2012-09-06

Two Applications of Physical Layer Network Coding in Multi-hop Wireless Networks

Zhou, Ruiting

UNIVERSITY OF CALGARY

Two Applications of

Physical Layer Network Coding in Multi-hop Wireless Networks

by

Ruiting Zhou

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

CALGARY, ALBERTA

August, 2012

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Two Applications of Physical Layer Network Coding in Multi-hop Wireless Networks" submitted by Ruiting Zhou in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE.

Dr. Zongpeng Li
Supervisor, Department of
Computer Science

Dr. Majid Ghaderi
Department of Computer Science

Dr. Carey Williamson
Co-supervisor, Department of
Computer Science

Dr. John Nielsen
Department of Electrical and
Computer Engineering

Date

Abstract

Physical layer network coding (PNC) is a relatively new technique that can perform network coding at the physical layer to boost the capacity of wireless ad hoc networks. By viewing overlapping data transmissions as their linear combinations, PNC can potentially achieve large improvement in physical-layer throughput over traditional transmissions and digital network coding at the higher layers. While existing research on PNC usually focuses on simple network topologies (e.g, the two-way relay channel), it appears natural and promising to further explore the opportunities of applying PNC in a large, general, multi-hop wireless network. This thesis covers two endeavours along this direction.

Firstly, we show how PNC can be combined with signal alignment (SA), another technique inspired from interference alignment (IA), for application in MIMO wireless networks. PNC coupled with SA (PNC-SA) has the potential of fully exploiting the precoding space at the senders, and can better utilize the spatial diversity of a MIMO network for higher transmission rates, outperforming existing techniques including MIMO or PNC alone, interference alignment (IA), and interference alignment and cancelation (IAC). We study the optimal precoding and power allocation problem of PNC-SA, for SNR maximization at the receiver. The mapping from SNR to BER is then analyzed, revealing that the throughput gain of PNC-SA does not come with a sacrifice in BER. Furthermore, the maximum throughput for the general N-user M-antenna uplink system is presented. We also demonstrate general applications of PNC-SA beyond a multi-user wireless uplink, and show via network level simulations that it can substantially increase the throughput of unicast and multicast sessions, by opening previously unexplored solution spaces in multi-hop MIMO routing.

Secondly, we focus on routing algorithm design in NanoNets, which are networks of nanomachines at extremely small dimensions, on the order of nanometers or micrometers.

Based on the salient features of a NanoNet, including low node cost and very low available power, we propose a new routing paradigm for unicast and multicast data transmission in NanoNets. Our design, termed Buddy Routing (BR), is enabled by PNC, and argues for pair-to-pair data forwarding in place of traditional point-to-point data forwarding. Through both analysis and simulations, we compare BR with point-to-point routing, in terms of raw throughput, error rate, energy efficiency, and protocol overhead, and show the advantages of BR in NanoNets.

Acknowledgements

First and foremost, I would like to sincerely thank my supervisor, Dr. Zongpeng Li. I cannot express my appreciation and thanks enough, how this thesis would not have been possible without him. His guidance, advice and support helped this thesis to come to light. Zongpeng sparked my interest in the area of physical layer network coding and gave me the opportunity to work in this area. I was so grateful for his innovative ideas and consistent precious feedback throughout this work. Moreover, Zongpeng has not only been my teacher, but also been my mentor and advisor in every aspects of my life. He helped me to adapt to the new environment in Calgary when I came, and pointed out the direction for my future career. Working together with Zongpeng has been the best experience that I will cherish for ever.

I would also like to thank my co-supervisor, Dr. Carey Williamson for his continual support and guidance. I am truly fortunate to have the opportunity to learn from Carey. He set me an example of being a scientist. His immense knowledge and strong sense of responsibility impressed me very much. I am also grateful for my friends and colleagues in the computer science department. They gave me a good time in Calgary. I would like to express my heartfelt thanks to my family, my mother and my father. Although they are thousands of miles away, their endless love, support and encouragement provided the fuel for me to complete the graduate study. Last but not the least, I would like to dedicate these words to my soul mate, Bing. "How many can say they have a true partner, friend, and lover that they can count on in good times and bad? Only a lucky few, like me."

Table of Contents

List of Tables

List of Figures

List of Acronyms

AP Access Point

A&F Amplify&Forward

AODV Ad hoc On-demand Distance Vector

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BP Belief Propagation

BPSK Binary Phase-Shift Keying

BPF Band Pass Filter

BR Buddy Routing

CSI Channel State Information

DNC Digital Network Coding

DoF Degree of Freedom

DSR Dynamic Source Routing

LPF Low Pass Filter

GPSR Greedy Perimeter Stateless Routing

IA Interference Alignment

IAC Interference Alignment and Cancelation

MAC Media Access Control

MIMO Multiple-Input and Multiple-Output

ML Maximum Likelihood

PNC Physical Layer Network Coding

PNC-SA Physical Layer Network Coding with Signal Alignment

QPSK Quadrature Phase-Shift Keying

QAM Quadrature Amplitude Modulation

SA Signal Alignment

SNR Signal-to-Noise Ratio

TDMA Time Division Multiple Access

ZF Zero Forcing

Chapter 1

Introduction

1.1 Motivation

Network coding is an elegant transmission technique proposed by Ahlswede *et al.* [1] in 2000. It can be applied to improve network throughput and robustness, by having intermediate nodes send out packets that are combinations of previously received information. Recently, network coding has generated substantial interest in the field of wireless communication. A successful extension of network coding to the wireless paradigm, at the physical layer, is physical layer network coding (PNC) [41]. PNC is seminal in that it exploits the natural additive nature of electro-magnetic waves in space. Viewing collided transmissions simply as superimposed signals, PNC applies tailored demodulation for translating them into linear combinations of transmitted data packets. Such demodulated linear combinations, similar to encoded packets in network coding [1], are then used to facilitate further data routing.

Most existing research on PNC focuses on simple network topologies, as exemplified by the two-way relay channel. For example, Zhang *et al.* [41] used the 3-node Alice-and-Bob example (the two-way relay channel) to show how a PNC-demodulation algorithm can be implemented at the relay, to extract the digital version of p_a (packet from Alice) xor p_b (packet from Bob). Zhang and Liew further studied PNC in the Alice-and-Bob scenario with two antennas at the relay [39]. They examined how the two different combinations at the relay can be exploited to improve the BER of PNC. Other papers discussed the optimal coding and decoding design [23, 38] or the synchronization issues in PNC [40], also in simple network topologies.

Multi-hop wireless networks have been a focal point of research for over a decade. In such a network, a packet may traverse multiple consecutive hops to reach its destination [27]. The term "wireless ad hoc network" has also been used in the literature, with a slight emphasis on the fact that a wireline backhaul infrastructure is not needed and rapid network deployment is feasible. Multi-hop forwarding naturally extends network coverage. It may also enhance the transmission throughput, by using shorter hops. Nowadays, with technology and engineering advances in wireless communication, applications of multi-hop routing are witnessed not only in mobile ad-hoc networks, wireless mesh networks and wireless sensor networks, but also in multi-input and multi-output (MIMO) wireless networks and nanonetworks. A large volume of works are published on routing protocol design, packet scheduling, and power control aspects for multi-hop wireless networks [6, 27, 34]. Yet little existing work discusses how PNC can be used in multi-hop networks to improve the network capacity and to facilitate the routing protocol design.

It appears promising and interesting to explore the opportunities of applying PNC in a large, general, multi-hop wireless network. This thesis presents two of our endeavors along this direction.

1.2 Research Objective

The main objective of this thesis is to study the potential and applications of PNC in a multi-hop wireless network, for improving network performance. We aim to identify suitable network scenarios where PNC appears promising in enabling new solutions, design and analyze such solutions, and compare them with previous ones. More specifically, we are interested in the following two directions:

- How can we apply PNC in multi-input and multi-output (MIMO) wireless networks,

to increase the network throughput?

In the first half of the thesis, we propose a new physical layer technique, signal alignment, and show how it can be combined with PNC for application in a MIMO network. The new PNC-SA scheme can outperform previous techniques including basic MIMO, interference alignment, and interference alignment and cancellation in a number of scenarios.

- Is it possible to design a new PNC-based routing algorithm for multi-hop wireless networks consisting of extremely small and power-limited nodes?

  Buddy Routing (BR), a PNC-based routing paradigm for NanoNets, is presented in the second half of this thesis. It explores the design space of PNC-enabled pair-to-pair forwarding, and compares that with traditional point-to-point routing algorithms, for both unicast and multicast transmissions.

1.3 Summary of Contributions

PNC-SA: Physical Layer Network Coding with Signal Alignment for MIMO Networking

Multi-input and multi-output (MIMO) is a new physical layer technology in modern wireless communication. It employs multiple antennas at both transmitter and receiver sides to improve the communication performance. By exploiting the spatial diversity, MIMO systems provide a number of advantages over traditional single-input and single-output (SISO) communication [33]. Hence, it is emerging as a natural choice for future wireless networks. However, the throughput of MIMO systems is fundamentally limited by the number of antennas per node. Gollakota *et al.* [10] presented interference alignment and cancellation (IAC) for the scenario of multi-user MIMO uplink transmission with limited receiver collaboration, for mitigating this limitation. One of the receivers

has its 'interferences' aligned, and one or more original packets demodulated (IA). The demodulated packets are then sent in digital form to another receiver to help further decoding (IC). Li *et al.* [21] studied the application of IAC in more general, multi-hop wireless networks. Inspired by PNC and IAC, we propose signal alignment (SA), and show how PNC can be combined with it for application in MIMO wireless networks. The main contributions include:

- A new technique, signal alignment (SA), is proposed, for enabling PNC in MIMO wireless networks. Through calculated precoding at the senders, the number of dimensions spanned by signals arriving at a receiver is reduced to exactly match its receive diversity. Consequently, the receiver can decode linear combinations of the transmitted packets. It provides a large throughput gain over existing techniques including MIMO, PNC, and IAC.

- We study the SNR-maximization through precoding at the sender side, and the decoding and demodulation method at the receiver side. The BER performance of PNC-SA and IAC are analyzed and compared. Numerical results show that the BER of PNC-SA is slightly better than that of IAC.

- We prove two theorems on the maximum throughput achievable in a general $N \times N \times M$ MIMO uplink scenario. Here N is the number of clients at the transmitter side, as well as the number of APs at the receiver side, and each node is equipped with M antennas.

- We demonstrate more general applications of PNC-SA in multi-hop wireless networks, beyond a MIMO uplink with limited receiver collaboration. Through packet level simulation studies, we demonstrate that PNC-SA can substantially improve the throughput of unicast and multicast sessions, by opening a new solution space in multi-hop MIMO routing.

Buddy Routing: A PNC Based Routing Paradigm for NanoNets

Recent advances in micro-electro-mechanical systems (MEMS), including nanotechnology and digital electronics, have enabled the development of low-cost, low-power, multi-functional nodes with remarkably small form factor, which can communicate over short distances. Applications include wireless sensor networks consisting of small wireless sensors and nanonetworks consisting of even smaller nanonodes. Nanonetworks are starting to attract attention in the research community. They can lead to new applications in biomedical, environmental, and other industrial fields [2]. The salient features of nanonodes and nanonetworks invite new networking solutions to be designed. In the second part of this thesis, we focus on the design of multi-hop routing algorithms in NanoNets. Our proposed routing algorithm, Buddy Routing, is a pair-to-pair routing scheme based on PNC. Our detailed contributions include:

- We propose Buddy Routing (BR), a PNC-based pair-to-pair routing algorithm for unicast and multicast data transmission in NanoNets. We compare two technologies that can enable BR, PNC and Amplify&Forward, and show the advantages of PNC.

- We calculate the capacity and the power consumption of BR through theoretical analysis. Compared with traditional point-to-point routing, we show that the extra power consumption overhead of BR is below 20%, while BR provides a potential capacity gain of a factor of 2.

- A pair-to-pair greedy geographic unicast routing algorithm is designed. Iterative MAC layer optimization, over both transmit power at nanonodes and lengths of time slots in a TDMA MAC are refined, for mitigating bottleneck interference and improving end-to-end route capacity. Simulation results verify the theoretical analysis that BR has a significant throughput gain over traditional point-to-point routing.

- We extend the solution design from multi-hop unicast to multi-hop multicast, by designing a pair-forwarding based multicast tree construction algorithm, and adapting the iterative MAC optimization algorithm from a unicast path to a multicast tree. A two fold increase in multicast throughput is observed in large scale network simulations.

