L-parallel Gaussian channel with mixed SNRs. In each codeword, αn bits experience SNR γ_1 and the other $\bar{\alpha} n$ bits experience γ_2. Theorem 4 in [62] shows the finite blocklength capacity and dispersion of L-parallel Gaussian channels, and we use that result to express the maximum feasible number of information bits with L codewords as

$$\begin{aligned} k^*_{\text{mixed}} &= L\Big(\frac{n}{L} C_L(\gamma_1, \gamma_2) - \sqrt{\frac{n}{L} V_L(\gamma_1, \gamma_2)} Q^{-1}(\epsilon) + \rho_n \Big) \\ &= nL \left(\alpha C_1(\gamma_1) + \bar{\alpha} C_1(\gamma_2) \right) \\ &\quad - \sqrt{n} L \sqrt{\alpha V_1(\gamma_1) + \bar{\alpha} V_1(\gamma_2)} Q^{-1}(\epsilon) + L\rho_n \end{aligned} \tag{3.3}$$

where $C_L(\gamma_1, \gamma_2)$ and $V_L(\gamma_1, \gamma_2)$ are the capacity and dispersion for the L-parallel AWGN channel comprising αL sub-channels with SNR γ_1 and $\bar{\alpha} L$ sub-channels with SNR γ_2.

If we consider a reasonable practical code length where the $\rho_n = O(\log n)$ term is negligible, we see that the gain in coding rate due to bit grouping can be expressed as

$$\begin{aligned} \Delta R &= \frac{k^*_{\text{split}} - k^*_{\text{mixed}}}{nL} \qquad (3.4) \\ &= \frac{Q^{-1}(\epsilon)}{\sqrt{n}} \Big[\sqrt{\alpha V_1(\gamma_1) + \bar{\alpha} V_1(\gamma_2)} \\ &\quad - \left(\alpha \sqrt{V_1(\gamma_1)} + \bar{\alpha} \sqrt{V_1(\gamma_2)} \right) \Big]. \qquad (3.5) \end{aligned}$$

Eq. (3.5) is always non-negative, since

$$\begin{aligned} &\alpha V_1(\gamma_1) + \bar{\alpha} V_1(\gamma_2) - \left(\alpha \sqrt{V_1(\gamma_1)} + \bar{\alpha} \sqrt{V_1(\gamma_2)} \right)^2 \\ &= \alpha(1-\alpha) \left(\sqrt{V_1(\gamma_1)} - \sqrt{V_1(\gamma_2)} \right)^2 \geq 0, \qquad (3.6) \end{aligned}$$

and this completes our proof that $\Delta R \geq 0$. □

We now make a few observations. Equality holds in Eq. (3.6) when $\gamma_1 = \gamma_2$ or $\alpha = 0, 1$. In other words, splitting the bits will always achieve a higher information rate than not splitting them for finite length codes, unless all bits experience the same SNR. It can also be seen that the rate gain ΔR increases as $\alpha \to 1/2$ for fixed SNRs and as the BLER constraint tightens ($\epsilon \to 0$). However, it decreases with blocklength, since both rates converge to the Shannon limit when $n \to \infty$.

We now support the previous result with numerical simulations. We consider the 802.11n LDPC codes of length $n = 648$ [43] sent over a mixed SNR channel and find the average throughput, defined as the total number of information bits in successful frames divided by the total number of bits sent. The mixed SNR channel has two fixed SNRs of $\gamma_1 = 0.8$ dB and $\gamma_2 = 3.0$ dB. We use $L = 12$ and simulate the channel with $\alpha = i/12$ for $i = 0, 1, \cdots, 12$. Practical LDPC decoders are often unable to characterize the SNR of each specific bit. Instead, they observe the variance of the noise across the whole codeword (or over a training sequence) and use it to estimate the SNR required to generate the LLRs.

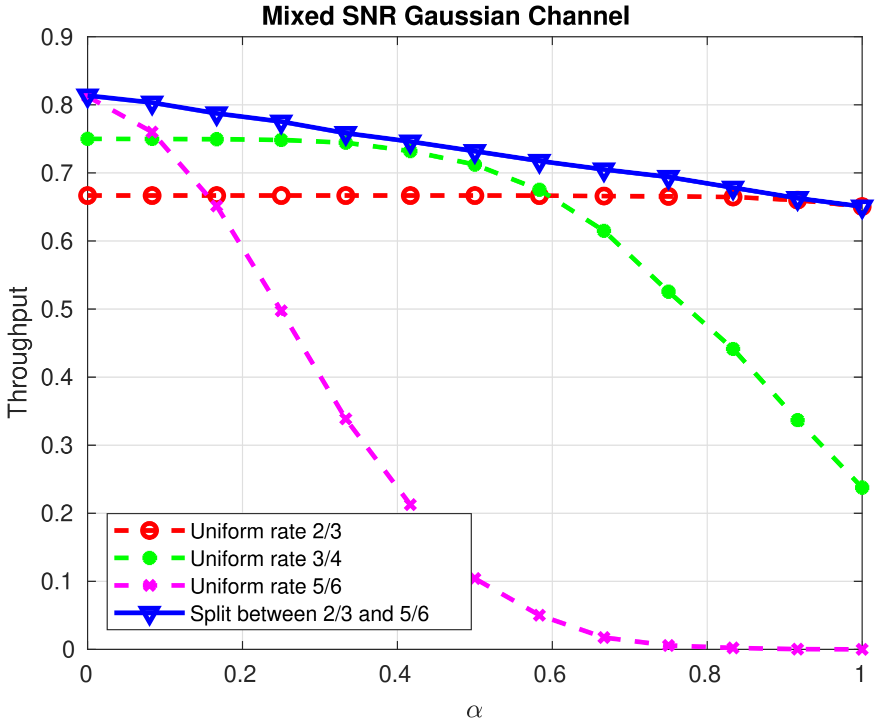


Figure 3.6. Throughput with and without bit splitting, as a function of α, i.e. the fraction of bits having γ_1. SNRs are fixed at $\gamma_1 = 0.8$ dB and $\gamma_2 = 3.0$ dB.

When the transmitter groups the bits into different codewords according to their anticipated qualities, the receiver may obtain a different noise variance estimate for each codeword, and such estimate will be used to compute the LLRs in our simulation.

We simulated 6000 random codewords to find the average throughput for each scenario, and used the MATLAB communications toolbox LDPC decoder with maximum of 40 iterations. The dashed curves in Fig. 3.6 show the throughput of LDPC codes with different rates when a fraction α of the bits in the codeword have SNR 0.8 dB and $\bar{\alpha}$ have 3.0 dB. The solid curve is obtained by sending a fraction α of the codewords with rate 2/3 over an AWGN channel with 0.8 dB, and the other $\bar{\alpha}$ with rate 5/6 over 3.0 dB. This captures the effective behavior of splitting and grouping the bits into homogeneous codewords as described earlier. It can be seen that the bit splitting technique outperforms all fixed rate codes, regardless of the proportion of good and bad bits in the codewords.

3.4 Conclusion

This chapter studied some of the modulation and coding trade-offs arising in mmWave systems due to fast channel variations. It analyzed how the mapping of bits and codewords to time, frequency, and modulation resources can affect the average latency and throughput in the network. Then it showed that the overall data rate can be improved by grouping bits into codewords according to their reliability and exploiting correlation between codewords when decoding. Finally, it proved that such grouping increases the maximum achievable channel coding rate for finite blocklength codes in a parallel AWGN channel. The gain was found to be most significant when the channel experiences high SNR variations. The proposed techniques are suitable in a wide range of channel conditions for future mmWave systems.

4. A ROUTING AND RESOURCE ALLOCATION ALGORITHM FOR COLLABORATIVE NETWORKS USING INFERRED INFORMATION

Even with the advent of 5G wireless communications and the millimeter wave spectrum, there will always be crowded frequency bands where multiple uncoordinated networks will have to contend (or collaborate) to squeeze as much throughput as possible while avoiding interference. This work presents a heuristic algorithm that individual networks can follow to collaboratively share the available bandwidth and route the traffic within their network, with the objective of maximizing the overall sum rate over all networks. The algorithm finds a greedy solution by independently optimizing the links and routes for each network, and then refines that solution by discarding inefficient and potentially harmful links. Simulation results show that, when the networks have different sizes and traffic loads, the proposed algorithm outperforms the original greedy solution as well as the optimal configuration with a uniform distribution of resources.

4.1 Introduction

The number of wireless devices is increasing exponentially and expected to continue doing so in the foreseeable future. Furthermore, users and applications demand more data at faster speeds, lower latencies, and higher reliability. Some researchers and telecommunications providers have turned towards the mmWave band (above 6 GHz) to find the necessary bandwidth to accommodate such growing demand, but the hardware and propagation limitations associated with high frequencies make mmWave communications unsuitable for many applications. The congestion in the sub-6GHz band is not expected to be alleviated anytime soon and it is therefore imperative that we find ways for different networks to coexist sharing the same spectrum.

Prior research has addressed this problem using either shared databases (e.g., TV whitespace [63] and CBRS [64]) or sensing-based techniques (e.g., DARPA XG [65]), but these approaches substantially underperform information theoretic limits. Information theory tells us that nearly any performance measure of multiple networks operating in a shared environment is bounded by the fully centralized case, where all nodes have infinite capacity links to a shared controller. Centralized performance is clearly unachievable in practice, but allowing some level of collaboration or information sharing could help us approach that bound.

The goal of this chapter is to design a protocol for autonomous routing and spectrum management across multiple wireless networks operating in a shared interference-limited environment. Each network will have to assign a carrier frequency and bandwidth to each of its links, as well as adapt their transmit power and the path that packets will take from their source to their destination. This will be done by studying the optimal (centralized) solutions for different random configurations and attempting to leverage their common traits in the design of our protocol.

We will use the information-theoretical capacity as a performance metric, ignoring latency, security, and other practical considerations. The concept of capacity in wireless networks was pioneered by Gupta and Kumar in [66]. Their paper focused on asymptotic bounds for the number of successful links that can be simultaneously scheduled under two interference models: the physical model, where transmissions are successful when the SINR is above a pre-fixed thresthold, and the protocol model, where they are successful as long as there are no other transmitters within a certain distance of the receiver. Subsequent works adopted the same models and extended the results to consider probabilistic success in the transmissions [67] and heuristics with constant approximation guarantees to the previous bounds [68]–[70]. However, they still focused on cases with asymptotically large number of nodes and offered loads. This chapter attempts to solve the problem for a pre-fixed set of flows with specific sources, sinks, and maximum data rates.

In environments where background noise is significantly stronger than the potential interference between networks (noise-limited), the capacity maximization problem becomes trivial: every node should transmit with as much power as it can using as much bandwidth as it can. However, when the background noise is relatively small (interference-limited) the problem of designing a strategy to maximize the information theoretical capacity of a network is very hard, mostly because the optimal strategy is different depending on the noise power and the positions of the nodes (equivalently, the channel gains). Consider the highly simplified scenario of four nodes located in the corners of a $D \times D$ square, with quadratic signal attenuation and additive white Gaussian noise. Assuming that the two nodes on the left want to transmit to the two nodes on the right, the question arises: would it be better to have both nodes transmit simultaneously or to alternate their transmissions? The answer depends on the distance between the nodes and how they are paired. If the links go along the sides of the square, the overall capacity for each strategy is given by

$$C_{\text{sim}} = 2\log\left(1 + \frac{2P}{N_0 + P}\right) \tag{4.1}$$

$$C_{\text{alt}} = \log\left(1 + \frac{2P}{N_0}\right), \tag{4.2}$$

where $N_0/2$ is the noise power and $P = P_T/D^2$ denotes the received signal power, given by the transmit power divided by the distance squared. When $P = N_0$ we have that $C_{\text{sim}} > C_{\text{alt}}$, but when P becomes large enough the inequality inverts. Similarly, if the nodes are attempting to transmit along the diagonals of the square, the overall capacities for the two strategies are given by

$$C_{\text{sim}} = 2\log\left(1 + \frac{P}{N_0 + 2P}\right) \tag{4.3}$$

$$C_{\text{alt}} = \log\left(1 + \frac{P}{N_0}\right). \tag{4.4}$$

When $P = N_0$ we have the opposite conclusion as in the previous case: $C_{\text{sim}} < C_{\text{alt}}$, but the inequality once again inverts if the received power becomes small enough. Finding the optimal strategy requires carefully analyzing the received, interference, and noise powers, even in a simplified scenario such as this one. In a more realistic situation with a larger number of nodes in arbitrarily locations the problem becomes exponentially more complex.