1.4 Thesis Organization

The rest of this thesis is organized as follows. Chapter 2 introduces the fundamental knowledge and background information related to this thesis. We first explain the concept of physical layer network coding, using the two-way relay channel (a three-node linear wireless network) to illustrate the PNC modulation and demodulation mapping. Then some related work on MIMO technologies in an uplink communication scenario is provided, including basic MIMO and interference alignment and cancelation (IAC). We also use the same communication system to show the idea of our PNC-SA scheme. Moreover, we provide some general introduction to NanoNets, and study two cooperative data forwarding schemes among paired nanonodes, PNC and Amplify & Forward (A&F). Some representative routing algorithms for multi-hop wireless networks, including DSR, AODV, and GPSR, are reviewed at the end of this chapter.

Chapter 3 is dedicated to PNC-SA. We outline the system model and assumptions in the first section of this chapter. A detailed PNC-SA scheme design is presented, including the precoding optimization problem at the client side and the decoding process at the AP side. The SNR-BER performance of PNC-SA is then analyzed, and compared to that of IAC. Throughput analysis in the N-client N-AP system is conducted in Sec. 3.4. The application of PNC-SA is not limited to scenarios of limited receiver collaboration; we show general applications of PNC-SA in multi-hop MIMO networks, for routing tasks

including information exchange, unicast, and multicast/broadcast.

Chapter 4 presents our Buddy Routing protocol for nanonetworks. A comprehensive comparison between PNC and A&F is provided at the beginning of this chapter. Then the capacity and power efficiency of a BR route are studied through theoretical analysis. Subsequently, we extend the geographical greedy routing algorithm [18] to its pair-to-pair forwarding version, for computing a BR unicast route and a BR multicast tree, respectively. Simulation results verify the theoretical analysis that BR has a potential to substantially improve the end-to-end throughput over traditional point-to-point routing.

In Chapter 5, we conclude this thesis by summarizing our work and discussing directions for future research.

Chapter 2

Background and Related Work

This thesis focuses on how we can apply physical layer network coding (PNC) in a general multi-hop wireless network. It will therefore be advantageous to first familiarize ourselves with some of the fundamental knowledge behind PNC and multi-hop wireless networks. In this chapter, we will first provide an introduction to physical layer network coding, and illustrate how it works in a three-node linear network. Then, Sec. 2.2 describes a general multi-input multi-output (MIMO) uplink scenario that motivates our work and shows how basic MIMO, interference alignment and cancelation (IAC), and PNC-SA can be applied in this MIMO network. Background information on nanonetworks is included in Sec. 2.3, while Sec. 2.4 presents two collaborative data forwarding techniques among paired nodes, Amplify&Forward (A&F) and PNC. Finally, some multi-hop routing algorithms designed for wireless networks are reviewed in Sec. 2.5.

2.1 Physical Layer Network Coding

Zhang *et al.* [41] initiated the study of physical layer network coding in the three-node linear network, as shown in Fig. 2.1. Each node is equipped with a single omni-directional antenna, and the channel is half duplex so that one node cannot transmit and receive packets in the same time slot. With the help of the relay node (Node C) in the middle, Node A and Node B are able to exchange information.

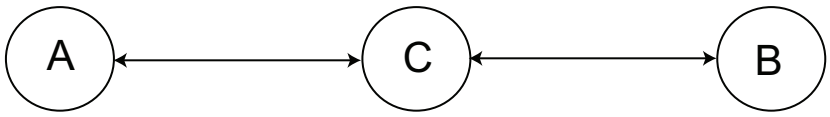

Figure 2.1: A three-node linear network, *a.k.a.* the two-way relay channel.

Using a traditional transmission scheduling scheme, in order to avoid interference, a total of four time slots are needed for the exchange of two packets between nodes A and B. Node A first sends its packet to Node C, then Node C relays it to Node B. After that, Node B transmits its packet in the reverse direction, first from Node B to Node C then from Node C to Node A.

2.1.1 A Digital Network Coding Based Scheme

If we apply (digital) network coding, the resulting transmission scheme can be more efficient, with a total of three time slots required [8, 36]. As shown in Fig. 2.2, first, Node A sends packet S_A to Node C. Next, Node B transmits packet S_B to Node C. After successfully decoding S_A and S_B, Node C constructs an encoded packet $S_C = S_A \oplus S_B$. Here $\oplus$ denotes the bit-wise exclusive-OR operation. The relay node C then broadcasts S_C to both directions. When Node A receives S_C, it can decode S_B from S_C using its own packet S_A as follows:

$$S_A \oplus (S_C) = S_A \oplus (S_A \oplus S_B) = S_B$$

Similarly, Node B can extract S_A using S_B.

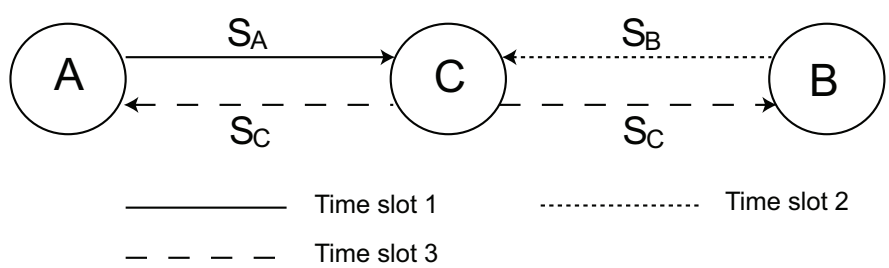

Figure 2.2: A transmission scheme based on digital network coding.

2.1.2 Physical-Layer Network Coding (PNC)

We next introduce PNC, whose transmission steps are illustrated in Fig. 2.3. In time slot 1, Node A and Node B transmit their respective packets S_A and S_B to Node C

simultaneously. Through PNC demodulation, Node C can decode $S_C = S_A \oplus S_B$. In the second time slot, Node C broadcasts packet S_C to both Node A and Node B. Node A and Node B each can extract the packet they want from what they receive. Note that these three packets are all the same size.

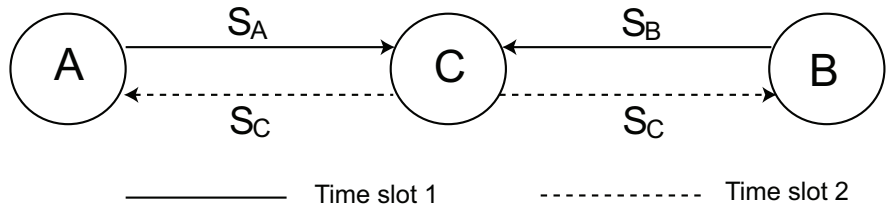

Figure 2.3: A transmission scheme based on physical layer network coding.

PNC employs a proper modulation-and-demodulation technique at the relay node, maps additions of E-M signals to GF(2^n) additions of digital bit signals, so that the interference becomes "addition performed by nature". We assume the use of BPSK modulation during the transmission to introduce PNC mapping. The background information about digital modulation techniques, including BPSK, QPSK, and 16QAM, can be found in Chapter 3 of Ref. [5]. The important assumptions made by Zhang *et al.* [41] are symbol-level and carrier-phase synchronization, and the use of power control, so that the packets from Node A and Node B arrive at Node C with the same phase and amplitude. The combined signal received by the relay node during one symbol period is

$$\begin{aligned} r_C(t) &= s_A(t) + s_B(t) \\ &= a_A \cos(\omega t) + a_B \cos(\omega t) \\ &= (a_A + a_B) \cos(\omega t) \end{aligned}$$

where $s_A(t)$ and $s_B(t)$ are the passband signals transmitted by Node A and Node B, respectively; $r_C(t)$ is the signal received by relay node C; ω is the carrier frequency; a_i, $i = A$ or B, is the BPSK modulated information bit. Node C receives a baseband signal $R = a_A + a_B$. From the combined signal, Node C cannot recover the individual

information a_A and a_B transmitted by Node A and Node B. However, the required function of the relay node is just forwarding necessary information to Node A and Node B for extracting a_A and a_B. Through PNC mapping, Node C can obtain the equivalence of GF(2) summation of bits from Node A and Node B at the physical layer.

Table 2.1: PNC Mapping: Modulation Mapping at Node A, Node B; Demodulation and Modulation Mappings at Node C

Modulation mapping at Node A and Node B				Demodulation mapping at Node C		
				Input	Output	
Input		Output			Modulation mapping at Node C	
					Input	output
s_A	s_B	a_A	a_B	$a_A + a_B$	s_C	a_C
1	1	1	1	2	0	-1
0	1	-1	1	0	1	1
1	0	1	-1	0	1	1
0	0	-1	-1	-2	0	-1

Table 2.1 shows the details of the PNC mapping scheme. Here $s_j \in \{0, 1\}, j \in \{A, B, C\}$ are the variables representing the data bit of Node A, B, and C. $a_j \in \{-1, 1\}$ is a variable representing the BPSK modulated bit of s_j such that $a_j = 2s_j - 1$. From this table, we can see that Node C can obtain the bit $s_C = s_A \oplus s_B$, then the signal

$$s_C(t) = a_C \cos(\omega t)$$

is transmitted. Node A and Node B can decode s_C via normal BPSK demodulation to recover the digital version of S_C. To summarize, the digital network coding operation $S_C = S_A \oplus S_B$ can be achieved through PNC mapping, while saving a time slot.

2.2 MIMO Technology

New physical layer techniques and their applications in wireless routing have been active areas of research in the recent past. A salient example is multi-input and multi-output (MIMO) communication. A MIMO link employs multiple transmit and receive antennas

that operate over the same wireless channel. MIMO transmission brings extra spatial diversity that can be exploited to break through capacity limits inherent in single-input and single-output (SISO) channels [10, 33]. It has a potential to offer significant improvement in data throughput and link range without additional bandwidth or increased per-antenna transmit power.

We propose signal alignment (SA), a new technique that enables PNC in wireless networks consisting of MIMO links. The idea and benefit of PNC-SA can be illustrated in an uplink communication scenario, designed to motivate interference alignment and cancelation (IAC) [10], a recent technique for improving throughput in MIMO networks. PNC-SA provides a further throughput gain of 33% over IAC, under high SNR in the 2-client 2-AP MIMO system.

Fig. 2.4 depicts the MIMO uplink from two clients to two APs. Each node is equipped with 2 antennas that operate on the same channel, with flat Rayleigh fading [10, 33]. During propagation, a signal experiences amplitude attenuation and phase shift, which can be modeled using a complex number. $\mathbf{H_{ij}}$ is the 2×2 complex matrix for the channel gains from client i to AP j. An Ethernet link connects the two APs, enabling limited collaboration: digital packets can be exchanged, but not analog ones [10]. The goal is to send packets from the clients to the APs as fast as possible. Note that the Ethernet traffic is comparable to the wireless throughput if the APs communicate digital packets. In contrast, analog packets are too large to transmit, because to capture an analog signal without loss one needs to sample it at twice its bandwidth, and each sample is about 8-bit long.

2.2.1 Basic MIMO

A naive solution uses one send-receive antenna pair to avoid any interference at all. Let's normalize a time unit to be one packet transmission time. Here, basic MIMO refers to a

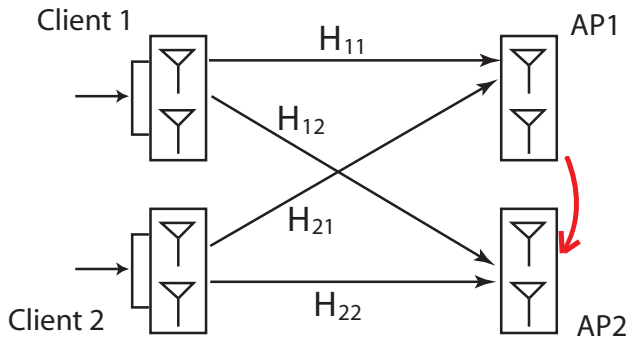

Figure 2.4: The 2-client 2-AP MIMO uplink.

MIMO uplink from one transmitter to one receiver, and each node is equipped with the same number of antennas. The word "basic" here is intended to contrast with PNC-SA, which is a MIMO mechanism enhanced with physical layer network coding as well as signal alignment. For a quick improvement, we can use a 2×2 MIMO link formed by a client-AP pair, to transmit two packets, x_1 and x_2, simultaneously. As shown in Fig. 2.5, the AP receives two overlapped signals y_1 and y_2 of x_1 and x_2. Here h_{ij} is the channel coefficient characterizing channel fading from antenna i at the client side to antenna j at the AP side, which includes amplitude attenuation and phase shift. $y_1 = h_{11}x_1 + h_{21}x_2$ and $y_2 = h_{12}x_1 + h_{22}x_2$. ML or ZF detection can be applied to recover x_1 and x_2, increasing the throughput from 1 packet per time unit to 2 packets per time unit.

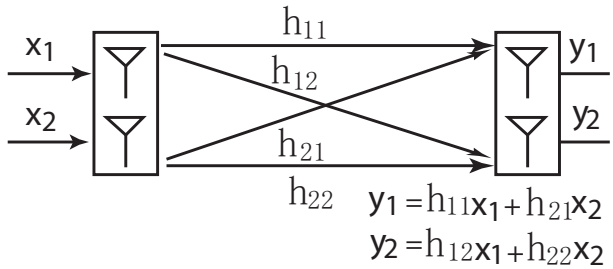

Figure 2.5: Basic MIMO achieves a throughput of 2 packets per time unit.

Can we utilize all available antennas to form a 4×4 MIMO link, to transmit >2 packets? The answer, unfortunately, is 'no'. Since the four receive antennas are distributed at two nodes, we do not have all four received analog signals at one location, as required

in MIMO decoding. Therefore, the throughput of all practical MIMO LANs is limited by the number of antennas per AP.

2.2.2 Interference Alignment and Cancelation (IAC)

IAC breaks through this bottleneck by combining interference alignment (IA) and interference cancelation (IC) techniques. It allows three concurrent packets to be transmitted by the clients and decoded at the AP side. Two properties of MIMO LANs are exploited by IAC. One is that MIMO transmitters can control the direction of their signals at a receiver. Another is the existence of a backend Ethernet connecting the two APs, enabling limited collaboration between them. Thus, in IAC, the two clients encode their transmissions in a calculated way to align the second and the third packets at AP1 but not at AP2. As shown in Fig. 2.6, IAC first performs precoding over 3 packets x_1, x_2 and x_3 at the clients, such that x_2 and x_3 arrive along the same direction at AP1. *Direction* here is an abstract concept defined as a signal's encoding vector when received at AP1. AP1 has two equations and two unknowns, from which it can solve for x_1. Next, AP1 transmits x_1 in digital format to AP2, through the Ethernet link between them. AP2 subtracts the component of x_1 from its received signals, leaving it with two equations and two unknowns, from which it recovers x_2 and x_3.

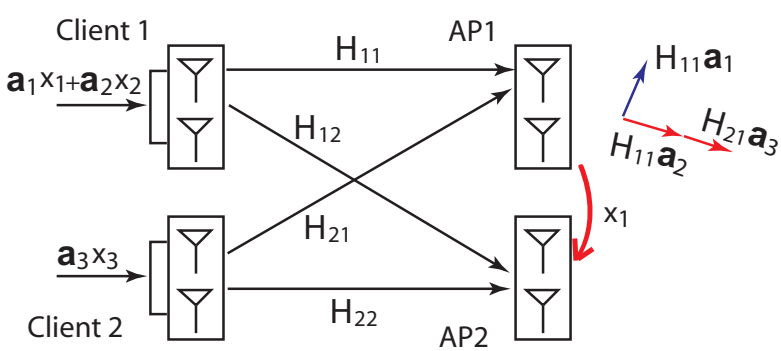

Figure 2.6: IAC achieves a throughput of 3 packets per time unit. Each $\mathbf{a_i}$ is a 2×1 precoding vector. $\mathbf{H_{11}a_1}$ is called the *direction* of x_1 at AP1.

Can we use IAC to transmit 4 packets in one time unit instead of 3? The answer is 'no'. With IAC, the intended signal has to take its own direction at AP1, while all other 'interferences' take another. As a result, the two packets from client 2 have to be aligned to the same direction at AP1. This requires identical precoding vectors for them, making them impossible to separate at AP2.

2.2.3 Physical Layer Network Coding with Signal Alignment (PNC-SA)

Departing from such a requirement of IA and IAC (each signal has to take a unique direction and be demodulated in uncoded form), SA allows multiple *signals* to be aligned to the same direction at a receiver. In fact, there is no *interference* in SA; all packet transmissions are treated as *signals*. As shown in Fig. 2.7, PNC-SA simultaneously transmits 4 same-sized packets, $x_1, \ldots, x_4$. Precoding is performed such that at AP1, x_1 and x_3 are aligned to the same direction, and the same for x_2 and x_4. AP1 has two equations and two unknowns (x_1+x_3, x_2+x_4), from which it solves x_1+x_3, x_2+x_4 to transmit in digital format to AP2, through the Ethernet link. Having accumulated 4 equations (two digital, two analog), AP2 then solves them to recover the 4 original packets, x_1, x_2, x_3, and x_4.

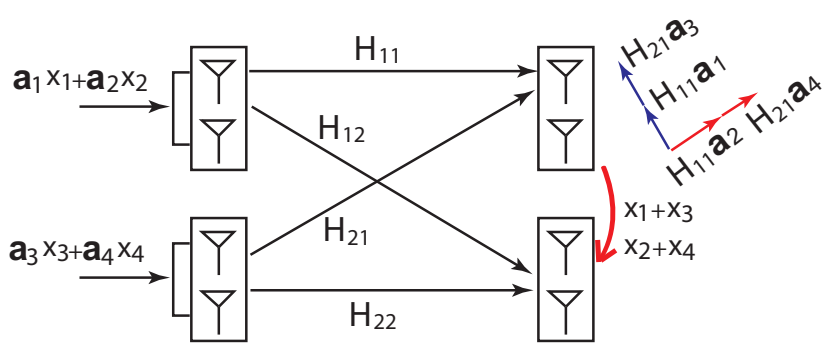

Figure 2.7: PNC-SA can achieve a throughput of 4 packets per time unit.

Two ideas work in concert in the PNC-SA solution. One is *demodulating a linear combination*, adapted from PNC. The other is *precoding at the sender for alignment at*

the receiver, inspired by IA. PNC-SA helps the exploration of the full precoding space at the senders, and the full spatial diversity of the system. As we will show, PNC and IAC can indeed be viewed as special cases of PNC-SA. When each node has a single receive diversity (one antenna per node), SA degrades into phase synchronization [40, 41], and PNC-SA degrades into PNC. With extra restrictions on precoding and decoding, PNC-SA degrades into IAC.

2.3 Introduction to Nanonetworks

Nanonetworks represent an emerging type of wireless sensor network consisting of nanonodes — wireless nodes at extremely small form factors, on the order of micrometers or nanometers. Nanonetworks are expected to expand the capabilities of single nanomachines both in terms of complexity and range of operation by allowing them to coordinate, share and fuse information. Some new applications of nanotechnology are enabled by nanonetworks in the biomedical field, environmental research, military technology, and industrial and consumer goods applications.

As shown in Fig. 2.8, the structure of a nanonode resembles that of a wireless sensor node to a great extent. Recent advances in physics and engineering technologies have made it possible to manufacture storage, processor, radio antenna and power supply at the nano-scale [3, 15]. For example, a typical nanotube based transmitter has a volume of 3.9×10^4 nm^3 [35]. Electro-magnetic communication between nanonodes can be enabled by either frequency modulation or phase modulation. Such invisibly small nanonodes can be easily attached to everyday objects or human bodies, for sensing antigen molecules, the immune system, or other physical parameters of interest.

Compared with a wireless mesh network and a 'regular' wireless sensor network, a NanoNet has a number of salient features. Nanotube radiation is at Terahertz domain,

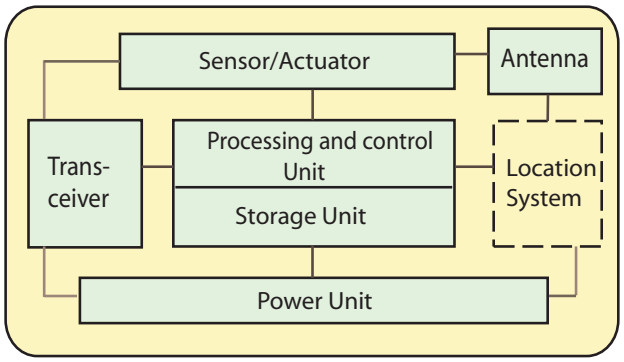

Figure 2.8: The architecture of a nanonode.

leading to wavelengths on the order of 0.1 mm, and usually travels in line-of-sight fashion. Nano-processors, nano-transceivers and nano-power supply are usually orders of magnitude weaker than their counterparts in wireless mesh networks. Due to limitations in nano-battery technologies, power supply is weak and short-lived, *e.g.*, providing current in the order of $45\mu Ah^{-1}cm^{-2}\mu m^{-1}$, and requiring periodical recharges [3, 4]. Consequently, direct nano communication can only happen over very short distances, and at very low rates. In short, NanoNets present an entirely new networking paradigm that invites radical revolutions in networking solutions, including error detection/correction, routing, and medium access control (MAC) algorithms [2].

By grouping nodes into collaborating pairs, pair-to-pair forwarding can overcome the fundamental nodal power constraint, enhancing the communication range and rate of nanonodes, and is therefore a promising paradigm for the routing algorithm design for NanoNets. Such routing algorithms are best coupled with a simple MAC algorithm, such as TDMA, so that execution on nano processors does not become a bottleneck.

2.4 Collaborative Data Forwarding

In Sec. 2.3, we mentioned that nanonodes can be grouped into collaborating pairs. There are two different physical layer techniques that can enable collaborative data forwarding

among paired nanonodes: amplify&forward (A&F), or physical layer network coding (PNC). A detailed comparison between the two, in terms of error rate and capacity, is provided in Sec. 4.1.

2.4.1 Pair-to-Pair Forwarding: A&F

The original A&F technique in cooperative transmission is used in a three-node relay network [32]. In order to improve or maximize total network channel capacities, the relay station would amplify the received signal from the source node and then forward it to the destination station. By combining the source and relay transmissions, and depending on the relaying protocol used, the destination can achieve diversity against fading without the use of an antenna array at any terminal [32]. A number of virtual MIMO forwarding schemes recently proposed are in essence based on A&F-enabled collaboration [14, 25].

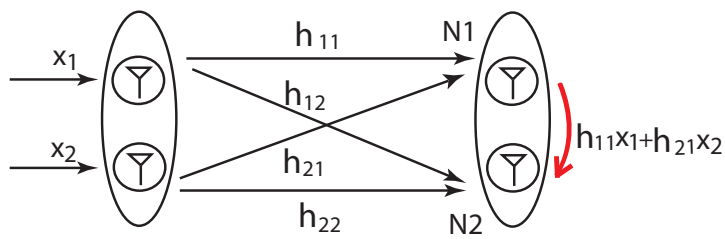

Figure 2.9: Pair-to-pair based buddy forwarding enabled by A&F.

Fig. 2.9 illustrates how A&F can enable pair-to-pair data forwarding that underlies our proposal of Buddy Routing (BR). Assume the source packet x for transmission is divided into two equal-length sub-packets x_1 and x_2. We pair up each of the Tx node and Rx node with a nearby 'buddy' node. The Tx node shares x_2 with its buddy, through a short intra-pair transmission. Next, the two nodes at the Tx side send x_1 and x_2 at the same time. Each node at the Rx side receives one combined analog signal respectively. Then the upper node (N1) forwards an amplified version of its received signal $h_{11}a_1x_1 + h_{21}a_2x_2$ to the lower node (N2). As a result, Node N2 has the two equations with two unknowns, allowing it to decode x_1 and x_2. Here h_{ij} is a complex number characterizing channel

fading from a node in the Tx pair to a node in the Rx pair, which includes amplitude attenuation and phase shift.

2.4.2 Pair-to-Pair Forwarding: PNC

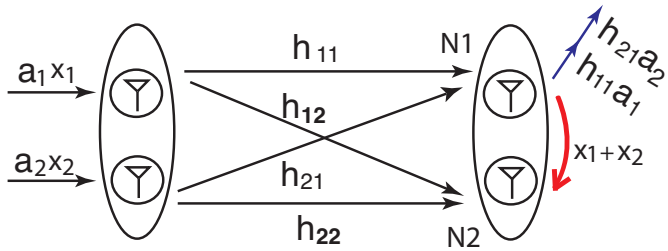

Figure 2.10: Pair-to-pair based buddy forwarding enabled by PNC. Precoding is performed at the Tx pair, for signal alignment at N1: $h_{11}a_1 = h_{21}a_2$.