Apart from assigning powers and scheduling transmissions, it is well known that the simultaneous routing and resource allocation problem is NP-hard [71]. Over the last decade there have been multiple attempts at finding efficient methods to solve it, mostly through decomposition. Palomar et al. published a compilation of the first results in this direction [72] and analyzed the advantages and disadvantages for each of them. However, there has been significant progress in this area since that compilation was published. Iterative approaches such as [73]–[75] have been shown to provide better results than methods purely based on decomposition. More recently, there have been multiple methods proposed based on fractional programming techniques [76]–[78], information-theoretical inequalities for the optimality of treating interference as noise [79], [80], and machine learning [81]–[84]. Heuristic methods based on the distance between the links have also been shown to provide good results [85].

In many applications, it is common for resources to be sliced (in time, in frequency, or both) and auctioned among multiple networks for their exclusive use. Some researchers have studied this scenario and attempted to optimize the bidding process [86]. Another common approach in modern software defined radio (SDR) -based networks is to map multiple virtual channels to a few shared physical ones [87]. Yang et. al. proposed using a dynamic program to optimize such mapping [88], allowing reuse of the physical channels when the collision probability is below a pre-fixed threshold.

However, all of these recent methods are aimed at solving the sum-rate maximization problem (or some variation of it) for a single network and frequency band. They seek the set of links that should be scheduled simultaneously to maximize a certain utility function, assuming that the nodes are somewhat coordinated and it is possible to sacrifice all transmissions from some nodes in favor of those from others. This chapter, on the other hand, attempts to solve the problem for multiple networks and frequency bands, for which there is very limited literature. It will assume that the networks operate independently without a standardized physical layer to enable the decoding and relaying of each other's messages as proposed in [1], [89]. It is therefore unclear how any of them would know to stop its transmissions for the benefit of others. We will propose a scheme that enables collaboration using only the information that the networks can infer from the power of overheard transmissions.

Notation: We use bold font to represent vectors (lower case) and matrices (upper case). The bold numbers $\mathbf{0}$ and $\mathbf{1}$ represent the all zeros and the all ones vectors, respectively, with dimensions that should be clear from the context. We use parentheses to construct column vectors from comma separated lists, i.e., $\boldsymbol{x} = (x_1, \dots, x_n) = [\ x_1 \ \cdots \ x_n\ ]^T$. The gradient of a scalar function respect to a vector is given by $\nabla_{\boldsymbol{x}} f = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}\right)$. The comparison operators ($\leq$, $\geq$, min, max) on vectors or matrices are understood as elementwise.

4.2 System Model

The existing literature typically divides time into discrete time slots, and then optimizes the transmission schedule of the network in each time slot. This is suitable for traditional radios where the Tx (transmitter) and Rx (receiver) signal processing chains are implemented in hardware, offering limited flexibility. A controller may switch the transmitter on or off and control the power in each time slot to achieve the desired schedule. However, with the recent advancement in SDRs, more dimensions can now be explored. An SDR transmitter can easily switch its frequency on demand, and cater to complicated signal processing requirements. We leverage this flexibility of SDRs and divide the available bandwidth into m identical bands or channels instead of dividing time. Theoretically, these are equivalent approaches, but dividing frequency instead of time offers some practical advantages. For instance, the nodes do not need to synchronize the time slots and multi-hop packets can be delivered to their destinations with less latency because the connecting hops can happen concurrently on different frequency channels, whereas in the time-division approach one hop may need to wait until the next cycle after the previous hop.

We consider N wireless networks deployed over a rectangular 2-dimensional region, with the i-th network consisting of n_i nodes distributed uniformly at random over the whole region. Each node is equipped with a single-antenna full-duplex transceiver, i.e., a node can be transmitting on one frequency band while it is receiving on another. Although some modern radios are capable of transmitting and receiving on the same frequency, this chapter does not address that scenario. To simplify our discussion, we will assume that every network has an even number of nodes and there are a total of $2n = \sum_{i=1}^{N} n_i$ nodes. This will allow grouping all the nodes into transmitter-receiver pairs on each frequency band, although not all those links will be necessarily active. If the number of nodes were odd, one node would be left out of the Tx-Rx pairing for each frequency band, but this should not have any impact in the overall performance since most of the chosen transmitters will need to stay inactive regardless to avoid excessive interference. For simplicity, we will assume that nodes that transmit do so continuously in time, on whatever band they are assigned.

The signal received by node i at time t and channel f, assuming it is not transmitting on that channel, is given by

$$y_i^f(t) = \sum_{j=1}^{2n} \sqrt{g_{ji}} x_j^f(t) + z_i^f(t), \tag{4.5}$$

where $x_j^f(t)$ is the signal transmitted by node j on channel f at time t (if any), g_{ji} the (power) path loss from node j to node i, and $z_i^f(t)$ the additive Gaussian noise at receiver i. For simplicity, we assume that the noise is white with the same variance σ^2 for all receiver nodes, that the path loss is identical for all the frequencies under consideration, and we neglect propagation delays. These assumptions could be easily removed, but it would complicate the equations unnecessarily.

Since there are $2n$ nodes and m frequency bands, there will be $L = nm$ potentially active links for any given set of pairings. These links are modeled as symmetric AWGN channels with attenuation proportional to the distance squared. Recent research has shown that, although simplistic, quadratic loss channel models are a sufficiently good approximation in many scenarios and enough for machine learning models to provide good results [84].

The capacity of the ℓ-th link ($\ell = 1, \dots, L$) is given by

$$c_\ell = \log(1 + \gamma_\ell), \tag{4.6}$$

where the unit of data rate is normalized by the bandwidth of each frequency channel and γ_ℓ denotes the SINR (signal-to-interference-and-noise ratio) of the link. Specifically, if the ℓ-th link goes from node ℓ_o to node ℓ_d,

$$\gamma_\ell = \frac{g_{\ell_o \ell_d} p_\ell}{\sum_{\nu \in \mathcal{T}_\ell \setminus \{\ell\}} g_{\nu_o \ell_d} p_\nu + \rho_{\ell_d}}, \tag{4.7}$$

where $\mathcal{T}_\ell$ denotes the set of all links on the same frequency band as link ℓ, ν_o denotes the transmitter node for link ν, p_ν its transmit power, and $\rho_{\ell_d} = \sigma^2$ the noise level at ℓ_d. We assume unit maximum power, so $0 \leq p_\ell \leq 1$ for all ℓ.

Each network is being asked to deliver a certain number of flows with randomly chosen sources, sinks, and offered data rates. We use F to denote the total number of flows offered across all networks and $\boldsymbol{r} := (r_1, \ldots, r_F)$ to denote the corresponding vector of maximum data rates. The traffic can take multiple hops to get from the source to the sink, but the relay nodes must be within the same network and the data rate on each hop is limited by the link capacity in Eq. (4.6). Each network can be centrally optimized with perfect knowledge of the path loss between its nodes and the source and destination of each flow, but no information can be exchanged between different networks.

However, to enable collaboration between networks, we assume that each network can infer certain information about the others from their transmissions, namely the node locations and transmit powers. Since concurrent transmissions are generally scattered into different regions to avoid excessive interference, this information can be found through triangulation. Even if certain nodes are mobile, we assume that the relative motion is slow and, with periodic re-estimations, the location and power estimates can be reasonably accurate for the period under consideration. We note that in some practical applications there could be pre-established collaboration protocols allowing networks to share the GPS locations and transmit powers of their nodes over a side channel, e.g., DARPA's Spectrum Collaboration Challenge [90]. This would render unnecessary the aforementioned information inference through triangulation.

The goal of this chapter is to maximize the sum of the rates that can be delivered by all N networks. Unlike many prior works (e.g., [81]), we do not include any fairness criteria in our objective function. Some networks or flows will be encouraged to deliver significantly more data than others if it increases the overall data rate. Additionally, we allow partial delivery of the flows.

4.3 Optimization of a Single Network

The centralized optimization of a single network can be decomposed into two subproblems: 1) mapping the nodes into Tx-Rx pairs for each frequency band, and 2) optimizing the transmit powers and flow routes for the chosen pairings. The search space of the first subproblem is very large: for a network with $2n$ nodes and m frequency bands there are $((2n)!/n!)^m$ possible distinct pairings. We are not aware of any method that can be used to solve the problem exactly, and exhaustive search is computationally intractable for the network sizes that we are considering. Hence, we resorted to a genetic algorithm (GA) for selecting the potentially active links combined with a gradient ascent optimization of the transmit powers.

In a nutshell, the genetic algorithm generates a pool of random pairings and then evaluates their fitness according to the optimal feasible data rate. It then keeps the best candidate in the pool and a few others selected at random with likelihood proportional to their fitness value, thereby simulating the process of natural selection. These randomly selected members experience mutations, consisting of swapping the pairing of two random nodes in each frequency band. Occasionally, one of the mutated candidates achieves a fitness value greater than the previous maximum, thereby becoming the best candidate in the gene pool. This process is repeated until the best candidate does not change for a certain number of iterations.

The second subproblem, i.e., joint routing and power optimization, can be solved with a combination of linear programming and gradient ascent. Given a set of pairings, we have $L = nm$ point-to-point links available to carry the traffic flows. To determine which flows are feasible with these links, we first perform an exhaustive search for all possible (single or multi-hop) paths from the sources to their respective sinks. Denote the total number of paths as P, each connecting a specific source to its sink via one or more links. Let $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_P)$ with θ_k representing the assigned data rate on the k-th path, then naturally our objective is to maximize $\mathbf{1}^T\boldsymbol{\theta} = \sum_{k=1}^{P} \theta_k$ with $\boldsymbol{\theta} \geq \mathbf{0}$. Additionally, we also need to enforce two more constraints:

- The total data rate on any link, found as the sum of the rates for the paths containing that link, cannot exceed its capacity. This can be expressed as $\boldsymbol{A\theta} \leq \boldsymbol{c}(\boldsymbol{p})$, where $\boldsymbol{A}$ is an $L \times P$ matrix with $A_{ij} = 1$ if the j-th path uses the i-th link and $A_{ij} = 0$ otherwise, and $\boldsymbol{c}(\boldsymbol{p}) = (c_1, \ldots, c_L)$ is the vector of link capacities for the transmit powers $\boldsymbol{p} = (p_1, \ldots, p_L)$, as computed by Eq. (4.6).

- The previously mentioned exhaustive search could yield multiple possible paths for each flow. The sum of the rates assigned to those paths must be below the total rate being offered by that flow. This can be expressed as $\boldsymbol{B\theta} \leq \boldsymbol{r}$, where $\boldsymbol{B}$ is an $F \times P$ matrix with $B_{ij} = 1$ if the j-th path delivers the i-th flow and $B_{ij} = 0$ otherwise.

Given a vector $\boldsymbol{p}$ of transmit power for each link, the optimization problem can be formulated as

$$\begin{aligned} \text{maximize} \quad & \mathbf{1}^T \boldsymbol{\theta} \\ \text{subject to} \quad & \boldsymbol{A\theta} \leq \boldsymbol{c}(\boldsymbol{p}) \\ & \boldsymbol{B\theta} \leq \boldsymbol{r} \\ & \boldsymbol{\theta} \geq \mathbf{0}. \end{aligned} \tag{4.8}$$

Unfortunately, the link capacities $\boldsymbol{c}(\boldsymbol{p})$ are not a concave function of the transmit powers, so optimizing $\boldsymbol{p}$ is not a convex problem. We use gradient ascent to seek a locally optimal solution: initialize $\boldsymbol{p}^{(0)}$ with a feasible set of values (e.g., 1/2 on all links) and then iterate between solving problem (4.8) to find the optimal $\boldsymbol{\theta}^\star(\boldsymbol{p})$ and updating $\boldsymbol{p}$ with a step size of μ along the gradient of the objective. In order to ensure that the updated $\boldsymbol{p}$ is feasible, its components are cropped to be between 0 and 1, i.e.,

$$\boldsymbol{p}^{(i+1)} = \left(\min\left(\mathbf{1}, \boldsymbol{p}^{(i)} + \mu \nabla_{\boldsymbol{p}} (\mathbf{1}^T \boldsymbol{\theta}^\star)\right)\right)_+, \tag{4.9}$$

where $(\cdot)_+$ is a function that replaces all negative elements with zeros. The iterations are repeated until convergence.

The gradient in Eq. (4.9) can be found using the chain rule as follows

$$\frac{\partial}{\partial p_\ell} (\mathbf{1}^T \boldsymbol{\theta}^\star) = \boldsymbol{\lambda}^T \cdot \frac{\partial \boldsymbol{c}(\boldsymbol{p})}{\partial p_\ell}, \tag{4.10}$$

where $\boldsymbol{\lambda} := \nabla_{\boldsymbol{c}} (\mathbf{1}^T \boldsymbol{\theta}^\star) = (\lambda_1, \ldots, \lambda_L)$ is the gradient of the optimal objective value with respect to the channel capacity vector $\boldsymbol{c}$ and can be obtained as the dual variables corresponding to the first constraint in problem (4.8) [54]. Applying Eqs. (4.6) and (4.7), Eq. (4.10) can be further expressed as

$$\frac{\partial(\mathbf{1}^T\boldsymbol{\theta}^\star)}{\partial p_\ell} = \frac{\lambda_\ell g_{\ell_o\ell_d}}{I(\ell_d)} - \sum_{\nu\in\mathcal{T}_\ell\backslash\{\ell\}} \left(\frac{\lambda_\nu g_{\ell_o\nu_d} g_{\nu_o\nu_d} p_\nu}{I(\nu_d)[I(\nu_d) - g_{\nu_o\nu_d}p_\nu]}\right), \tag{4.11}$$

where $I(\ell_d) := \sum_{\nu\in\mathcal{T}_\ell} g_{\nu_o\ell_d}p_\nu + \rho_{\ell_d}$ represents the total power (signal, interference, and noise) being received by node ℓ_d. The other variables are defined in the same way as in Eq. (4.7).

4.4 Optimization of Multiple Networks

With multiple networks sharing the same time-frequency resources, we consider two algorithms that the networks may use. In the first algorithm, every network behaves greedily trying to maximize its own data rate irrespective of the interference it causes onto other networks. In the second algorithm, the networks attempt to achieve a better data rate as a whole than the purely greedy approach.

Note that if the networks were allowed to share perfect information (sources and sinks, offered flow rates, etc.) with each other, then the optimal strategy could be computed by a centralized controller with the techniques described in the previous section. However, such computation can be costly because of the super-exponential growth in search space when the centralized controller needs to consider all networks' nodes, and in practical scenarios the different networks may not be able to share every piece of information due to limited inter-network communications capability, privacy concerns, etc.

Our goal is to find a collaborative algorithm that takes advantage of the limited information that each network can infer about its peers and steers the networks to a transmission configuration that resembles what would be found by a centralized controller.

4.4.1 Greedy algorithm

Every network can keep track of the expected amount of noise plus interference from other networks' transmissions at each of its own nodes in each frequency band. In the greedy algorithm, a network does not care about the performance of its peers, and it optimizes its

Algorithm 1: Greedy and collaborative networks algorithms

```
for $iter \leftarrow 1$ to $nIterations$ do
    for $i \in \{networks\}$ do
        for $(j, f) \in nodes(i) \times \{frequency\ slots\}$ do
            $\rho_j^{(f)} \leftarrow$ noise plus interference estimate at node $j$ at frequency slot $f$;
        $links(i), \boldsymbol{p}, \boldsymbol{\theta} \leftarrow$ optimized links, powers and flow scheduling of network $i$ by
          GA and Eqs. (4.8) and (4.9);
        if $usingCollaboration$ then
            refine link powers by Algorithm 2;
        update transmission according to $links(i), \boldsymbol{p}, \boldsymbol{\theta}$;
    // for data analysis
    evaluate total throughput, given the transmissions on all networks;
```

Algorithm 2: Refine link powers

```
for $\ell \in links(i)$ do
    // discard links of low rate-to-power ratio
    if $[\boldsymbol{A\theta}]_\ell / p_\ell \leq \eta$ then
        $p_\ell \leftarrow 0$;
```

transmission powers and flow routing treating the interference as white noise [80]. The same algorithm described in section 4.3 is used, except that the constant noise ρ_{ℓ_d} in Eq. (4.7) is replaced with the corresponding noise plus interference estimates. The networks then take turns updating their estimates and re-optimizing their transmissions. Algorithm 1 with *usingCollaboration* set to false shows the pseudocode for the greedy algorithm.

Although the greedy algorithm is able to provide a feasible transmission plan, it is clearly sub-optimal. An optimal configuration should only allow a link to transmit when it yields a throughput greater than the total decrease in throughput that its interference is causing on other networks. The greedy algorithm neglects this consideration and results in a transmission plan that is often too crowded. Every network tries to shout as loudly as possible to get their data through, causing severe interference to its peers and having a counterproductive effect. A typical sample of the transmission links found by the greedy algorithm, discussed in more detail in section 4.6, can be seen in Fig. 4.4.

4.4.2 Collaborative algorithm

In light of the above considerations, we design a collaborative algorithm which starts from the greedy solution in each iteration and then refines it by discarding links that offer a low rate-to-power ratio. Namely, link ℓ is discarded if $[\boldsymbol{A\theta}]_\ell / p_\ell \leq \eta$ where $\boldsymbol{A}$ is taken from problem (4.8), $[\,\cdot\,]_\ell$ denotes the ℓ-th component of a vector (so $[\boldsymbol{A\theta}]_\ell$ is the data rate on link ℓ), and η is a threshold that will be described later. If a specific node transmits with high power but only gets a small throughput due to excessive interference, it should stop transmitting. The removal of such a link will have a small impact on the performance of that network, but the interference it caused could have been having a significant impact on peer networks.

Some studies have attempted to facilitate collaboration by assuming that networks can share certain information, such as the GPS locations of their nodes and their planned transmissions, with their peers. One of these studies is the DARPA Spectrum Collaboration Challenge (SC2), where ensembles of intelligent SDR networks exhibited autonomous collaborative behaviors and outperformed traditional RF schemes with evenly divided or individually reserved spectrum bands. The challenge was a big success, but its outstanding performance relied heavily on collaboration messages between the networks through a side channel [90]. This chapter, on the other hand, assumes that the networks lack a protocol for exchanging collaboration messages, and proposes a collaboration algorithm which only uses inferred information from other networks' transmissions.

Algorithm 1 with *usingCollaboration* set to true summarizes the collaborative algorithm.

4.5 Optimal Rate-to-Power Threshold

This section will justify the heuristic of discarding the links with low rate-to-power ratio and propose a method for choosing the threshold η. The greedy algorithm in the previous section uses Eqs. (4.10) and (4.11) to find the gradient of the network capacity with respect to the transmit powers, treating other networks' transmissions as noise. One natural extension to this scheme is to include the expected capacity of peer networks into the objective function of problem (4.8).

Let $\mathcal{S}$ and $\mathcal{P}$ be the set of nodes of our own network and of all the peer networks, respectively. Since a network can estimate the locations and transmit powers of the peer nodes through triangulation, it can also estimate the total received radio power at each node. For a node i, denote its location as x_i, and let $\hat{g}_{ij} := \|x_i - x_j\|^{-2}$ be the estimated channel gain between nodes i and j. Denote the estimated transmit power of node i as $\hat{p}_i$. Let $I(j)$ be the total received power (including signal, noise, and interference) at node j, and let $\tilde{I}_i(j) := I(j) - \hat{g}_{ij}\hat{p}_i$ be the noise-plus-interference estimate at node j when node i is the transmitter it is paired with.

For peer networks, although we are aware of their transmitters, we do not know the routing information, i.e., which node is intended as the receiver, nor do we know which nodes belong to which networks. Thus we assume that every peer node is equally likely to be the receiver of any transmitting nodes. In this way, we can express the total expected capacity of peer networks as follows:

$$E[C_{\text{peers}}] = \frac{1}{|\mathcal{P}| - 1} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{P} \setminus \{i\}} \log \left(1 + \frac{\hat{g}_{ij}\hat{p}_i}{\tilde{I}_i(j)} \right). \tag{4.12}$$

Adding this to the objective function in problem (4.8) results in $\mathbf{1}^T \boldsymbol{\theta} + \alpha E[C_{\text{peers}}]$, where α is a parameter that determines how "considerate" our network is in weighing our own data rate with respect to that in peer networks. Unfortunately, the objective function is not concave and gradient ascent is not guaranteed to converge to a global maximum, as shown by the following example.

Example 1. *Consider a simple case where our two nodes and two peer nodes are located at the corners of a unit square. The transmissions are along opposite sides of the square, and in opposite directions. Suppose the peer transmission is at half power* 0.5*, our transmit power is* p*, and the noise variance is* 0.01*, then the total capacity is*

$$C(p) = \log \left(1 + \frac{p}{0.51} \right) + \log \left(1 + \frac{0.5}{p + 0.01} \right).$$

dC/dp is always positive when $p > 0.49$ and in fact C goes to infinity as p increases, but this is infeasible due to the unit power constraint. If we start with an initial value of p greater

than 0.49*, then gradient ascent eventually pushes* p *to* 1*, whereas on the interval* $p \in [0,1]$*,* C *actually achieves its maximum at* $p = 0$*.*

This example shows that although it may locally appear that increasing the power of a link is beneficial to the overall capacity, turning it off completely can offer a larger gain in peer networks' capacity that offsets our loss. This is also supported by the empirical evidence in Fig. 4.1, which shows that dropping links according to the rate-to-power ratio typically performs better than gradient ascent.

The rate-to-power threshold that determines whether a link is dropped cannot be a fixed constant. For networks with different parameters, there exists different optimal thresholds to use. For instance, if the network is densely packed with nodes competing to transmit, it is desirable to limit the number of simultaneous transmissions to reduce interference, so more links should be dropped; however, if the nodes are scattered across a very large region, the background noise dominates the interference and it is better to allow more concurrent transmissions to achieve a higher network data rate, and thus a lower threshold should be used. We will derive a method to heuristically find the optimal threshold to use for collaborative networks, and verify the result through numerical simulations.