The idea of Buddy Routing can alternatively be realized through PNC. The pair-to-pair forwarding gadget depicted in Fig. 2.10 illustrates the operation process. First, the two Tx nodes simultaneously transmit x_1 and x_2 respectively to the two Rx nodes. Precoding is performed such that their signals are aligned at the buddy node ($N1$) in the Rx pair. Node N1 then applies PNC to demodulate $x_1 + x_2$, and forwards it to the Rx node ($N2$). The Rx node can recover the original packet x from the analog signal it receives, $h_{12}a_1x_1 + h_{22}a_2x_2$, and the encoded packet from its buddy, $x_1 + x_2$, *e.g.*, through an adapted version of Maximum-Likelihood (ML) decoding [42]. Higher communication rate is targeted for data sharing within each pair, with a higher modulation rate. For example, BPSK modulation can be applied for the inter-pair transmission, and 16QAM for intra-pair.

The differences between PNC and A&F include two aspects. First, there is no alignment in A&F. The two nanonodes can transmit the original packets x_1 and x_2 without the precoding vectors. Second, in A&F, the upper node doesn't need to recover the digital version of the received packet, it only needs to transmit an amplified version of the

received analog signal $h_{11}a_1x_1 + h_{21}a_2x_2$ to the lower node; in PNC, the upper Rx node needs to decode the combined packet, then it transmits a digital version of $x_1 + x_2$.

2.5 Multi-hop Routing Algorithms

We propose Buddy Routing, a PNC-enabled pair-to-pair routing solution, for nanonetworks in Chapter 4. The routing algorithm we designed is based on the Greedy Perimeter Stateless Routing (GPSR) protocol for the wireless network [18]. We also compare the throughput of BR with that of traditional point-to-point routing protocols through analysis and simulation studies. This section provides two representative point-to-point routing algorithms: Ad hoc On-demand Distance Vector (AODV) routing and Dynamic Source Routing (DSR) and some background information about GPSR.

2.5.1 DSR and AODV

Dynamic Source Routing (DSR) [16] is a routing protocol for multi-hop wireless networks. It utilizes source routing to discover and maintain the routes in an ad hoc network. It consists of two main phases that work together. Route discovery is the mechanism by which a node S wishing to send a packet to a destination node D obtains a source route to D. Route Discovery is used only when S wants to send a packet to node D, but does not know a route to D. Route maintenance is the mechanism by which node S is able to discover, while using a source route to D, if the network topology has changed such that it can no longer forward the packet along the route because some hops along the route are broken.

Ad hoc On-demand Distance Vector (AODV) routing is another routing protocol for wireless ad-hoc networks [26]. It is similar to DSR in that it creates a route on-demand when a transmitting node requests one. DSR includes source routes in packet headers, resulting in large headers that can sometimes degrade performance. AODV attempts to

improve DSR by maintaining routing tables at wireless nodes, so that data packets do not have to contain routes. In AODV, the network is active only when a connection is needed. At that point, the network node that needs a connection broadcasts a request. Other nodes make a record for the node that they heard from, then create a temporary route back to the node in need. It will also forward this message to all the neighbors. If a node finds that it already has a route to the desired destination when receiving such a message, it sends a message backwards through a temporary route to the requesting node. The source node then begins using the path with the fewest hops. Unused entries in the routing tables are recycled after a period.

2.5.2 Greedy Perimeter Stateless Routing (GPSR)

The main idea behind GPSR [18] is to exploit location information of wireless nodes for making packet forwarding decisions during the routing process. Different from the traditional shortest-path algorithm, it only requires the propagation of network topology information within a single hop: each node only needs to know its neighbors' locations. There are two main components in GPSR: one is *greedy forwarding*, which is used wherever possible, *i.e.*, whenever a node can forward the packet to one of its neighbors who is closer to the final destination of that packet; another is *perimeter forwarding*, which is used when a packet reaches an area where greedy forwarding is impossible.

Greedy Forwarding:

In GPSR, the source marks all the packets with their destination's location. As a result, a forwarding node can choose a packet's next hop based on the distance to the destination. Specifically, if a node can detect its neighbors' positions, the locally optimal choice of next hop is the neighbor geographically closest to the packet's destination. The packet is forwarded according to this method successively, getting geographically closer to its destination, until the destination is reached. Fig. 2.11 shows an example of a greedy

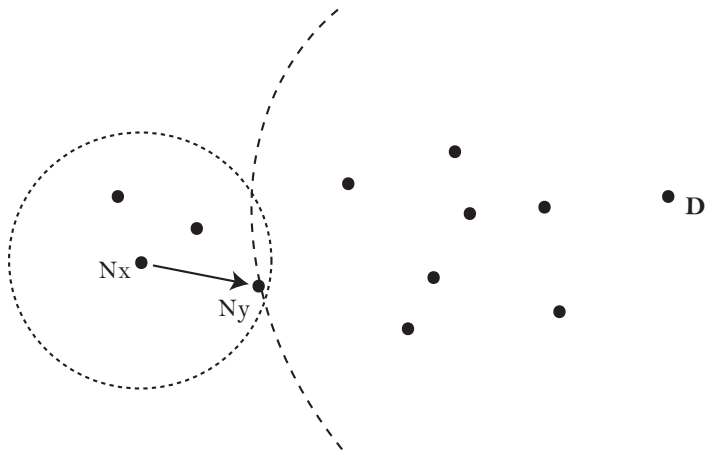

Figure 2.11: Greedy forwarding example. Ny is Nx's neighbor closest to D.

next hop choice. The destination of the packet is node D. After Nx receives the packet, it makes a greedy choice to forward the packet to its neighbor Ny. As shown in the figure, Nx's radio range is denoted by a dotted circle around Nx, and the arc with radius equal to the distance between Ny and D is shown as the dashed arc around D. Because the distance between Ny and D is less than that between D and any of Nx's other neighbors, Nx forwards the packet to Ny. This greedy forwarding process repeats, until the packet reaches D.

Perimeter Forwarding:

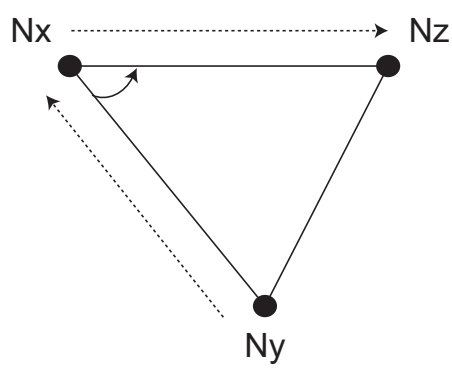

Figure 2.12: The right-hand rule.

A potential problem for greedy forwarding arises when the current node is closer to the destination than all its neighbors, and cannot reach the destination through a direct

transmission due to its limited transmission radius. In this case, it is impossible to apply greedy forwarding. A well-known solution here is the *right-hand rule* for traversing a planar graph. As shown in Fig. 2.12, this rule states that when a packet is arriving at a node Nx from node Ny, if the destination is located out of Nx's transmission radius, it forwards this packet to its first neighbor counterclockwise about itself, Nz.

2.6 Summary

In this chapter, we first described the main principles of physical layer network coding that are relevant to our work presented in this thesis. We have seen how PNC can be applied in the three-node linear network for throughput improvement. We also discussed the multi-input multi-output (MIMO) technology in a 2-client 2-AP system, and showed how interference alignment and cancelation (IAC) can improve the performance in this scenario. We then presented a brief overview of our PNC-SA scheme, and showed the transmission process in the same system. Finally, we also covered the background information on nanonetworks and three traditional routing algorithms, which are related to our routing algorithm design in Chapter 4.

Chapter 3

Physical Layer Network Coding with Signal Alignment for MIMO Wireless Networks

The main idea of PNC-SA has been shown in Fig. 2.7, in the MIMO uplink scenario. We can see that four packets can be obtained at the AP side in each time slot. In this chapter, we outline the system model and assumptions in Sec. 3.1. We present a detailed PNC-SA precoding and decoding solution in Sec. 3.2, and analyze its BER performance in Sec. 3.3. Sec. 3.5 analyzes the degree of freedom of PNC-SA in a general N-user M-antenna uplink scenario. Sec. 3.6 presents applications of PNC-SA in multi-hop routing, and packet level simulations. Sec. 3.7 concludes the chapter.

3.1 Model and Notation

We consider a multi-hop wireless network where each node is equipped with one or more antennas. Flat Rayleigh channel fading [10, 21, 33] is assumed, in which a signal experiences amplitude attenuation and phase shift through a channel. In each one-hop transmission, the sender transmits an N_t-dimensional signal vector $\mathbf{x}$, using the same carrier frequency. The receiver records an N_r-dimensional complex signal vector $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$. Here $\mathbf{H}$ is the channel matrix of dimension $N_t \times N_r$, and each entry $h_{i,j}$ is the channel gain from Tx antenna i to Rx antenna j. All entries in $\mathbf{H}$, $\mathbf{x}$, and $\mathbf{y}$ are complex numbers. The length and direction of the vector representation of the complex number represent the amplitude (or amplitude attenuation) and phase (or phase shift) of the signal, respectively. An additive white Gaussian noise (AWGN) $\mathbf{n}$ with zero mean and variance σ_n^2 is assumed.

We assume that full channel state information (CSI) is available, *i.e.*, each node knows the channel matrices of all adjacent (MIMO) links. This information describes how a signal propagates from transmitters to receivers, and captures the effect of fading. A rich-scattering environment is assumed, such that channel matrices are of full rank. One of the popular approaches for MIMO channel estimation is the superimposed pilot sequence technique [13]. It transmits a known pilot signal (or training sequence), then estimates the channel information based on the received signal and knowledge of the training symbols. There exist a number of different approaches of channel estimation, including Least-Squares (LS) and Linear Minimum Mean Squared Error (LMMSE) methods [12, 30]. After successfully detecting the instantaneous CSI, a receiver can send it back to the transmitter for the adaptation of future transmissions. However, the feedback packets would consume channel bandwidth as well, which in turn may decrease the net throughput from the transmitter to the receiver.

On the other hand, Lu *et al.* [22] argue that PNC does not need perfect carrier-phase and symbol-level synchronization, and hence does not require accurate estimation of channel state information. They proposed a belief propagation (BP) algorithm for decoding at the receiver side. For unchannel-coded PNC, the BP method can significantly reduce the asynchrony penalties, while for channel-coded PNC, incomplete channel state information improves the system performance, compared with the perfectly synchronous case. It is an interesting future research direction to study the performance of PNC-SA with relaxation of perfect CSI.

The *trace* of a matrix $\mathbf{A}$ is $Tr(\mathbf{A}) = \sum_i A_{ii}$. $\mathbf{A}^*$ denotes the *conjugate transpose* of a matrix $\mathbf{A}$, obtained by transposing $\mathbf{A}$ first, and then negating the imaginary component of each entry. The *Frobenius norm* of a matrix $\mathbf{A}$ is $\|\mathbf{A}\|_F = (\sum_i \sum_j |A_{ij}|^2)^{\frac{1}{2}} = (Tr(\mathbf{A}^*\mathbf{A}))^{\frac{1}{2}}$. The *Euclidean norm* of a vector $\mathbf{v}$ is $\|\mathbf{v}\| = (\sum_i |v_i|^2)^{\frac{1}{2}}$. A matrix $\mathbf{A}$ is a *unitary matrix* if it satisfies $\mathbf{A}^* = \mathbf{A}^{-1}$. A unitary matrix $\mathbf{A}$ preserves the Frobenius

norm, *i.e.*, $\|\mathbf{AB}\|_F = \|\mathbf{B}\|_F$.

Through this chapter, matrices are denoted with boldface capital letters, vectors with boldface lowercase letters, and scalars with non-boldface letters.

3.2 A Detailed PNC-SA Scheme Design

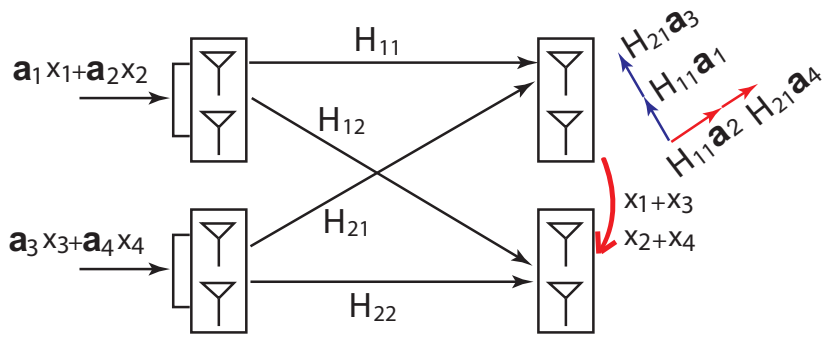

Figure 3.1: PNC-SA can achieve a throughput of 4 packets per time unit.