Consider an active link ℓ in our network with power p_ℓ. If it is turned off, we lose the data rate currently carried by that link $[\boldsymbol{A\theta}]_\ell$, but the expected capacity for peer networks and our remaining links will increase due to the reduced interference. The gain in our own network capacity can be expressed as

$$\Delta C_{\text{us}} = \sum_{\nu \in \mathcal{T}_\ell \setminus \{\ell\}} \log \left(1 + \frac{\hat{g}_{\nu_o \nu_d} p_\nu}{\tilde{I}_{\nu_o}(\nu_d) - \hat{g}_{\ell_o \nu_d} z} \right) \Bigg|_{z=0}^{z=p_\ell} . \tag{4.13}$$

For peer networks, since we do not know the routing information, we assume that every peer node is equally likely to be the receiver of every transmission. Then, we can find the total expected capacity gain in peer networks as follows:

$$\Delta E[C_{\text{peers}}] = \frac{1}{|\mathcal{P}| - 1} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{P} \setminus \{i\}} \log \left(1 + \frac{\hat{g}_{ij} \hat{p}_i}{\tilde{I}_i(j) - \hat{g}_{\ell_o j} z} \right) \Bigg|_{z=0}^{z=p_\ell} . \tag{4.14}$$

If a network finds that the data rate on link ℓ is smaller than the overall expected gain in capacity, it should drop the link, and vice versa. In other words, the condition that link ℓ should be dropped is

$$[\boldsymbol{A\theta}]_\ell < \Delta C_{\mathrm{us}} + \Delta E[C_{\mathrm{peers}}]. \tag{4.15}$$

Applying the mean value theorem to Eqs. (4.13) and (4.14), we find that this condition is equivalent to

$$\begin{aligned} \frac{[\boldsymbol{A\theta}]_\ell}{p_\ell} < &\sum_{\nu \in \mathcal{T}_\ell \setminus \{\ell\}} \frac{\hat{g}_{\ell_o \nu_d} \hat{g}_{\nu_o \nu_d} p_\nu}{K_{\nu_o \nu_d}(K_{\nu_o \nu_d} + \hat{g}_{\nu_o \nu_d} p_\nu)} \\ &+ \frac{1}{|\mathcal{P}| - 1} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{P} \setminus \{i\}} \frac{\hat{g}_{\ell_o j} \hat{g}_{ij} \hat{p}_i}{K_{ij}(K_{ij} + \hat{g}_{ij} \hat{p}_i)}, \end{aligned} \tag{4.16}$$

for some $0 < \xi < p_\ell$, where $K_{ij} := \tilde{I}_i(j) - \hat{g}_{\ell_o j}\xi$.

This shows that dropping links with low rate to power ratio is likely to be beneficial for the ensemble. The exact threshold below which the links should be dropped is unknown, since the mean value theorem guarantees the existence of ξ but does not specify how it can be found. However, this serves as a justification for our policy of using the rate-to-power ratio as a metric to determine whether a link should be kept or dropped.

We sort the links in increasing order of the rate-to-power ratios, assume that they are turned off sequentially, and compute the expected network capacity gain according to Eqs. (4.13) and (4.14). Note that the estimated interference $\tilde{I}_i(j)$ for each pair of nodes (i, j) should be updated every time a link is turned off. Finally the threshold for the links to be dropped is chosen to be at the point that maximizes the accumulated (expected) gain in network capacity minus the accumulated loss in link data rate. This procedure is summarized in Algorithm 3.

4.6 Numerical Results

This section simulates multiple networks with nodes uniformly distributed over a 100 m × 100 m square region and background noise variance of $\sigma^2 = 10^{-4}$. Each network will be asked to carry a few flows with randomly chosen sources and sinks. Subsection 4.6.1 focuses on a

Algorithm 3: Find rate-to-power ratio threshold

```
sort links in increasing order of rate-to-power ratio;
$\eta_0, a_0, b_0 \leftarrow 0$; $k \leftarrow 1$;
$\boldsymbol{p}_{\mathrm{copy}} \leftarrow \boldsymbol{p}$;
for $\ell \in \mathrm{sorted}(\mathit{links})$ do
    $\eta_k \leftarrow [\boldsymbol{A\theta}]_\ell / p_\ell$;
    $a_k \leftarrow \Delta C_{\mathrm{us}} + \Delta E[C_{\mathrm{peers}}]$ by Eqs. (4.13) and (4.14);
    $b_k \leftarrow [\boldsymbol{A\theta}]_\ell$;
    $p_\ell \leftarrow 0$;
    for $(i, j) \in (\mathcal{S} \cup \mathcal{P})^2$ do
        update interference estimate $\tilde{I}_i(j)$;
    $k \leftarrow k + 1$;
$k^* \leftarrow \arg\max_{k \geq 0} \sum_{k'=0}^{k} (a_{k'} - b_{k'})$;
$\boldsymbol{p} \leftarrow \boldsymbol{p}_{\mathrm{copy}}$;
return $\eta_{k^*}$;
```

simplified scenario with homogeneous networks, a single frequency band, no maximum data rates, and direct links only (no multi-hop). These results support the validity of our method and conclusions in the absence of routing and saturation effects. Subsection 4.6.2 then presents simulation results for heterogeneous networks with different sizes and requirements. These results illustrate the performance of our method in a more complex scenario.

4.6.1 Single-hop

Before we attempt to study a complex scenario with multi-hop, multiple frequency bands, and bounded flows, we want to analyze the capacity of our algorithm in a simplified case with a single channel, point-to-point links, and unlimited offered rates for every flow. There is therefore no need for routing or scheduling different sets of links. The problem is reduced to finding the set of links that should be simultaneously activated so as to maximize the overall capacity of the ensemble. Specifically, we performed our simulations in a scenario with $N = 8$ networks with $n_i = 8$ nodes each, and $F = 32$ flows (4 per network).

First, we wanted to check whether our solution, which drops some links completely and leaves others untouched, is likely to provide better results than progressively adjusting the power of all the links according to a gradient descent algorithm. We implemented both options and saw that it was indeed the case. Fig. 4.1 shows the overall data rate that both algorithms provide, averaged over eight ensemble instances. It can be seen that our algorithm not only converges faster than gradient descent, but it also reaches higher data rates.

We then wanted to compare our algorithm with the greedy and centralized solutions. Fig. 4.2 once again illustrates the gain in overall capacity relative to the initial solution for all three algorithms, showing that our algorithm greatly outperformed the greedy solution in all the network instances that we simulated. Furthermore, there is a relatively small gap between the performance of our algorithm and that of the centralized solution.

4.6.2 Multi-hop

This subsection addresses a more complex scenario, with $N = 4$ heterogeneous networks consisting of 2, 4, 8, and 10 nodes, and 1, 1, 4, and 5 flows, respectively. It will be assumed

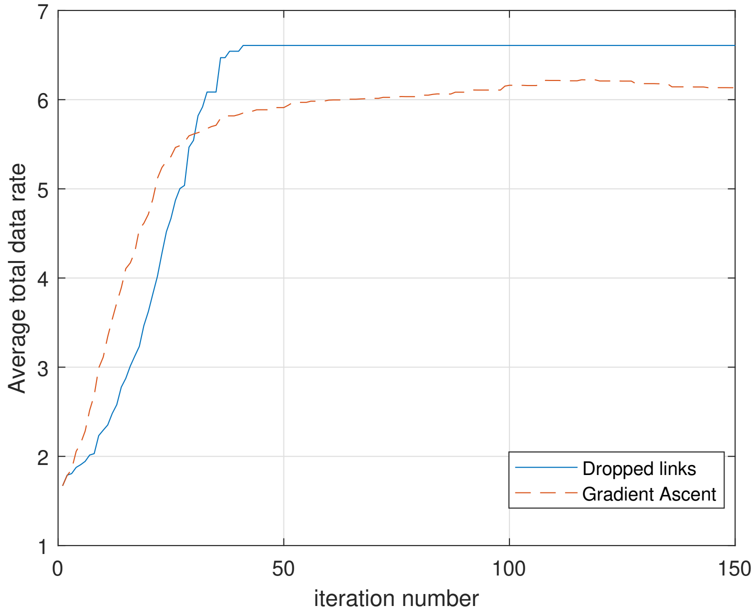


Figure 4.1. Total data rate averaged over eight instances of eight networks with eight nodes each. Dropping links offers a clear advantage over smooth power adaptation.

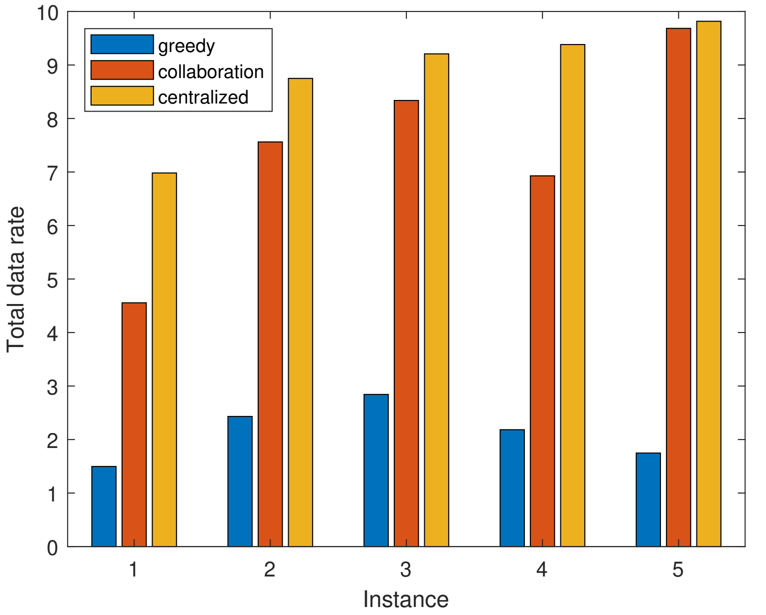


Figure 4.2. Total data rate for five different instances of eight networks, unlimited offered rate, without multi-hop. Our collaborative algorithm is significantly better than greedy in all instances and not much worse than centralized.

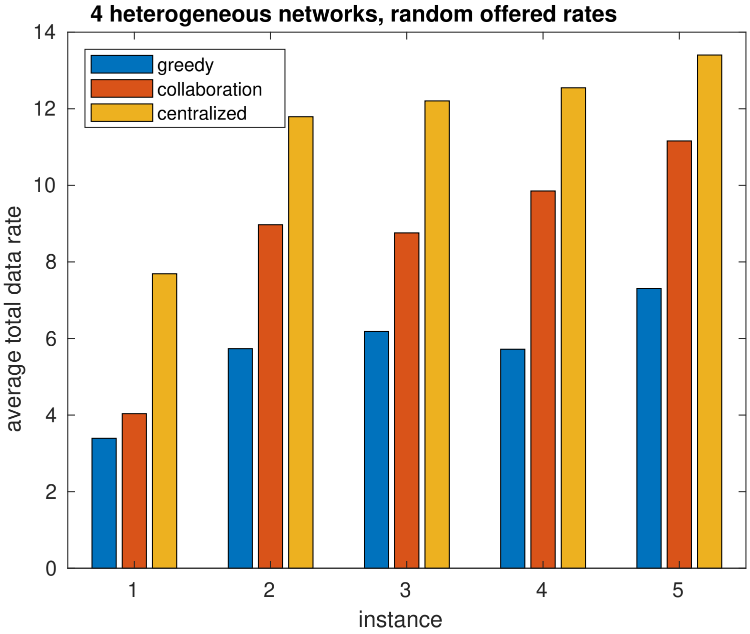


Figure 4.3. Average total data rate for five different instances of four networks. The offered data rate on each flow is uniformly distributed in $[3, 5]$.

that there are $m = 4$ available frequency channels for transmission, all of them with equal bandwidth and background noise $\sigma^2 = 10^{-4}$.

We generate five different instances of the 4 networks, with the offered rate per flow uniformly distributed in $[3, 5]$, and simulate three transmission schemes: centralized, greedy, and collaborative. The centralized method consists of applying the GA algorithm described in section 4.3 to all the nodes as if they were a single network, while ensuring that the paths for each flow do not cross between networks. The resulting solution should be close to optimal. The greedy and collaborative methods correspond to those described in section 4.4 and Algorithms 1 and 2. The greedy and collaborative methods are iterative, so they can occasionally get caught up in a cycle of updates without converging. To address this possibility, our plots report their average data rate over the last 20 iterations, which should be a representative measure of their average data rate in practice.

The total data rate of the five instances, averaged as previously mentioned, is shown in Fig. 4.3. It can be observed that the collaborative algorithm outperforms the greedy algorithm by a wide margin in all five instances, and the centralized solution has the highest

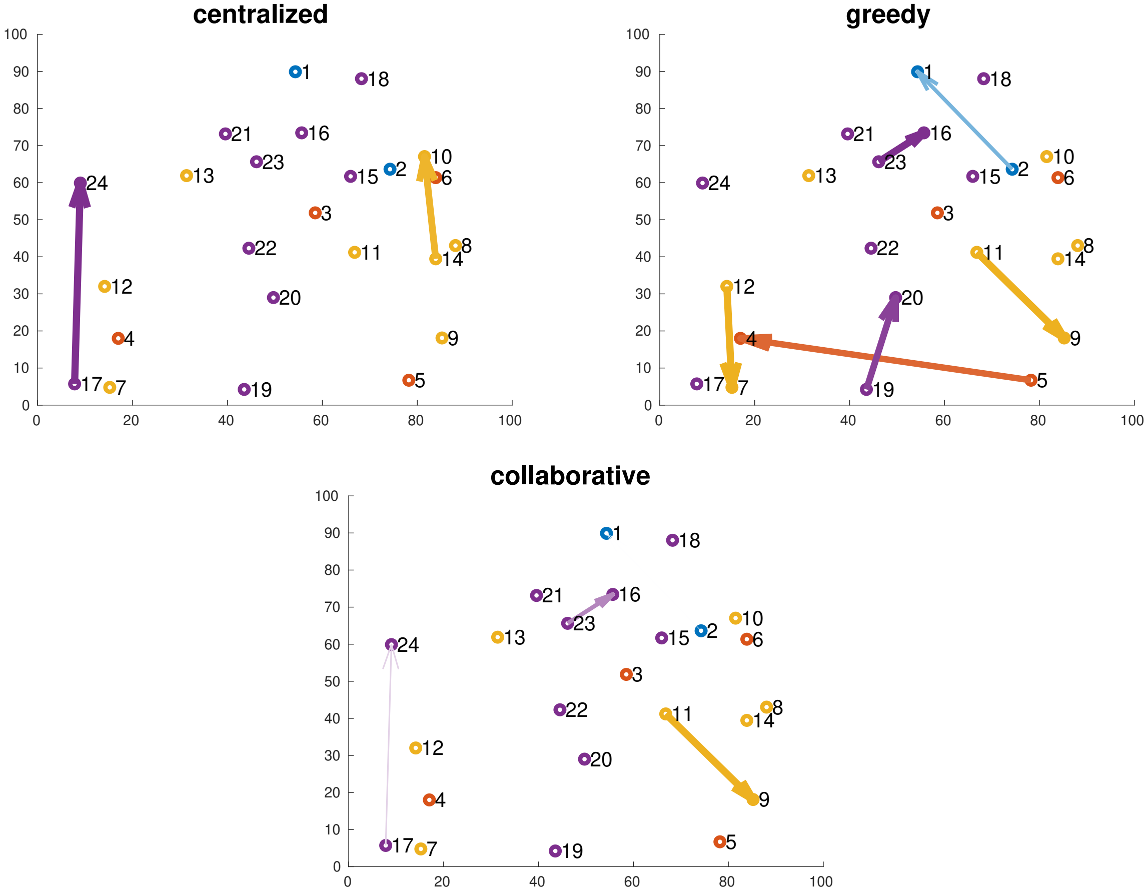


Figure 4.4. Sample links and powers with different algorithms.

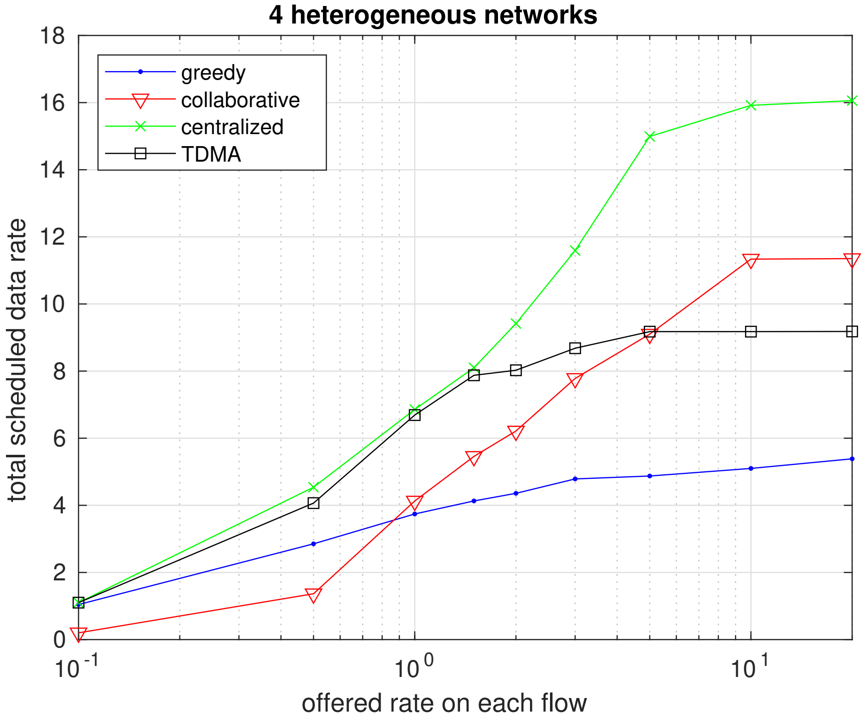


Figure 4.5. Average total throughput with different offered rates.

throughput in all instances, as expected. Fig. 4.4 illustrates a few representative samples of the links and powers suggested by the three algorithms for one frequency channel in instance 5. The different colors represent different networks, and the thickness and color intensity of an arrow indicate the power of the link. A visual comparison of these chosen links and powers shows an important difference between them: the pseudo-optimal (centralized) and collaborative solutions have a lot fewer links than the greedy solution. Although the figures only show a single frequency band, the same is observed in most of the others.

Fig. 4.5 shows the total data rate that the algorithms can achieve as a function of the offered rate per flow. We take instance 4 from Fig. 4.3 and impose identical offered data rate for all flows, with values ranging between 0.1 and 20. In addition to the three previously described algorithms, the figure also shows the performance of a fourth scheme labeled "TDMA", which splits time evenly among the four networks allowing them to use all $m = 4$ frequency channels exclusively during their allotted interval. This is essentially equivalent to the status quo of the licensed RF spectrum, which was used as a framework for comparison in

SC2 [90]. In each exclusive interval, a network optimizes the transmission using the method in section 4.3. As before, the throughput shown for the greedy and collaborative algorithms corresponds to an average over the last 20 iterations.

At the lowest offered data rate, all flows can be easily delivered by the greedy, centralized, and TDMA solutions. Our collaborative algorithm, however, compels the networks to drop some of their links sacrificing some performance. This seems counterproductive from a global perspective, but it is important to remember that networks operate independently without any communication between them. Since those links are using a significant amount of power and barely carrying any traffic, the networks believe that dropping them would benefit their peers. In a practical scenario, it would be sensible to include a failsafe mechanism that keeps all the links active (or defers to the greedy method) when the ensemble can support it.

When the offered data rate is less than 1.5 per flow, there is no significant difference between the optimal and TDMA solution; the offered rate is low enough that evenly splitting the resources does not throttle the network that needs them most. At this point the improvement that collaboration brings upon greedy is moderate.

As the offered rate increases past 1.5, the TDMA performance saturates while the collaborative method quickly gains an edge over the greedy and eventually over the TDMA methods. The greedy performance is severely limited by the high interference that it causes. When the offered rate is more than 5, the collaborative algorithm surpasses TDMA because it divides the resources among the 4 heterogeneous networks more effectively than simply splitting them evenly. At the highest offered rate of 20, all 4 methods have saturated.

4.7 Conclusion

This chapter studied the problem of routing and resource allocation in a communications network. It proposed algorithms suitable for a single network and for multiple collaborative communications networks sharing the same time-frequency resources to autonomously achieve a transmission assignment that reduces interference and increases throughput. The collaborative algorithm employs heuristics, uses only information inferred from overheard transmissions, and does not require any side information exchanged between the networks.

Numerical simulations show that the collaborative algorithm significantly outperforms the non-collaborative greedy approach, and it outperforms the uniform splitting of resources by a moderate margin for high offered data rates and heterogeneous networks.

5. CHANNEL ALLOCATION ALGORITHM FOR COGNITIVE RADIOS IN THE DARPA SPECTRUM COLLABORATION CHALLENGE

5.1 Introduction

The Spectrum Collaboration Challenge (SC2) was launched by DARPA and lasted for 3 phases from 2017 to 2019. It seeks to find a solution for the long-existing problem of radio spectrum scarcity, which is getting more and more significant with the increasingly rapid growth in the worldwide demand of more data over the wireless spectrum. In SC2, the participating teams are challenged to each develop an intelligent software defined radio (SDR) network, known as a Collaborative Intelligent Radio Network (CIRN). The CIRNs from different teams share the same very crowded spectrum for their data channels, and they need to compete against each other to deliver as much intra-network traffic as possible, while being collaborative and intelligent in order not to jam each other. The collaboration is made possible by cognitive abilities of every radio and a separate broadcast channel (referred to as the collaboration channel) where networks can share collaboration messages that describe their network status, including information such as node locations, current transmission channels, performance of each node etc.

The wireless channel between each pair of transmitting and receiving nodes is simulated by a very large channel emulator developed by DARPA, known as the Colosseum. The Colosseum simulates various practical scenarios, where each unique scenario is characterized by the channel conditions, location and mobility of the users, and the rise and fall in traffic etc. The transceiving nodes in each network are given data packets from the Colosseum to be delivered to their designated destinations, and there is a different set of requirements for every type of data, such as a continuous minimum data rate for a data stream, or a deadline line before which a file must be delivered. Each of these is called a mandate, and it is associated with a score. Briefly speaking, a team's overall score is determined by the set of mandates that they can achieve, conditioned on a minimum score achieved by all the competing teams, so the teams cannot simply be selfish and interfering with others is indirectly penalized.

The BAM! Wireless team consists of graduate students and faculty members from Purdue University and Texas A&M University. This chapter will discuss certain aspects of the technical design of the BAM! Wireless CIRN, especially the dynamic channel allocation algorithm which determines the transmitting channel of each node. It takes into account both the spectrum availability at our own receiving nodes by reading their power spectral density (PSD) measurements, as well as the interference that a transmission channel may cause to other teams. This algorithm allows for potential inter-network spectrum reuse if the interference level is low.

5.2 BAM! Wireless CIRN Overview

We present in this chapter important aspects of the technical design of the BAM! Wireless Collaborative Intelligent Radio Network (CIRN) that participated in Phase 2 of the DARPA SC2. The CIRN consists of multiple Standard Radio Nodes (SRN) and one gateway node, as depicted in Fig. 5.1. All nodes interact directly with the Colosseum emulator, while only the gateway can send and receive collaboration messages according to the defined CIRN Interaction Language (CIL).

Following the Phase 2 scoring procedure, the approach behind the BAM! Wireless CIRN design is centrally focused on maximizing the minimum number of achieved mandates by a team in any given match (ensemble minimum), by regularly monitoring the reported performance and use of resources of peer teams through the received collaboration messages. The CIRN design specifically relies on lightweight intelligent algorithms that are robust to changing environmental conditions. The final outcome is a robust intelligent solution that jointly optimizes the decisions for the modulation and coding scheme, flow scheduling, as well as channel allocation to maximize the ensemble performance in the wide range of Colosseum scenarios. We next describe in detail the various technical components of the CIRN.