We now present a detailed PNC-SA solution design, with reference to the uplink in Fig. 3.1 for ease of exposition. Applications of PNC-SA elsewhere share a similar workflow. In particular, we study SNR-maximizing precoding at the sender side, and tailored detection and decoding algorithms at the receiver side. The BER performance will be analyzed in Sec. 3.3.

3.2.1 PNC-SA Precoding at Clients

Recall the PNC-SA scheme in Fig. 3.1. Let $\mathbf{v_1}$ and $\mathbf{v_2}$ be two 2×1 vectors that denote the target directions at AP1 for signal alignment and $\mathbf{v_1} \neq \mathbf{v_2}$. We have the following *alignment constraints*:

$$\mathbf{H_{11}a_1} = \mathbf{H_{21}a_3} = \mathbf{v_1}, \quad \mathbf{H_{11}a_2} = \mathbf{H_{21}a_4} = \mathbf{v_2}$$

The degree-of-freedom of our PNC-SA scheme, *i.e.*, the number of packets that can

be successfully transmitted in high SNR with low BER, is limited by the inequality:

$$\text{Number of (precoding) variables} \geq \text{Number of (alignment) constraints.}$$

The set of equations have at least one solution when the number of variables is at least as large as the number of equations. The same condition applies in the general N-user M-antenna MIMO uplink scenario, where APs are interconnected through Ethernet links. A detailed discussion is presented in Sec. 3.4. For the 2×2 system in Fig. 3.1, four packets are simultaneously transmitted at the client side. We have four 2×1 precoding vectors, $\mathbf{a_1}, \ldots, \mathbf{a_4}$. So

$$\text{Number of variables for precoding} = 4 \times 2.$$

Next, we have two equations in the alignment constraints, and each equation includes 2 subequations. Therefore,

$$\text{Number of constraints for alignment} = 2 \times 2.$$

Apparently, $4 \times 2 > 2 \times 2$. Therefore, transmitting 4 packets in the 2-client 2-AP MIMO system with PNC-SA is feasible.

Another type of constraint in PNC-SA comes from the power budget available at each client, E_T. Let $\mathbf{A_1} = (\mathbf{a_1}, \mathbf{a_2})$ and $\mathbf{A_2} = (\mathbf{a_3}, \mathbf{a_4})$ be the 2×2 precoding matrices at clients 1 and 2, respectively. The *nodal power constraint* requires:

$$\|\mathbf{A_1}\|_F^2 = Tr(\mathbf{A_1^*}\mathbf{A_1}) \leq E_T,$$

$$\|\mathbf{A_2}\|_F^2 = Tr(\mathbf{A_2^*}\mathbf{A_2}) \leq E_T.$$

Optimal PNC-SA Precoding: Formulation

Given the two types of constraints, the client-side precoding aims to maximize the SNR of x_1+x_3 and x_2+x_4, for demodulation at AP1, leading to the following optimal PNC-SA precoding problem:

Maximize $\quad f(\mathbf{V}) = |\mathbf{v_1^\dagger} \cdot \mathbf{v_2}| \qquad (1)$

Subject to:

$$\begin{cases} \mathbf{H_{11}A_1} = \mathbf{V} = \mathbf{H_{21}A_2} & (2) \\ \|\mathbf{A_1}\|_F^2 \leq E_T & (3) \\ \|\mathbf{A_2}\|_F^2 \leq E_T & (4) \end{cases}$$

Here $\mathbf{v_1}^\dagger$ is an orthogonal vector to $\mathbf{v_1}$ with equal length: if $\mathbf{v_1} = (c_1, c_2)^T$, then $\mathbf{v_1^\dagger} = (c_2^*, -c_1^*)^T$, and $\mathbf{v_1} \cdot \mathbf{v_1^\dagger} = 0$. The inner product $f(\mathbf{V}) = |\mathbf{v_1^\dagger} \cdot \mathbf{v_2}|$ targets two goals. The first is maximizing $|\mathbf{v_1}|$ and $|\mathbf{v_2}|$, for large received signal strength at AP1. The second is to make $\mathbf{v_1}$ and $\mathbf{v_2}$ as orthogonal as possible. The two goals together help maximize the SNR of detecting x_1+x_3 and x_2+x_4.

PNC-SA Precoding: Solution

Solving the vector programming problem in (1) is in general computationally expensive [37], especially when the number of antennas is large. The classic water filling approach [33] allocates more power to "better" subchannels with higher signal-to-noise ratio if the channel can be partitioned into parallel independent subchannels. Nevertheless, it cannot be directly applied to our PNC-SA scheme, due to the extra alignment constraints in (2). We design an efficient approximate solution instead, which leads to a closed-form representation of the precoding scheme, and becomes optimal with two reasonable restrictions on the precoding space.

Consider the following refinements on the precoding space: (a) $\mathbf{v_1}$ and $\mathbf{v_2}$ are orthogonal. Having orthogonal signals for x_1+x_3 and x_2+x_4 is in general beneficial to their detection; (b) $\|\mathbf{v_1}\| = \|\mathbf{v_2}\|$. This is also reasonable, assuming information contained in x_1+x_3 and in x_2+x_4 are equally important.

Given (a) and (b) above, $\mathbf{V}$ can be scaled to a unitary matrix $\mathbf{V_0}$ with total power of 2. We compute how much power is required at each client, for its transmitted signals

to fade into a unitary $\mathbf{V_0}$ at AP1. The power required at client 1 is:

$$\|\mathbf{A_1}\|_F^2 = \|\mathbf{H_{11}^{-1}V_0}\|_F^2$$

Since $\mathbf{V_0}$ is unitary, it preserves the Frobenius norm of $\mathbf{H_{11}^{-1}}$, hence $\|\mathbf{A_1}\|_F^2 = \|\mathbf{H_{11}^{-1}}\|_F^2$. This significantly simplifies the precoding design, by decoupling joint precoding at both clients to independent precoding at each of them. Similarly, the power required at AP2 is $\|\mathbf{A_2}\|_F^2 = \|\mathbf{H_{21}^{-1}}\|_F^2$. Let

$$\xi = \max(\|\mathbf{H_{11}^{-1}}\|_F^2), \|\mathbf{H_{21}^{-1}}\|_F^2),$$

we can set the precoding matrices by first picking an arbitrary unitary matrix $\mathbf{V_0}$, and then set:

$$\mathbf{A_1} = \sqrt{\frac{E_T}{\xi}}\mathbf{H_{11}^{-1}V_0}, \mathbf{A_2} = \sqrt{\frac{E_T}{\xi}}\mathbf{H_{21}^{-1}V_0}.$$

The solution above satisfies both the alignment constraint in (2), and the power constraints in (3)-(4) (at least one of them is tight), and maximizes the objective function in (1) under the two simplifying assumptions in (a) and (b).

3.2.2 PNC-SA Demodulation at AP1

The digital packets x_1+x_3 and x_2+x_4 are demodulated at AP1 in two steps. Assuming BPSK modulation (+1 for 1, −1 for 0) at the clients, AP1 first detects ternary values in $\{-2, 0, +2\}$, then maps them to binary values in $\{0, 1\}$ through PNC mapping. We next discuss two detection schemes, ZF and ML, followed by PNC mapping.

ZF Detection. Conceptually, AP1 can view x_1+x_3 and x_2+x_4 as two variables, and solve them through the two received signals at its antennas. ZF detection does so by projecting the combined signals to a direction orthogonal to x_2 (x_1), for detecting x_1 (x_2). ZF is particularly well-suited for PNC-SA, if we have restricted $\mathbf{v_1}$ and $\mathbf{v_2}$ to be orthogonal, as described in Sec. 3.2.1. The ZF projection matrix is a scaled conjugate transpose of $\mathbf{V_0}$ selected in Sec. 3.2.1, $\sqrt{\frac{\xi}{E_T}}\mathbf{V_0^*}$:

$$
\begin{aligned}
\tilde{\mathbf{y}} &= \sqrt{\frac{\xi}{E_T}}\mathbf{V_0^*}\mathbf{y} \\
&= \sqrt{\frac{\xi}{E_T}}\mathbf{V_0^*}(\mathbf{H_{11}A_1}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \mathbf{H_{21}A_2}\begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \mathbf{n}) \\
&= \sqrt{\frac{\xi}{E_T}}\mathbf{V_0^*}\left(\sqrt{\frac{E_T}{\xi}}\mathbf{V_0}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \sqrt{\frac{E_T}{\xi}}\mathbf{V_0}\begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \mathbf{n}\right) \\
&= \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \sqrt{\frac{\xi}{E_T}}\mathbf{V_0^*}\mathbf{n} \\
&= \begin{pmatrix} x_1 + x_3 \\ x_2 + x_4 \end{pmatrix} + \tilde{\mathbf{n}}
\end{aligned}
$$

Since the projection is linear, the projected noise $\tilde{\mathbf{n}} = \sqrt{\frac{\xi}{E_T}}\mathbf{V_0^*}\mathbf{n}$ is still AWGN.

ML Detection. Alternatively, we can apply the *a posteriori* method of ML detection. ML infers which source vector is most likely to have been transmitted, based on receiver side information. ML has a higher computational complexity than ZF, but provides optimal BER performance.

A salient difference between a standard ML scheme and ML for PNC-SA is that the former 'guesses' what's transmitted at each Tx antenna, while the latter 'guesses' the most probable linear combinations of the transmitted data. Equivalently, ML for PNC-SA views the multi-user MIMO channel from both clients to AP1 as a *virtual* 2×2 MIMO channel, with channel matrix $\sqrt{\frac{E_T}{\xi}}\mathbf{V_0}$ and ternary modulation, and detects the desired linear combination as:

$$
\begin{pmatrix} x_1 + x_3 \\ x_2 + x_4 \end{pmatrix} = \arg\min_{\mathbf{x}\in\{-2,0,2\}^2} \|\mathbf{y} - \sqrt{\frac{\mathbf{E_T}}{\xi}}\mathbf{V_0}\mathbf{x}\|
$$

PNC Mapping. While BPSK demodulation simply maps from $\{-1,\ 1\}$ to $\{0,\ 1\}$, PNC demodulation maps from $\{+2,\ 0,\ -2\}$ to $\{0,\ 1\}$ [41]. The basic rule is: $+2$ and

-2 map to 0, and 0 maps to 1. The intuition is that when $+2$ (-2) is seen, x_1 and x_3 (or x_2 and x_4) must have both been $+1$ (-1), and $x_1{+}x_3$ (or $x_2{+}x_4$) should be 0. Otherwise, $x_1{+}x_3$ (or $x_2{+}x_4$) should be 1. In the case of ZF detection, one may merge the ternary detection and ternary-to-binary mapping into a single step. Based on a maximum posterior probability criterion, Zhang and Liew [41] derived the following optimal decision rule for such direct mapping: map values between $-1-\alpha$ and $1+\alpha$ to 1, and other values to 0, for $\alpha = \frac{\sigma_n^2}{2}\ln(1+\sqrt{1-e^{-4/\sigma_n^2}})$.

3.2.3 PNC-SA Decoding at AP2

After receiving $x_1{+}x_3$ and $x_2{+}x_4$ from AP1, AP2 has accumulated four packets: two digital ones from AP1, and two analog ones from its own antennas:

$$\begin{cases} \begin{pmatrix} x_1 + x_3 \\ x_2 + x_4 \end{pmatrix} \\ \mathbf{y}' = \mathbf{H_{12}A_1}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \mathbf{H_{22}A_2}\begin{pmatrix} x_3 \\ x_4 \end{pmatrix} + \mathbf{n} \end{cases}$$

AP2 uses these four packets to decode x_1, x_2, x_3 and x_4. How does AP2 solve the four equations? We describe two approaches below: adapted ML decoding, and decoding via remodulation. The former provides better BER performance, while the latter scales better with the source symbol space and the number of antennas.