A high-level description of our CIRN is provided in Fig. 5.2. Note that the Network Controller function pertains only to the gateway node; all other components and features

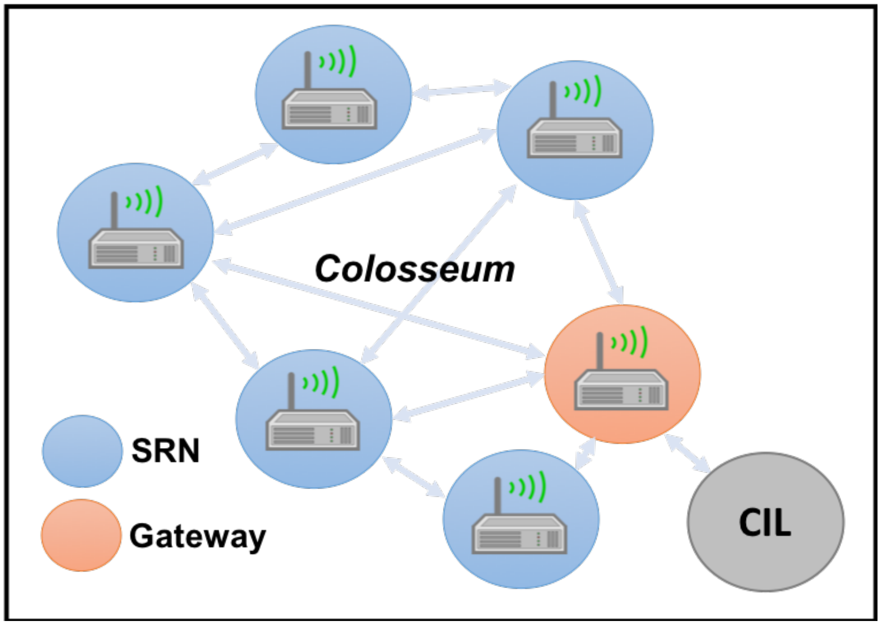


Figure 5.1. BAM! Wireless Network for the DARPA SC2.

are common to all SRN nodes as well as the gateway. We use a centralized control structure for channel allocation: the gateway node gathers all the needed information from each SRN, makes the channel allocation decision for every node in the network, and then broadcast this decision to all the nodes. Every node only transmits on the one channel that it is assigned, but receives on multiple channels that belong to other nodes. The channel allocation algorithm resides in the Network Controller in Fig. 5.2.

5.3 Channel Allocation

The gateway first looks at our historical performance, and decides whether our performance is the worst in the ensemble or is much better than the peer networks in the ensemble. If we are the worst, the gateway re-allocates all of our channels; if we are much better than our peers, we reduce the bandwidth of the widest channel to allow for more space in the spectrum for the peer networks. This procedure is summarized in Algorithm 4.

When the gateway node decides to re-allocate all channels, it first estimates the channel gain between each of our nodes and all others, assumed to be frequency-independent, symmetric, and slowly varying. These gains are initialized using the GPS locations reported

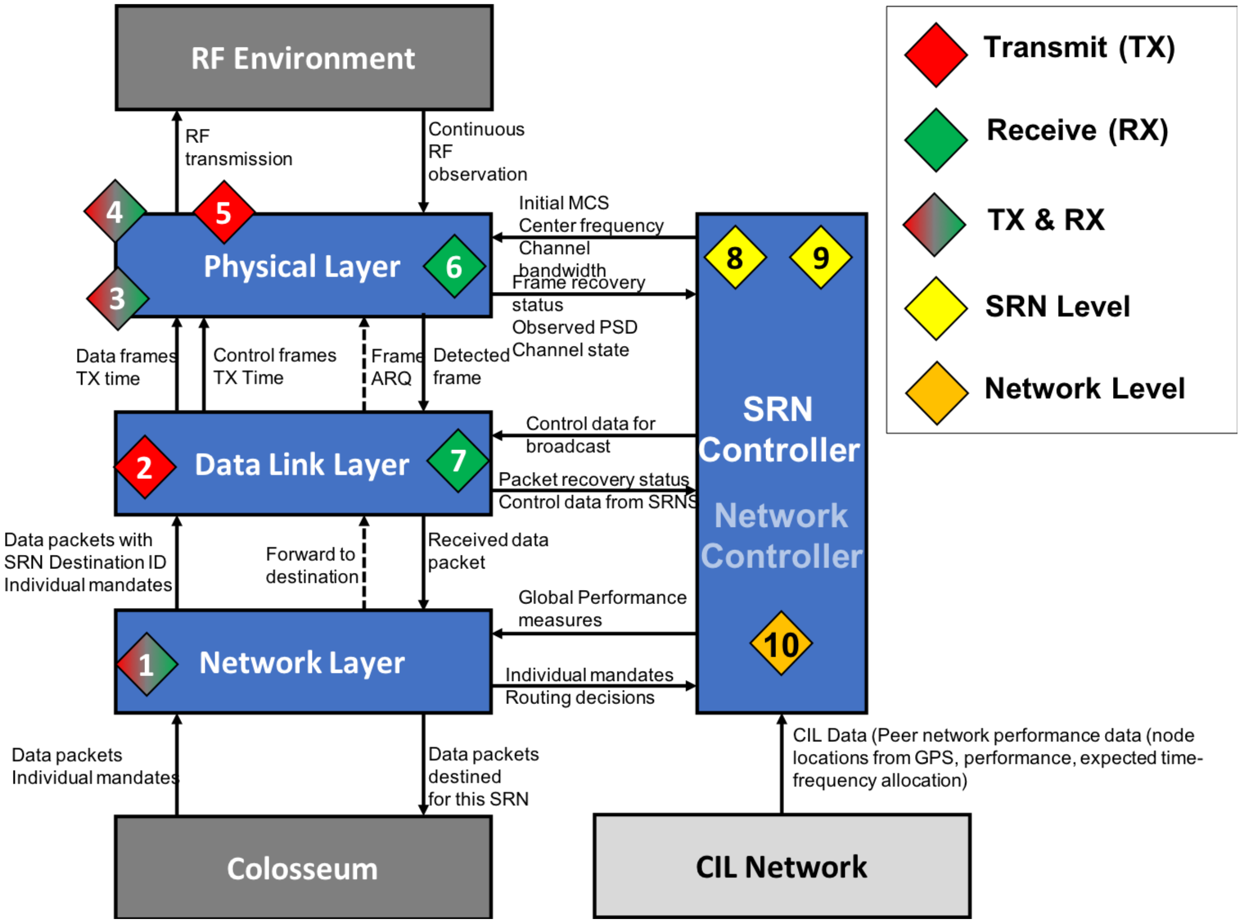


Figure 5.2. High-level diagram showing the different layers and functional components of the BAM! Wireless CIRN. The numbered features in diamonds are: 1- Multi-Hop Routing, 2- Transmit Scheduling, 3- Broadcast of Control Data, 4- High Rate Data Link, 5- Transmit Power Control, 6- Adaptive Modulation and Coding Scheme, 7- Automatic Repeat Request, 8- Program Configuration through Database, 9- Visualization of Performance Measures, 10- Channel Allocation Algorithm.

Algorithm 4: Pseudo Code for Channel Allocation Algorithm - Routine call every one second

```
if we are the worst performing team for 5 consecutive routine calls then
    call channel re-allocation algorithm;
else if we are two mandates above the worst performing team for 5 consecutive
  routine calls then
    reduce bandwidth allocated for widest channel;
```

through the CIL and assuming free space propagation with loss exponent of 3. These values are then refined adaptively based on our own PSD measurements and CIL messages from other nodes specifying their transmit power and frequency. The channel gain estimates and the spectrum usage collaboration messages from other teams are hence used to construct an interference map for each frequency.

The interference spectrum is different for each of our nodes. The gateway node goes through all the frequency bands and determines the value that each band would have for each node. The value function is computed as the weighted average of the SNR estimates from a node i to all of the receiving nodes j in our network, at center frequency f_c:

$$V(i, f_c) = \sum_{j \neq i, j \in BAM} w(i,j) SNR(i,j,f_c), \tag{5.1}$$

This value function takes into account the interference maps as well as the flows that each node needs to deliver. A detailed explanation of the symbols can be found in Algorithm 5. The estimated SNR from node i to a specific node j at center frequency f_c is computed as the average ratio of the channel gain to the interference power at the receiver over the bandwidth centered at f_c:

$$SNR(i,j,f_c) = \frac{1}{B(i)} \sum_{f=f_c-\frac{B(i)}{2}}^{f=f_c+\frac{B(i)}{2}} \frac{G(i,j)}{IP(j,f)}, \tag{5.2}$$

where $IP(j,f)$ denotes the interference power at node j at a specific frequency f, which is estimated based on the PSD measurement at node j.

The gateway node then assigns each node a channel (center frequency and bandwidth) to transmit on. This choice is made through a combination of a greedy algorithm that attempts to maximize the sum of the values and a considerate algorithm that reallocates channels to help improve the ensemble minimum. The channel allocation mechanism responds to environmental updates and changes in offered mandates. If a node needs to transmit more traffic to a specific receiver, or the flows have more stringent requirements, the gateway will take that into account when picking the center frequency.

It is important to highlight that an obstacle we faced towards making the channel allocation algorithm more biased towards a collaborative behavior was that the information that we were receiving from peer nodes (specifically, the GPS locations and spectrum usage) was frequently inaccurate.

In summary, each of our nodes will receive on multiple channels, but transmit on a single one. The bandwidth required for each node is chosen based on the offered load. See Algorithm 5 for a pseudo code of the channel allocation algorithm.

5.4 Results

We show in Fig. 5.3 an example behavior of the channel allocation mechanism in an Alleys of Austin scenario. We note that the apparent vacancies in the spectrum is due to blockages at the receiver. Certain peer network transmissions are too faint to be picked up on the PSD, and if we combine the PSD from all of our nodes the spectrum is in fact very crowded.

Algorithm 5: Pseudo Code for Channel Re-allocation Algorithm

```
Input: Vector of bandwidths assigned to the receivers.
Data: $IP(j, f)$: Estimated interference power at node $j$ and frequency $f$.
$G(k, j) = G(j, k)$: Channel gain between nodes $k$ and $j$.
$P(k)$: Transmit power at node $k$.
$\mathcal{N}(j)$: The set of neighbors (with significant interference) of receiver at node $j$.
$d(k, j)$: Distance between nodes $k$ and $j$.
$B(i)$: Bandwidth allocated for transmitter at node $i$.
$\mathcal{F}$: Set of all possible discretized frequencies.
$SNR(i, j, f_c)$: Estimated Average SNR between transmitter at node $i$ and receiver at node $j$ if center frequency $f_c$ is selected.
$V(i, f_c)$: Estimated value of selecting center frequency $f_c$ for the transmitter at node $i$.
$w(i, j)$: Weight proportional to traffic from node $i$ to node $j$.
$BAM$: Set of indices for nodes in our network.
for $j \in BAM$ do
    compute $IP(j, f)$ based on the PSD measurement;
for $i, j \in BAM$ and $f_c \in \mathcal{F}$ do
    compute $SNR(i, j, f_c) = \frac{1}{B(i)} \sum_{f=f_c-B(i)/2}^{f=f_c+B(i)/2} \frac{G(i,j)}{IP(j,f)}$;
for $i \in BAM$ and $f_c \in \mathcal{F}$ do
    compute $V(i, f_c) = \sum_{j \in BAM \setminus \{i\}} w(i, j) SNR(i, j, f_c)$;
Seek allocation $\arg\max_{f_c : BAM \to \mathcal{F}} \sum_{i \in BAM} V(i, f_c(i))$ without channel overlap, i.e.,
  find a mapping from each node $i$ to a center frequency $f_c(i)$ to maximize total
  value function, subject to $|f_c(i) - f_c(j)| \geq \frac{1}{2}(B(i) + B(j))$ for $i \neq j$. This is done
  through a greedy depth first search in decreasing order of required bandwidth $B(i)$.
```

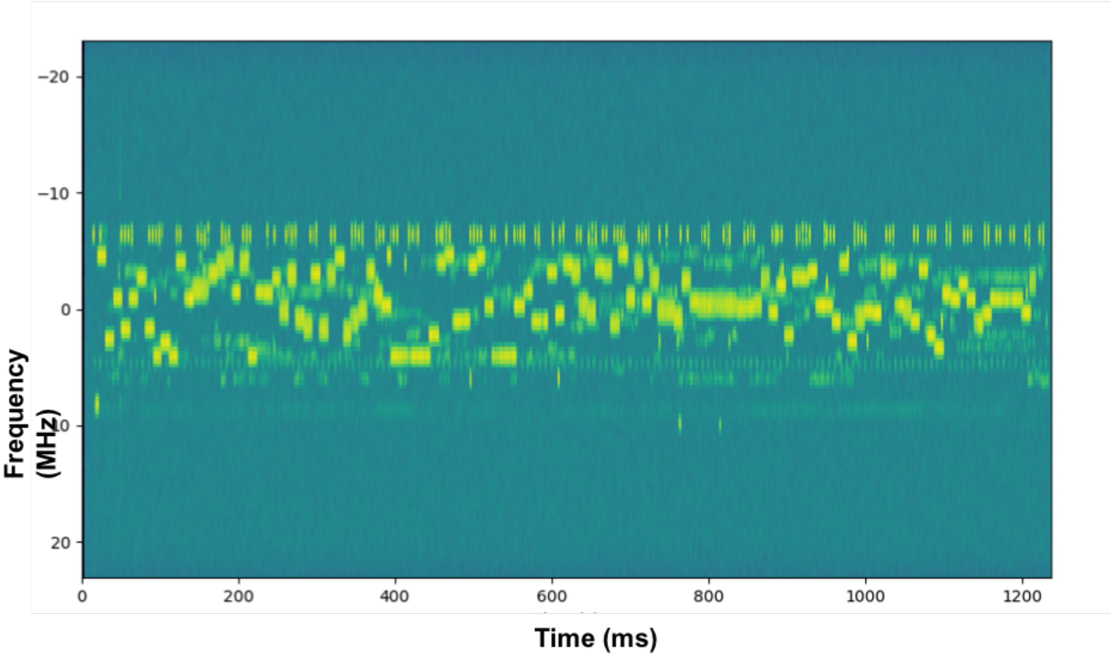


Figure 5.3. Occupied spectrum in an Alleys of Austin scenario.

6. SUMMARY AND FUTURE WORK

This thesis studies several areas of wireless communications that can benefit from the use of adaptive methods and dynamic resource allocation to improve their performance. This chapter summarizes the work and points out some possible future research directions.

Chapter 2 studies existing HARQ techniques and proposes adaptive HARQ schemes with codeword bundling that are suitable for a variety of scenarios including single-link systems, relay systems, and multi-user broadcast systems. The IR bits sent by our proposed HARQ protocol are optimized under the assumption that the feedback channel can only support a few bits of feedback per bundle of codewords (or packets). Apart from the traditional extension IR bits consisting of a few additional bits for each codeword, this thesis also considers bundle IR, consisting of encoded IR bits which the receiver can use to refine the LLRs in multiple codewords. Simulation results show that the proposed methods provide a modest increase in throughput with respect to traditional fixed-length HARQ schemes, where codewords are individually acknowledged instead of bundled. It also proposes an adaptive relaying policy which achieves a significant gain in throughput compared to relays with fixed forwarding strategy. Simulations show that the proposed policy greatly reduces the total number of IR bits needed in multi-user systems.

The relay system studied in chapter 2 for which the HARQ scheme is proposed is a 2-hop system. Future work may extend this result to multi-hop relays, or even to a combination of chains of relays and broadcasting nodes, forming a network of nodes with optimized HARQ bits strategically sent from certain nodes.

Chapter 3 shows that the overall data rate of a communication link using ECC can be improved by grouping bits into codewords according to their reliability and exploiting correlation between codewords when decoding. Specifically, in a system using 16-QAM (or higher), the bits in a modulation symbol present different error rates. Further, the modulation symbols are not symmetric themselves, for example knowing the first bit of a symbol will alter the conditional distribution of the second bit. This thesis proposes methods to leverage these two asymmetries and shows the improvement in goodput through simulations. In addition, theorem 3.3.1 proves that the proposed method for grouping the bits increases the maximum achievable channel coding rate for finite blocklength codes in a binary-input parallel AWGN channel. The gain is found to be most significant when the channel experiences high SNR variations.

However, the proof for the AWGN channel is still an approximated analysis to what we propose for the 16-QAM scenario. The modulation symbols effectively create parallel channels for the bits, but the channels are not Gaussian. Thus one possible extension to this work is to generalize the proof to channels other than AWGN, including the 16-QAM scenario.

Chapter 4 studies the problem of routing and resource allocation in a communications network. It proposes an algorithm suitable for multiple collaborative communications networks sharing the same time-frequency resources to autonomously achieve a transmission assignment that reduces interference and increases throughput. The algorithm enables collaboration using only information that can be inferred from other networks' transmissions, and no side information is shared between the networks. Numerical simulations show that the collaborative algorithm performs significantly better than a non-collaborative greedy approach. It also outperforms the uniform splitting of resources by a moderate margin for high offered data rates and heterogeneous networks.

Chapter 5 presents an adaptive algorithm that allocates disjoint transmission channels in a cognitive radio network used by Team BAM! Wireless' submission to the DARPA SC2. The algorithm computes a value function that measures the benefit that each possible frequency has for a given network. The value function combines an estimation of the SNR and the interference a network causes to peer networks on each frequency. The estimation uses PSD information collected from our cognitive SDR nodes and augmented with collaboration messages received from other networks. By maximizing the value function, the algorithm finds a feasible set of disjoint channels for our network while being considerate to peer networks, thus allowing for inter-network collaborative behaviors that improve the overall throughput.

The last two chapters of the thesis deal with the optimal allocation of resources among collaborative communication networks. Such problems are NP-hard, so our proposed solutions all utilized heuristics, and the effectiveness is verified by simulations and experiments.

With the rapid advancement in recent AI and machine learning (ML) research, it would be interesting to see whether ML techniques can be applied here to find better heuristics.

We also assume that the proposed collaborative algorithms are respected and followed by all the networks to achieve communal success. However, this is a fragile balance; in practice, every network can simply increase its own transmission power to benefit itself, at the expense of reduced data rate in peer networks, which causes the sum of throughput of all networks to drop. A possible future research direction is to study the dynamics of the system in the presence of non-collaborators, or assuming that even the collaborative networks have certain probability of cheating. Game theoretic results may be relevant in the analysis.

With the exponential explosion of demand in wireless communications traffic and the increasingly scarce spectrum, the throughput and spectrum efficiency can be greatly improved by a combination of the adaptive transmission techniques, collaboration across networks, and the dynamic resource allocation methods proposed in this thesis.

REFERENCES

[1] M. Zhang, A. Castillo, and B. Peleato, “Optimizing harq and relay strategies in limited feedback communication systems,” *Applied Sciences*, vol. 10, no. 21, p. 7917, 2020.

[2] M. Zhang, J. Song, D. J. Love, D. Ogbe, A. Ghosh, and B. Peleato, “Increasing throughput in wireless communications by grouping similar quality bits,” *IEEE Communications Letters*, vol. 24, no. 11, pp. 2450–2453, 2020.

[3] M. Zhang and B. Peleato, “An algorithm for routing and resource allocation among collaborative networks,” submitted to 2021 IEEE Global Communications Conference.

[4] J. Tang and X. Zhang, “Quality-of-service driven power and rate adaptation over wireless links,” *IEEE Transactions on Wireless Communications*, vol. 6, no. 8, pp. 3058–3068, 2007.

[5] J. Luo, C. Rosenberg, and A. Girard, “Engineering wireless mesh networks: Joint scheduling, routing, power control, and rate adaptation,” *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, pp. 1387–1400, 2010.

[6] F. Peng, J. Zhang, and W. E. Ryan, “Adaptive modulation and coding for IEEE 802.11 n,” in *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*, IEEE, 2007, pp. 656–661.

[7] C. U. Castellanos, D. L. Villa, C. Rosa, K. I. Pedersen, F. D. Calabrese, P.-H. Michaelsen, and J. Michel, “Performance of uplink fractional power control in UTRAN LTE,” in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, IEEE, 2008, pp. 2517–2521.

[8] B. Furht and S. A. Ahson, *Long Term Evolution: 3GPP LTE radio and cellular technology*. Crc Press, 2016.

[9] E. Uhlemann, L. K. Rasmussen, A. J. Grant, and P.-A. Wiberg, “Optimal incremental-redundancy strategy for type-ii hybrid arq,” in *IEEE International Symposium on Information Theory, Pacifico Yokohama, Yokohama, Japan, June 29-July 4, 2003*, IEEE, 2003, p. 448.

[10] E. Visotsky, Y. Sun, V. Tripathi, M. L. Honig, and R. Peterson, “Reliability-based incremental redundancy with convolutional codes,” *IEEE Transactions on communications*, vol. 53, no. 6, pp. 987–997, 2005.

[11] L. Szczecinski, S. R. Khosravirad, P. Duhamel, and M. Rahman, “Rate allocation and adaptation for incremental redundancy truncated harq,” *IEEE Transactions on Communications*, vol. 61, no. 6, pp. 2580–2590, 2013.

[12] S. M. Kim, W. Choi, T. W. Ban, and D. K. Sung, "Optimal rate adaptation for hybrid arq in time-correlated rayleigh fading channels," *IEEE transactions on wireless communications*, vol. 10, no. 3, pp. 968–979, 2011.

[13] K. D. Nguyen, L. K. Rasmussen, A. G. i Fàbregas, and N. Letzepis, "Mimo arq with multibit feedback: Outage analysis," *IEEE transactions on information theory*, vol. 58, no. 2, pp. 765–779, 2011.

[14] S. Lin, D. J. Costello, and M. J. Miller, "Automatic-repeat-request error-control schemes," *IEEE Communications magazine*, vol. 22, no. 12, pp. 5–17, 1984.

[15] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Feedback in the non-asymptotic regime," *IEEE Transactions on Information Theory*, vol. 57, no. 8, pp. 4903–4925, 2011.

[16] M.-M. Zhao, G. Zhang, C. Xu, H. Zhang, R. Li, and J. Wang, "An adaptive IR-HARQ scheme for polar codes by polarizing matrix extension," *IEEE Communications Letters*, vol. 22, no. 7, pp. 1306–1309, 2018.

[17] K. Vakilinia, S. V. Ranganathan, D. Divsalar, and R. D. Wesel, "Optimizing transmission lengths for limited feedback with nonbinary LDPC examples," *IEEE Transactions on Communications*, vol. 64, no. 6, pp. 2245–2257, 2016.

[18] K. Vakilinia, A. R. Williamson, S. V. Ranganathan, D. Divsalar, and R. D. Wesel, "Feedback systems using non-binary LDPC codes with a limited number of transmissions," in *Information Theory Workshop (ITW), 2014 IEEE*, IEEE, 2014, pp. 167–171.

[19] A. R. Williamson, T.-Y. Chen, and R. D. Wesel, "A rate-compatible sphere-packing analysis of feedback coding with limited retransmissions," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, IEEE, 2012, pp. 2924–2928.

[20] M. Jabi, A. El Hamss, L. Szczecinski, and P. Piantanida, "Multipacket hybrid ARQ: Closing gap to the ergodic capacity," *IEEE Transactions on Communications*, vol. 63, no. 12, pp. 5191–5205, 2015.

[21] S. H. Kim, S. J. Lee, and D. K. Sung, "HARQ rate selection schemes in a multihop relay network with a delay constraint," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 6, pp. 2333–2348, 2014.

[22] R. D. Wesel, K. Vakilinia, S. V. Ranganathan, and D. Divsalar, "Resource-aware incremental redundancy in feedback and broadcast," in *International Zurich Seminar on Communications*, 2016, p. 63.

[23] X. Wang, Q. Liu, and G. B. Giannakis, "Analyzing and optimizing adaptive modulation coding jointly with arq for qos-guaranteed traffic," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 2, pp. 710–720, 2007.

[24] L. Szczecinski, P. Duhamel, and M. Rahman, "Adaptive incremental redundancy for HARQ transmission with outdated CSI," in *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*, IEEE, 2011, pp. 1–6.

[25] *Lte; evolved universal terrestrial radio access (e-utra); medium access control (mac) protocol specification*, 3GPP TS 36.321, Version 12.5.0 Release 12, European Telecommunications Standards Institute, Apr. 2015.

[26] *Lte; evolved universal terrestrial radio access (e-utra); physical layer procedures*, 3GPP TS 36.213, Version 14.2.0 Release 14, European Telecommunications Standards Institute, Apr. 2017.

[27] *5g; study on new radio (nr) access technology*, 3GPP TR 38.912, Version 14.0.0 Release 14, European Telecommunications Standards Institute, May 2017.

[28] L. Vangelista and M. Centenaro, "Performance evaluation of HARQ schemes for the internet of things," *Computers*, vol. 7, no. 4, p. 48, 2018.