Adapted ML Decoding. The ML method can be adapted for decoding at AP2. AP2 traverses all possible combinations of (x_1, x_2, x_3, x_4). Before applying the normal min-distance criterion in ML, it first filters out the enumerated vectors that are not in agreement with the known values for $x_1{+}x_3$ and $x_2{+}x_4$. Consequently, adapted ML reduces the computational complexity of ML by a factor of 2^{N_r}, or a factor of 4 for the uplink in Fig. 3.1.

Decoding via Remodulation. Alternatively, AP2 may first re-construct the analog

version of x_1+x_3 and x_2+x_4 after modulation. Next, AP2 can apply low-complexity MIMO decoding methods (*e.g.*, ZF or MMSE-SIC [33]) to decode $x_1, \ldots, x_4$ as at a 4×4 MIMO receiver. The IC technique, as in IAC, is essentially decoding via remodulation in its simplest form, where only subtracting the remodulation of an uncoded packet is performed.

3.2.4 Discussions

PNC-SA provides full flexibility in precoding. Unlike IA or IAC, it places no restrictions on the precoding matrix, except that each sender can only encode locally available data. PNC-SA also opens new solution spaces for fully exploring the spatial diversity of a MIMO network, augmenting its achievable capacity region. This will be further demonstrated in Sections 3.6.1, 3.6.2, and 3.6.3. PNC alone can be viewed as a special case of PNC-SA, where each node has a receive diversity of 1, and SA degrades into signal synchronization. IAC can be viewed as a special case of PNC-SA, which further restricts the way SA is performed, precludes the application of PNC demodulation, and applies decoding via remodulation in its simplest form only.

The precoding optimization in Sec. 3.2.1 in general under-utilizes the available power at one of the clients, for exact signal matching between x_1 (x_2) and x_3 (x_4). It is possible to relax exact matching, and fully utilize all available power. An adapted PNC detection scheme will be required, with 4 instead of 3 possible values for combined signal strength. The current precoding optimization focuses on SNR at AP1 only. As a more comprehensive solution, one may formulate a global optimization that further considers the signal reception at AP2. We leave such a formulation and its solution as future work.

3.3 BER Analysis and Comparison

In this section, we analyze the BER performance of PNC-SA, and compare it with IAC. We first review the BER analysis of a general ML decoder, which will be helpful in analyzing the BER of PNC-SA and IAC.

3.3.1 BER of ML Detection

Consider a $N_t \times N_r$ MIMO channel. ML Detection searches for a source vector that was most likely to have been transmitted, based on information available at the receiver side:

$$\tilde{\mathbf{x}}_{\mathbf{ml}} = \arg\max_{\tilde{\mathbf{x}}_{\mathbf{i}}} p(\mathbf{y}|\mathbf{H}, \tilde{\mathbf{x}}_{\mathbf{i}}) = \arg\min_{\tilde{\mathbf{x}}_{\mathbf{i}}} \|\mathbf{y} - \mathbf{H}\tilde{\mathbf{x}}_{\mathbf{i}}\|^2$$

where the search space of the $N_t \times 1$ source vector $\tilde{\mathbf{x}}_{\mathbf{i}}$ has a size of M^{N_t}, M being the modulation alphabet cardinality. For flat Rayleigh fading with AWGN, the pairwise error probability (PEP), *i.e.*, the probability that MLD mistakenly outputs $\tilde{\mathbf{x}}_{\mathbf{k}}$ when a different source vector $\tilde{\mathbf{x}}_{\mathbf{i}}$ is transmitted, is

$$Pr(\tilde{\mathbf{x}}_{\mathbf{i}} \to \tilde{\mathbf{x}}_{\mathbf{k}}) = Q\left(\sqrt{\frac{d_{ik}^2}{2\sigma_{BPSK}^2}}\right) = Q\left(\sqrt{\frac{\|\mathbf{H}(\tilde{\mathbf{x}}_{\mathbf{i}} - \tilde{\mathbf{x}}_{\mathbf{k}})\|^2}{2\sigma_n^2}}\right) \qquad (5)$$

Here d_{ik} is the Euclidean distance between $\tilde{\mathbf{x}}_{\mathbf{i}}$ and $\tilde{\mathbf{x}}_{\mathbf{k}}$. Function Q computes the area under the tail of a Gaussian PDF. Using Boole's inequality, one can derive the average MIMO vector error probability:

$$Pr_s \leq \frac{1}{M^{N_t}} \sum_{\tilde{\mathbf{x}}_{\mathbf{i}}} \sum_{\tilde{\mathbf{x}}_{k \neq i}} Pr(\tilde{\mathbf{x}}_{\mathbf{i}} \to \tilde{\mathbf{x}}_{\mathbf{k}}), \qquad (6)$$

and, an approximation on BER can be found with

$$Pr_b \approx Pr_s/(N_t \log_2 M). \qquad (7)$$

3.3.2 BER Analysis of PNC-SA

The analysis of the BER performance of PNC-SA involves two phases. In phase one, we study the BER at AP1, for decoding x_1+x_3 and x_2+x_4. In phase two, we study the BER at AP2, using adapted ML for decoding $x_1, \ldots, x_4$.

BER at AP1. As discussed in Sec. 3.2.2, AP1 can demodulate x_1+x_3 and x_2+x_4 by applying ML detection over a virtual 2×2 MIMO channel. Let $\mathbf{c} = (c_t, c_b)^T$, where $c_t = x_1 + x_3$ and $c_b = x_2 + x_4$ are in the $\{-2, 0, 2\}$ domain, before PNC mapping. Let $\mathbf{c_i}$ and $\mathbf{c_k}$ be two possible 2×1 transmit vectors, with $i, k \in \{1, \ldots, 9\}$. Assume $\mathbf{c_i}$ is transmitted, from (5), the probability that AP1 incorrectly outputs $\mathbf{c_k}$ is:

$$\begin{aligned} Pr(\mathbf{c_i} \to \mathbf{c_k}) &= Q\left(\sqrt{\frac{d_{ik}^2}{2\sigma_{PNC-SA}^2}}\right) \\ &= Q\left(\sqrt{\frac{E_T/\varepsilon \|\mathbf{V}\|^2 \lambda_{ik}}{2\sigma_n^2}}\right) = Q\left(\sqrt{\frac{\lambda_{ik}\rho_{S1}}{2}}\right), \end{aligned}$$

where $\lambda_{ik} = (\mathbf{c_i} - \mathbf{c_k})^T(\mathbf{c_i} - \mathbf{c_k})$, and ρ_{S1} is SNR at AP1.

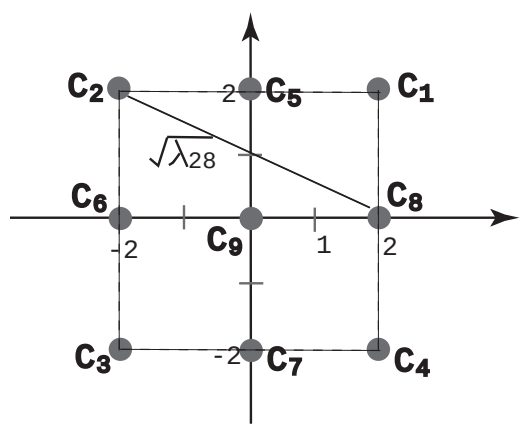

Figure 3.2: Constellation diagram for PNC-SA, at AP1.

Let's define constellation points $\mathbf{c_1}, \ldots, \mathbf{c_9}$ as shown in Fig. 3.2. Assuming 0 and 1 are equally likely to appear in the source packets, the ternary values in $\{-2, 0, 2\}$ appear in $\mathbf{c}$ with probabilities of 25%, 50%, and 25%, respectively. As a result, $P(\mathbf{c_1}) = P(\mathbf{c_2}) = P(\mathbf{c_3}) = P(\mathbf{c_4}) = 1/12$; $P(\mathbf{c_5}) = P(\mathbf{c_6}) = P(\mathbf{c_7}) = P(\mathbf{c_8}) = 1/8$; $P(\mathbf{c_9}) = 1/6$. AP1 wishes to demodulate the digital bits $\mathbf{d} = (d_t, d_b)^T$, where $d_t = x_1 + x_3$ and $d_b = x_2 + x_4$.

Thus, $Pr(\mathbf{c_i} \to \mathbf{c_k}) = 0$ when both $\mathbf{c_i}$ and $\mathbf{c_k}$ are in $(\pm 2, \pm 2)^T$. In other words, judging -2 to be $+2$ or vice versa does not lead to an error in $\mathbf{d}$. The average vector error probability for $\mathbf{d}$ is

$$Pr_s(\mathbf{d}) = 4P(\mathbf{c_1}) \sum_{i=5}^{9} Pr(\mathbf{c_1} \to \mathbf{c_i}) +$$

$$4P(\mathbf{c_5}) \sum_{i \neq 5} Pr(\mathbf{c_5} \to \mathbf{c_i}) + P(\mathbf{c_9}) \sum_{i=1}^{8} Pr(\mathbf{c_9} \to \mathbf{c_i})$$

BER at AP2. Consider applying adapted ML to decode $x_1, \ldots, x_4$ at AP2. We first study the case that $x_1 + x_3$ and $x_2 + x_4$ from AP1 are correct. We only need to search over source vectors that agree with the given $x_1 + x_3$ and $x_2 + x_4$ values. Under BPSK modulation, there are 4 such vectors, with dimension 4×1. Let $\tilde{\mathbf{x}}_\mathbf{i}$ and $\tilde{\mathbf{x}}_\mathbf{k}$ $(i, k \in \{1, \ldots, 4\})$ be two distinct vectors among the four. Assume $\tilde{\mathbf{x}}_\mathbf{i}$ is transmitted. By (5), the probability that AP2 outputs $\tilde{\mathbf{x}}_\mathbf{k}$ erroneously equals:

$$Pr(\tilde{\mathbf{x}}_\mathbf{i} \to \tilde{\mathbf{x}}_\mathbf{k} | \mathbf{d}_c) = Q\left(\sqrt{\frac{\lambda'_{ik} \rho_{S2}}{2}}\right).$$

Here $\lambda'_{ik} = (\tilde{\mathbf{x}}_\mathbf{i} - \tilde{\mathbf{x}}_\mathbf{k})^T (\tilde{\mathbf{x}}_\mathbf{i} - \tilde{\mathbf{x}}_\mathbf{k})$, and ρ_{S2} is the SNR at AP2. Let $\mathbf{d}_c$ and $\mathbf{d}_w$ denote the events that AP2 gets the correct and wrong data in $\mathbf{d}$ from AP1, respectively. The average vector error probability is:

$$Pr_s(\tilde{\mathbf{x}} | \mathbf{d}_c) = \frac{1}{4} \sum_{i=1}^{4} \sum_{k=1 k \neq i}^{4} Q\left(\sqrt{\frac{\lambda'_{ik} \rho_{S2}}{2}}\right).$$

Further including the case that $x_1 + x_3$ and $x_2 + x_4$ transmitted from AP1 contain errors, we have $Pr_s(\tilde{\mathbf{x}}) = Pr_s(\tilde{\mathbf{x}} | \mathbf{d}_c) Pr_s(\mathbf{d}_c) + Pr_s(\tilde{\mathbf{x}} | \mathbf{d}_w) Pr_s(\mathbf{d})$. When information from AP1 is wrong, AP2 outputs a wrong vector with probability 1, *i.e.*, $Pr_s(\tilde{\mathbf{x}} | \mathbf{d}_w) = 1$. Therefore, the vector error rate of the overall PNC-SA scheme is:

$$Pr_s(\tilde{\mathbf{x}}) = Pr_s(\tilde{\mathbf{x}} | \mathbf{d}_c)(1 - Pr_s(\mathbf{d})) + Pr_s(\mathbf{d}). \qquad (9)$$

Since the probability of more than two bit errors happening in the same vector is rather small, we ignore such probabilities. In adapted ML decoding, two of four bits

will be decided correctly in each vector, no matter whether AP2 has received the correct combination of $(x_1 + x_3, x_2 + x_4)$ or not. Thus, the average bit error probability is half the vector error probability. Now we can approximate the BER from the vector error rate:

$$Pr_b(\tilde{\mathbf{x}}) = Pr_s(\tilde{\mathbf{x}})/2. \qquad (10)$$

3.3.3 BER Analysis of IAC

The analysis of BER performance for IAC is also carried out in two steps (at AP1 and AP2), similar to the case of PNC-SA in Sec. 3.3.2.

BER at AP1 with ML detection.