[29] X. Ge, Z. Li, and S. Li, "5g software defined vehicular networks," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 87–93, 2017.

[30] N.-N. Dao, M. Park, J. Kim, J. Paek, and S. Cho, "Resource-aware relay selection for inter-cell interference avoidance in 5g heterogeneous network for internet of things systems," *Future Generation Computer Systems*, vol. 93, pp. 877–887, 2019.

[31] W. Shahjehan, S. Bashir, S. L. Mohammed, A. B. Fakhri, A. Adebayo Isaiah, I. Khan, and P. Uthansakul, "Efficient modulation scheme for intermediate relay-aided iot networks," *Applied Sciences*, vol. 10, no. 6, p. 2126, 2020.

[32] G. Levin and S. Loyka, "Amplify-and-forward versus decode-and-forward relaying: Which is better?" In *22th International Zurich seminar on communications (IZS)*, Eidgenössische Technische Hochschule Zürich, 2012.

[33] K. Pang, Y. Li, and B. Vucetic, "An improved hybrid arq scheme in cooperative wireless networks," in *2008 IEEE 68th Vehicular Technology Conference*, IEEE, 2008, pp. 1–5.

[34] C.-C. Wang, D. J. Love, and D. Ogbe, "Transcoding: A new strategy for relay channels," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, IEEE, 2017, pp. 450–454.

[35] Z. Chen, T. Li, P. Fan, T. Q. Quek, and K. B. Letaief, “Cooperation in 5g heterogeneous networking: Relay scheme combination and resource allocation,” *IEEE Transactions on Communications*, vol. 64, no. 8, pp. 3430–3443, 2016.

[36] J. Metzner, “An improved broadcast retransmission protocol,” *IEEE Transactions on Communications*, vol. 32, no. 6, pp. 679–683, 1984.

[37] P. Larsson, B. Smida, T. Koike-Akino, and V. Tarokh, “Analysis of network coded harq for multiple unicast flows,” *IEEE transactions on communications*, vol. 61, no. 2, pp. 722–732, 2013.

[38] H. Zhu, B. Smida, and D. J. Love, “Optimization of two-way network coded harq with overhead,” *IEEE Transactions on Communications*, 2020.

[39] Z. Bar-Yossef, Y. Birk, T. Jayram, and T. Kol, “Index coding with side information,” *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1479–1494, 2011.

[40] N. Lee, A. G. Dimakis, and R. W. Heath, “Index coding with coded side-information,” *IEEE Communications Letters*, vol. 19, no. 3, pp. 319–322, 2015.

[41] M. C. Davey and D. MacKay, “Low-density parity check codes over GF (q),” *IEEE Communications Letters*, vol. 2, no. 6, pp. 165–167, 1998.

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

[43] IEEE, “Ieee 802.11n wireless lan medium access control mac and physical layer phy specifications.,” 2006.

[44] M. Zhang, A. Castillo, and B. Peleato, “Optimizing HARQ feedback and incremental redundancy in wireless communications,” in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2018, pp. 1–6.

[45] M. Zhang and B. Peleato, “HARQ strategies for relay systems with limited feedback,” in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, IEEE, 2019, pp. 1328–1332.

[46] Z. Li, L. Chen, L. Zeng, S. Lin, and W. H. Fong, “Efficient encoding of quasi-cyclic low-density parity-check codes,” *IEEE Transactions on Communications*, vol. 54, no. 1, pp. 71–81, 2006.

[47] Z. Wang and Z. Cui, “Low-complexity high-speed decoder design for quasi-cyclic LDPC codes,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, no. 1, pp. 104–114, 2007.

[48] P. Frenger, S. Parkvall, and E. Dahlman, "Performance comparison of HARQ with Chase combining and incremental redundancy for HSDPA," in *Vehicular Technology Conference, 2001. VTC 2001 Fall. IEEE VTS 54th*, IEEE, vol. 3, 2001, pp. 1829–1833.

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

[50] T. Richardson and R. Urbanke, *Modern coding theory*. Cambridge university press, 2008.

[51] D. P. Bertsekas, *Dynamic programming and optimal control*, 2. Athena scientific Belmont, MA, 1995, vol. 1.

[52] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2856–2867, 2014.

[53] S. Wang and B. Peleato, "Coded caching with heterogeneous user profiles," in *2019 IEEE International Symposium on Information Theory (ISIT)*, IEEE, 2019, pp. 2619–2623.

[54] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[55] A. Goldsmith, *Wireless communications*. Cambridge univ. press, 2005.

[56] TR-38.802, *Study on new radio access technology physical layer aspects*, 3GPP, 2017.

[57] A. Ghosh, "The 5g mmwave radio revolution.," *Microwave Journal*, vol. 59, no. 9, 2016.

[58] T. Wo and P. A. Hoeher, "Superposition mapping with application in bit-interleaved coded modulation," in *2010 International ITG Conference on Source and Channel Coding (SCC)*, IEEE, 2010, pp. 1–6.

[59] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over GF(q)," *IEEE Transactions on Communications*, vol. 55, no. 4, pp. 633–643, 2007.

[60] J. Du, L. Yang, J. Yuan, L. Zhou, and X. He, "Bit mapping design for ldpc coded bicm schemes with multi-edge type exit chart," *IEEE Communications Letters*, 2016.

[61] L. Gong, L. Gui, B. Liu, B. Rong, Y. Xu, Y. Wu, and W. Zhang, "Improve the performance of LDPC coded QAM by selective bit mapping in terrestrial broadcasting system," *IEEE transactions on broadcasting*, vol. 57, no. 2, pp. 263–269, 2011.

[62] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Dispersion of gaussian channels," in *2009 IEEE International Symposium on Information Theory*, IEEE, 2009, pp. 2204–2208.

[63] A. B. Flores, R. E. Guerra, E. W. Knightly, P. Ecclesine, and S. Pandey, “Ieee 802.11 af: A standard for tv white space spectrum sharing,” *IEEE Communications Magazine*, vol. 51, no. 10, pp. 92–100, 2013.

[64] M. M. Sohul, M. Yao, T. Yang, and J. H. Reed, “Spectrum access system for the citizen broadband radio service,” *IEEE Communications Magazine*, vol. 53, no. 7, pp. 18–25, 2015.

[65] M. McHenry, K. Steadman, A. E. Leu, and E. Melick, “Xg dsa radio system,” in *2008 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, IEEE, 2008, pp. 1–11.

[66] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Transactions on information theory*, vol. 46, no. 2, pp. 388–404, 2000.

[67] J. Li, C. Blake, D. S. De Couto, H. I. Lee, and R. Morris, “Capacity of ad hoc wireless networks,” in *Proceedings of the 7th annual international conference on Mobile computing and networking*, 2001, pp. 61–69.

[68] O. Goussevskaia, R. Wattenhofer, M. M. Halldórsson, and E. Welzl, “Capacity of arbitrary wireless networks,” in *IEEE INFOCOM 2009*, IEEE, 2009, pp. 1872–1880.

[69] M. Dinitz, “Distributed algorithms for approximating wireless network capacity,” in *2010 Proceedings IEEE INFOCOM*, IEEE, 2010, pp. 1–9.

[70] M. Andrews and M. Dinitz, “Maximizing capacity in arbitrary wireless networks in the sinr model: Complexity and game theory,” in *IEEE INFOCOM 2009*, IEEE, 2009, pp. 1332–1340.

[71] L. Xiao, M. Johansson, and S. P. Boyd, “Simultaneous routing and resource allocation via dual decomposition,” *IEEE Transactions on Communications*, vol. 52, no. 7, pp. 1136–1144, 2004.

[72] D. P. Palomar and M. Chiang, “Alternative distributed algorithms for network utility maximization: Framework and applications,” *IEEE Transactions on Automatic Control*, vol. 52, no. 12, pp. 2254–2269, 2007.

[73] L. Fu, S. C. Liew, and J. Huang, “Fast algorithms for joint power control and scheduling in wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 3, pp. 1186–1197, 2010.

[74] H. Tabrizi, B. Peleato, G. Farhadi, J. M. Cioffi, and G. Aldabbagh, “Spatial reuse in dense wireless areas: A cross-layer optimization approach via admm,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 12, pp. 7083–7095, 2015.

[75] Q. C. Li, R. Q. Hu, Y. Xu, and Y. Qian, “Optimal fractional frequency reuse and power control in the heterogeneous wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2658–2668, 2013.

[76] K. Shen and W. Yu, “Fplinq: A cooperative spectrum sharing strategy for device-to-device communications,” in *2017 IEEE International Symposium on Information Theory (ISIT)*, IEEE, 2017, pp. 2323–2327.

[77] K. Shen and W. Yu, “Fractional programming for communication systems—part i: Power control and beamforming,” *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2616–2630, 2018.

[78] K. Shen and W. Yu, “Fractional programming for communication systems—part ii: Uplink scheduling via matching,” *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2631–2644, 2018.

[79] N. Naderializadeh and A. S. Avestimehr, “Itlinq: A new approach for spectrum sharing in device-to-device communication systems,” *IEEE journal on selected areas in communications*, vol. 32, no. 6, pp. 1139–1151, 2014.

[80] X. Yi and G. Caire, “Optimality of treating interference as noise: A combinatorial perspective,” *IEEE Transactions on Information Theory*, vol. 62, no. 8, pp. 4654–4673, 2016.

[81] X. Cao, R. Ma, L. Liu, H. Shi, Y. Cheng, and C. Sun, “A machine learning-based algorithm for joint scheduling and power control in wireless networks,” *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4308–4318, 2018.

[82] R. M. Dreifuerst, S. Daulton, Y. Qian, P. Varkey, M. Balandat, S. Kasturia, A. Tomar, A. Yazdan, V. Ponnampalam, and R. W. Heath, “Optimizing coverage and capacity in cellular networks using machine learning,” *arXiv preprint arXiv:2010.13710*, 2020.

[83] B. Bojović, E. Meshkova, N. Baldo, J. Riihijärvi, and M. Petrova, “Machine learning-based dynamic frequency and bandwidth allocation in self-organized lte dense small cell deployments,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, pp. 1–16, 2016.

[84] W. Cui, K. Shen, and W. Yu, “Spatial deep learning for wireless scheduling,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1248–1261, 2019.

[85] J. Yu, K. Yu, D. Yu, W. Lv, X. Cheng, H. Chen, and W. Cheng, “Efficient link scheduling in wireless networks under rayleigh-fading and multiuser interference,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 8, pp. 5621–5634, 2020.

[86] X. Lv, A. Xiong, S. Zhang, and X.-s. Qiu, “Vcg-based bandwidth allocation scheme for network virtualization,” in *2012 IEEE Symposium on Computers and Communications (ISCC)*, IEEE, 2012, pp. 000 744–000 749.

[87] B. Cao, Y. Li, C. Wang, G. Feng, S. Qin, and Y. Zhou, “Resource allocation in software defined wireless networks,” *IEEE Network*, vol. 31, no. 1, pp. 44–51, 2017. DOI: 10.1109/MNET.2016.1500273NM.

[88] M. Yang, Y. Li, D. Jin, J. Yuan, L. Su, and L. Zeng, “Opportunistic spectrum sharing based resource allocation for wireless virtualization,” in *2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, IEEE, 2013, pp. 51–58.

[89] T. C.-Y. Ng and W. Yu, “Joint optimization of relay strategies and resource allocations in cooperative cellular networks,” *IEEE Journal on Selected areas in Communications*, vol. 25, no. 2, pp. 328–339, 2007.

[90] P. Tilghman, “AI will rule the airwaves: A DARPA grand challenge seeks autonomous radios to manage the wireless spectrum,” *IEEE Spectrum*, vol. 56, no. 6, pp. 28–33, 2019.

VITA

Mai Zhang received his B.Eng. in Electronic Engineering from The Hong Kong University of Science and Technology with First Class Honors in 2016. He is currently pursuing his Ph.D. degree in Electrical and Computer Engineering at Purdue University. His research interests involve wireless communications, HARQ, and the application of AI in communication systems.