With ML, AP1 can decode x_1 and x_2+x_3 simultaneously. Let $\mathbf{e}$ be the spatial source vector with $\mathbf{e} = (e_t, e_b)^T$. There are six possible vectors: $\mathbf{e_1} = (1,2)^T, \mathbf{e_2} = (1,0)^T, \mathbf{e_3} = (1,-2)^T, \mathbf{e_4} = (-1,2)^T, \mathbf{e_5} = (-1,0)^T, \mathbf{e_6} = (-1,-2)^T$. Assume $\mathbf{e_i}$ is transmitted, the probability that AP1 makes a wrong decision in favor of $\mathbf{e_k}$ $(k \neq i)$ equals

$$Pr(\mathbf{e_i} \to \mathbf{e_k}) = Q\left(\sqrt{\frac{d_{ik}^2}{2\sigma_{IAC}^2}}\right) = Q\left(\sqrt{\frac{\lambda_{ik}^{I1}\rho_{I1}}{2}}\right).$$

Here $\lambda_{ik}^{I1} = (\mathbf{e_i} - \mathbf{e_k})^T(\mathbf{e_i} - \mathbf{e_k})$, and ρ_{I1} is SNR at AP1. IAC only utilizes information in x_1. The BER of x_1 is:

$$\begin{aligned} Pr_b(x_1) &= Pr_s(x_1) = 4P(\mathbf{e_1})\sum_{i=4}^{6} Pr(\mathbf{e_1} \to \mathbf{e_i}) \\ &+ 2P(\mathbf{e_2})\sum_{i=4}^{6} Pr(\mathbf{e_2} \to \mathbf{e_i}). \end{aligned}$$

BER at AP2. After subtracting x_1 from its received signals, AP2 has two equations for x_2 and x_3. It can then decode x_2 and x_3 using ML detection. Let x_{1_c} and x_{1_w} denote the events that AP2 receives correct and wrong data in x_1 from AP1, respectively. If x_1

from AP1 is correct, the vector error rate at AP2 is:

$$Pr_s(AP2|x_{1\_c}) = \frac{1}{4}\sum_{i=1}^{4}\sum_{k=1k\neq i}^{4} Q\left(\sqrt{\frac{\lambda_{ik}^{I2}\rho_{I2}}{2}}\right),$$

where ρ_{I2} is SNR at AP2, $\lambda_{ik}^{I2} = (\tilde{\mathbf{z}}_\mathbf{i} - \tilde{\mathbf{z}}_\mathbf{k})^T(\tilde{\mathbf{z}}_\mathbf{i} - \tilde{\mathbf{z}}_\mathbf{k})$, $\tilde{\mathbf{z}}_\mathbf{i}$ and $\tilde{\mathbf{z}}_\mathbf{k}$ are two possible spatial source vectors and $i, k \in \{1, \ldots, 4\}$.

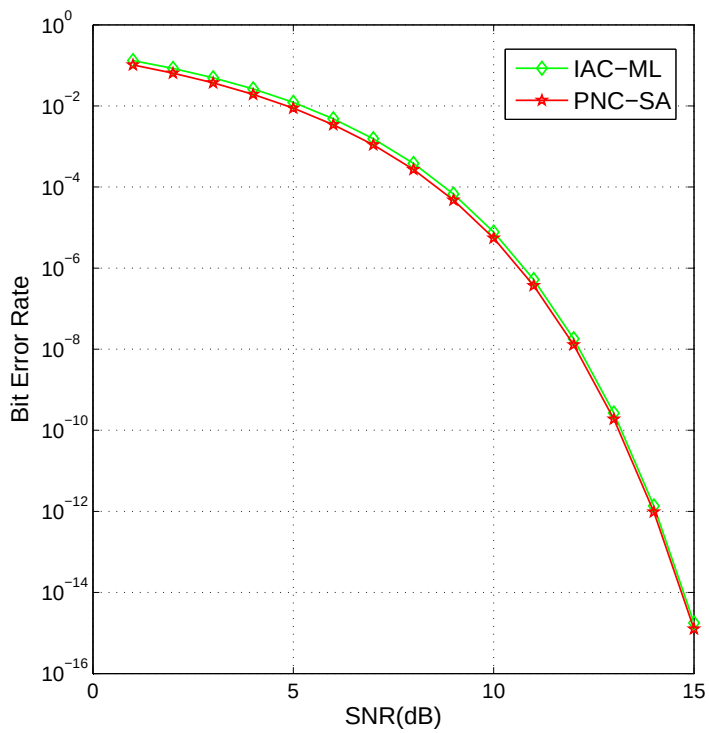

Figure 3.3: BER performance comparison: PNC-SA *vs* IAC.

There are two sources of BER in IAC. First, BER under event x_{1_c} can be calculated from the vector error rate directly.

$$\begin{aligned} Pr_b(x)_1 &= Pr_s(x)_1/N_t \log_2^M = Pr_s(AP2|x_{1\_c})Pr_s(x_{1\_c})/2 \\ &= Pr_s(AP2|x_{1\_c})(1 - Pr_s(x_1))/2. \end{aligned}$$

Second, when x_1 from AP1 is incorrect, the output of (x_2, x_3) at AP2 is almost surely wrong, due to error propagation. In other words,

$$Pr_b(x)_2 = Pr_s(x)_2 = Pr_s(AP2|x_{1\_w})Pr_s(x_{1\_w}) = Pr_s(x_1).$$

Now, the overall BER of the IAC scheme can then be approximated as: $Pr_b(x) = Pr_b(x)_1 + Pr_b(x)_2$.

3.3.4 Comparison of BER Performance

Fig. 3.3 shows the comparison of the BER performance of PNC-SA and IAC, as computed in Sec. 3.3.2 and Sec. 3.3.3, under varying SNR levels. We can observe that the BER of PNC-SA is rather close to, yet slightly better than, that of IAC, under the same SNR at the receiver's antennas.

3.4 PNC-SA with QPSK modulation

In the pervious sections, we introduced PNC-SA decoding and its BER preformation by assuming BPSK modulation. Note that the technique of PNC-SA is independent of the modulation scheme. We referred to BPSK simply for ease of exposition. Similar to PNC, PNC-SA can be applied with more sophisticated modulation schemes such as QPSK or 16QAM. In this section, we will discuss in detail how PNC-SA works with the QPSK modulation scheme.

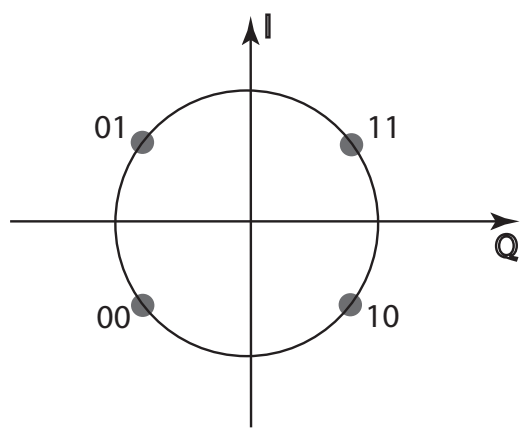

Figure 3.4: Constellation diagram for QPSK.

First, we would like to provide a simple introduction to the principle of QPSK mod-

ulation. Quadrature Phase-Shift Keying (QPSK) is a digital modulation scheme that conveys data by changing the phase of a reference carrier wave [5]. QPSK includes two components, it modulates by changing the phase of the in-phase (I) carrier from 0° to 180° and the quadrature-phase (Q) carrier between 90° and 270°. As shown in the constellation diagram in Fig. 3.4, QPSK uses four points around a circle to represent digital data. With four phases, QPSK can encode two bits per symbol.

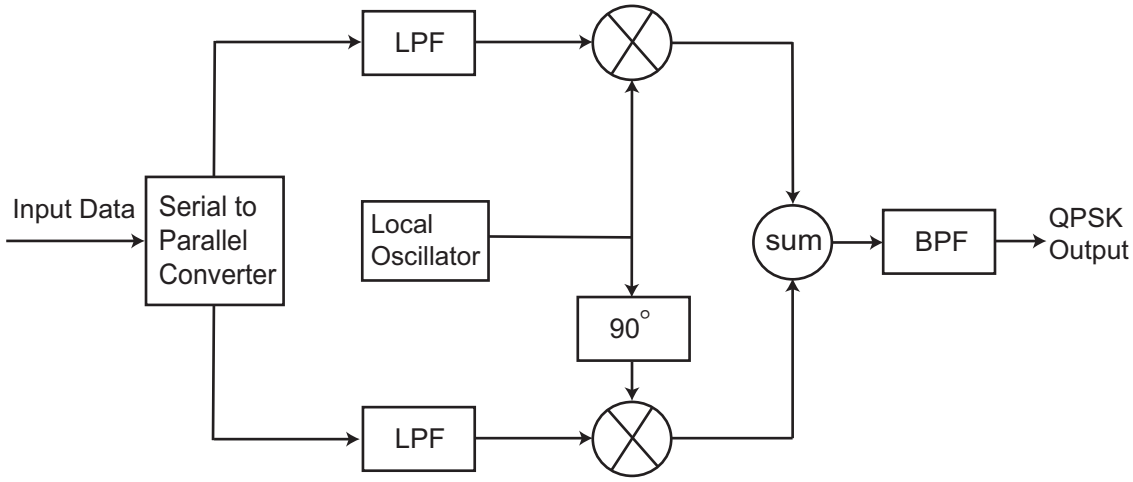

Figure 3.5: Block diagram of a QPSK transmitter

Fig. 3.5 shows a block diagram of a typical QPSK transmitter. The input binary data stream is split into the in-phase and quadrature-phase components by a serial to parallel converter. Then the two bit streams are fed to two orthogonal modulators after passing through the low pass filter (LPF). In the last step, the two modulated bit streams are summed and fed to the band pass filter (BPF) for producing the QPSK output.

When PNC-SA works with QPSK modulation applied at the client side in Fig. 3.1, each transmitted signal includes two substreams, in-phase and quadrature-phase stream. However, the client actually sends the sum of the in-phase and quadrature-phase waves, which is a composite wave with the same frequency. Furthermore, when we align x_1 and x_3 to the same direction at AP1, it is the composite signal, rather than the in-phase or quadrature-phase signal, that is being aligned. *Direction* here is an abstract

concept defined as a signal's encoding vector when received at AP1. When we restrict the alignment directions at AP1, $\mathbf{v_1}$ and $\mathbf{v_2}$, to be orthogonal, the directions of the composite signals become orthogonal.

Another issue is PNC-SA demodulation at AP1. Because the in-phase and quadrature-phase components of one combined QPSK signal propagate through the same fading channel, they would arrive with the same amplitude attenuation and phase shift, which means I and Q components are still orthogonal to each other. Therefore, if two composite signals are aligned to the same direction, their I and Q components should also have been aligned to the same direction. By ZF detection, we can first separate the two combined QPSK signals by projection, then apply QPSK demodulation and PNC-mapping to obtain the I and Q substreams that together form the digital version of $(x_1 + x_3)$ and $(x_2 + x_4)$. Alternatively, we can also apply ML detection to guess the most probable linear combinations of the transmitted data.

Similar to the case of QPSK, PNC-SA can be adapted to work with more complex schemes such as 16QAM. The higher the data rate that a modulation scheme can provide, the worse its BER performance is. There is always a tradeoff between the BER performance and the raw data rate.

3.5 General PNC-SA Throughput Analysis

The analysis in Sec. 3.2 focuses on a 2-client 2-AP scenario, where each node is equipped with two antennas. It illustrates that the *degree of freedom* of the PNC-SA scheme (the number of packets that can be transmitted in one slot, in high SNR) depends on both the number of variables in the precoding vectors at the Tx side and the number of constraints for signal alignment at the Rx side. In this section, we perform a close examination on such a dependence, as the system size grows, and summarize our findings in Theorem

3.1 and Theorem 3.2.

3.5.1 PNC-SA DoF: Direction and Amplitude Alignment

We first keep the assumption that in SA, both direction and amplitude should be equal for aligned signals. Consider a general $N \times N \times M$ uplink communication scenario with N clients on the Tx side and N APs on the Rx side, each equipped with M antennas. We make a practical assumption that each client is in possession of up to M packets for precoding and transmission.

Theorem 3.1: In the $N \times N \times M$ system, the DoF of PNC-SA, X, satisfies $(X - M)(\lceil \frac{X-2M}{M-1} \rceil + 1) \leq NM$ when $M > 1$.

Proof: When X packets are concurrently transmitted from the clients, successful signal alignment at the AP-side requires:

$$\#\text{ of precoding variables} \geq \#\text{ of alignment constraints}$$

Each node is equipped with M antennas, and the total number of precoding vectors is NM. For X concurrent packets, each AP has $X-M$ alignment constraints. For example, assume $N = 3$ and $M = 2$. The maximum X allowed by the theorem is $X = 5$. Five packets would be aligned into two directions: three packets along one direction and two along another. The number of alignment constraints for the two directions are $3-1$ and $2-1$, respectively. Therefore, the total number of alignment constraints is $2+1 = 5-2$.

Meanwhile, X independent equations need to be accumulated at the AP side, to recover all the X packets. Note that the two sets of equations from any pair of APs are dependent (the bit-wise xor of all these packets is always an all-zero packet). Therefore we can only obtain $M - 1$ useful equations from each AP, except the first, which can provide M useful equations. Now, there are $\lceil \frac{X-2M}{M-1} \rceil + 1$ APs where SA takes place. We also need to have different alignment patterns in each AP, to ensure linear independence

of the accumulated packets. This is possible because the number of APs that receive the alignment equations is less than X when M is greater than one, and there are already X combination patterns if we align a single packet in one direction and other packets in other directions.

Moreover, each constraint has M components. Hence,

$$\begin{cases} \#\text{ of alignment variables} = M \times NM \\ \#\text{ of alignment constraints} = M(X-M)(\lceil \frac{X-2M}{M-1} \rceil + 1) \end{cases}$$

Therefore, $(X-M)(\lceil \frac{X-2M}{M-1} \rceil + 1) \leq NM$ is satisfied when X packets are delivered.

On the other hand, we can perform precoding over X packets at the client side if X satisfies the above inequality. The last AP must be able to decode all the original packets by accumulating X independent equations at the AP side. □

We present the following Corollary as an illustrating example of Theorem 3.1.

Corollary 3.1. In a 3-client 3-AP system with M antennas per node, the DoF of PNC-SA $X = \lfloor \frac{5}{2}M \rfloor$.

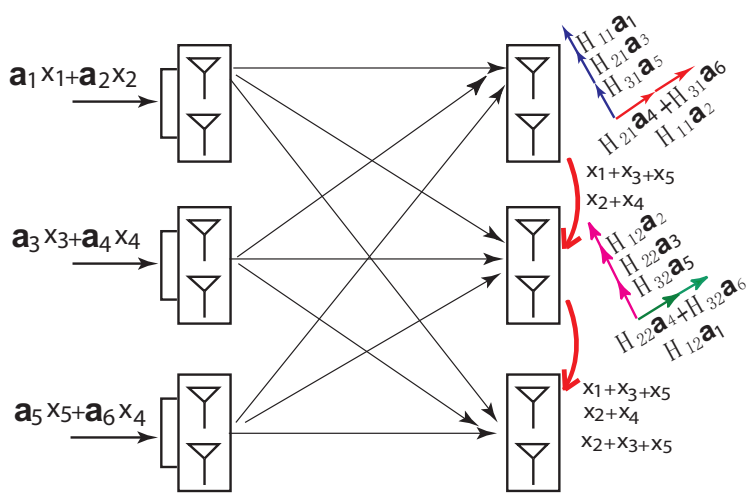

Figure 3.6: The DoF of PNC-SA is 5 in a $3 \times 3 \times 2$ system.

Fig. 3.6 shows the PNC-SA solution for transmitting 5 packets, in a $3 \times 3 \times 2$ system. Note that the third AP transmits two packets, x_4 and x_5 rather than x_5 and x_6.

At the first AP, we align x_1, x_3, and x_5 along one dimension, x_2 and x_4 along another dimension. AP1 decodes two unknowns $(x_1 + x_3 + x_5, x_2 + x_4)$ and transmits them to AP2. From Fig. 3.6, we can see the second step is performing another alignment in AP2, such that x_1 and x_4 arrive along the same direction, and same for x_2, x_3 and x_5. Because these four packets $(x_1 + x_3 + x_5, x_2 + x_4, x_2 + x_3 + x_5, x_1 + x_4)$ are linearly dependent, AP2 will only forward the first three packets to the last AP. With the five accumulated equations, AP3 then solves them to recover the five original packets, $x_1, \ldots, x_5$.

Now, let's apply this procedure in the M antennas case. Alignment is only performed in two APs.

$$\begin{cases} \#\text{ of variables} = M \times 3M \\ \#\text{ of constraints} = 2M(X - M) \end{cases}$$

Obviously, we can apply PNC-SA only under the condition that the number of constraints is less than or equal to the number of variables. Finally, we arrive at the conclusion that $X \leq \frac{5}{2}M$ by solving the inequality. Since X is an integer, Corollary 3.1 can be deduced easily.

3.5.2 PNC-SA DoF: Direction Alignment Only

We next consider the option of enforcing equal direction on the aligned signals only, while allowing them to arrive at a Rx antenna with unequal strengths. Note that new PNC mapping schemes different than the original proposal [41] are required, such as a lattice code based solution in Ref. [37].

Theorem 3.2. In a $N \times N \times M$ system, with direction alignment only, the DoF of PNC-SA, X, satisfies the inequality $(M-1)(X-M)(\lceil \frac{X}{M} \rceil - 1) \leq NM^2$ when $M > 1$.

Proof: The number of precoding vectors equals NM, and each vector has M dimensions. Furthermore, we have $X - M$ constraints at each AP. Meanwhile, we need to accumulate X equations to recover all the X packets. Since each AP can obtain M independent

equations, alignment constraints are only considered in $\lceil \frac{X}{M} \rceil - 1$ APs. The last AP, which collects M equations, doesn't need to perform alignment.

Under the direction only alignment scheme, each constraint has $(M-1)$ components. For example, $\mathbf{a} = (a_1, a_2, a_3), \mathbf{b} = (b_1, b_2, b_3)$, if we want $\mathbf{a}$ and $\mathbf{b}$ in the same direction, then we need $a_1/b_1 = a_2/b_2 = a_3/b_3$, so the total number of subequations equals $3-1 = 2$. Hence,

$$\begin{cases} \text{\# of alignment variables} = M \times NM \\ \text{\# of alignment constraints} = (M-1)(X-M)(\lceil \frac{X}{M} \rceil - 1) \end{cases}$$

On the other hand, if X satisfies the given inequality, it is possible to perform precoding over X packets at the client side, for desired SA at the AP side. Having accumulated X independent equations at the AP side, the last AP must be able to recover all the original packets. □

3.6 General Applications of PNC-SA and Packet-level Throughput

Applications of PNC-SA in wireless routing are diverse, and are not restricted to cases where receivers have limited collaboration (Fig. 3.1). In this section, we first present Matlab simulation results on packet level comparisons between PNC-SA and alternative solutions for the uplink scenario in Fig. 3.1. We then extend the discussions to more general applications of PNC-SA, for information exchange, unicast, and multicast/broadcast.

Fig. 3.7 shows the comparison of packet-level throughput achieved by PNC-SA, IAC, and MIMO, respectively. In the Matlab simulations, we assume a synchronized environment where the nodes transmit packets in rounds. For this first set of simulations, we let the system run for 100 rounds. During each round, each antenna transmits 1 packet of length 50 bits. Consequently, in each round, PNC-SA, IAC, and MIMO transmit 4, 3, and 2 raw packets, respectively. On the receiver side, we assume the existence of an

error detection scheme that can identify bit errors. A packet received with 1 or more bits in error is discarded and not counted towards total throughput. The bit errors are computed from SNR as discussed in Sec. 3.3. The SNR level is assumed to be equal at all nodes.

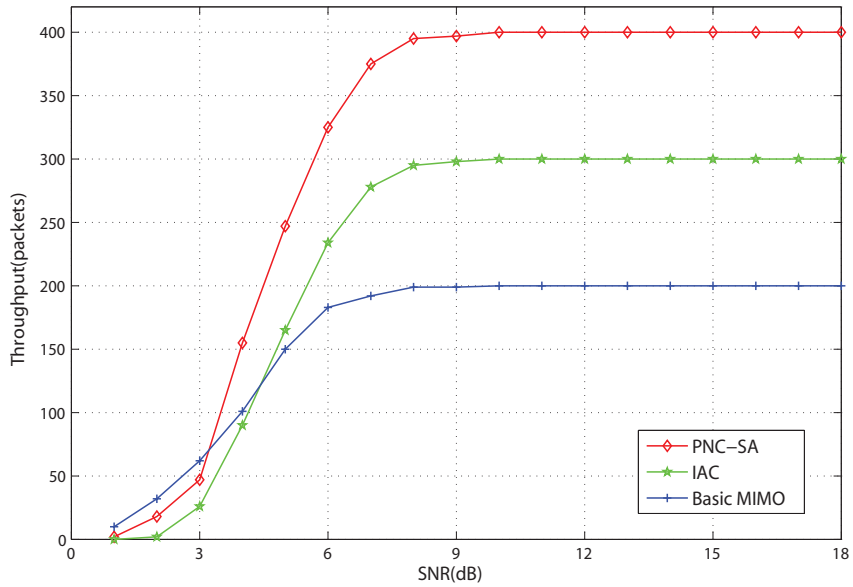

Figure 3.7: Packet-level throughput for multi-AP uplink communication, PNC-SA *vs.* IAC *vs.* MIMO alone.

From Fig. 3.7, we can see that at high SNR (> 9), the ratio of throughput achieved by the three schemes converges to $4:3:2$, with PNC-SA performing the best. As SNR decreases, the gap between PNC-SA and IAC slightly increases, due to the slightly better SNR-BER performance of PNC-SA, as shown in Fig. 3.3. It is interesting to note that basic MIMO actually performs the best at low SNR here, due to its better SNR-BER performance.

3.6.1 PNC-SA for Info Exchange

Fig. 3.8 shows the well-known Alice-and-Bob communication scenario in a wireless network, where Alice and Bob wish to exchange data packets with the help of a relay [19, 39].

Each node is equipped with 3 antennas. Transmitting simultaneously, Alice and Bob can align their six packets to three common directions at the relay. The relay then demodulates x_1+x_4, x_2+x_5, and x_3+x_6, and broadcasts them to both Alice and Bob. Alice and Bob each subtract their known packets from the three combined signals received, and apply normal demodulation to recover the other three packets.

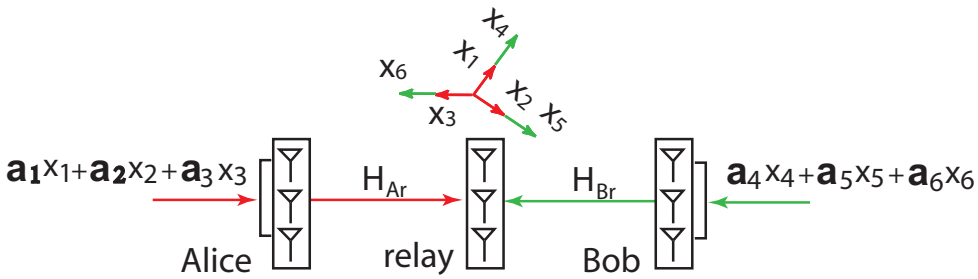

Figure 3.8: PNC-SA with three antennas per node. Here and in the rest of this chapter we label an aligned direction with the corresponding signal instead of its vector direction, for simplicity. For example, the direction of $\mathbf{H_{11}a_1}$ is simply labelled as x_1.

With PNC-SA, 6 packets can be exchanged in 2 time slots. Without PNC-SA, it takes 3 time slots with digital network coding, and 4 time slots with no coding at all [19]. Without SA, PNC alone does not fully exploit the full degree of freedom of such a MIMO network. For example, Zhang and Liew [39] studied the utilization of multiple antennas at the relay, by combining its received signals for generating a single encoded packet, for better BER.

We can see that the application of PNC-SA is not limited to scenarios with limited receiver collaboration; nor is it limited to 2 antennas per node. Examples shown in this chapter can all be generalized to work with 3 or more antennas per node.

Fig. 3.9 shows the packet-level throughput comparison between PNC-SA, DNC, and basic MIMO. Here the system is run for 200 time slots, during each of which an antenna can transmit 1 packet of 50 bits. We can observe that at high SNR, the throughput ratio converges to 6 : 4 : 3, with PNC-SA leading the alternatives. At low SNR, DNC performs

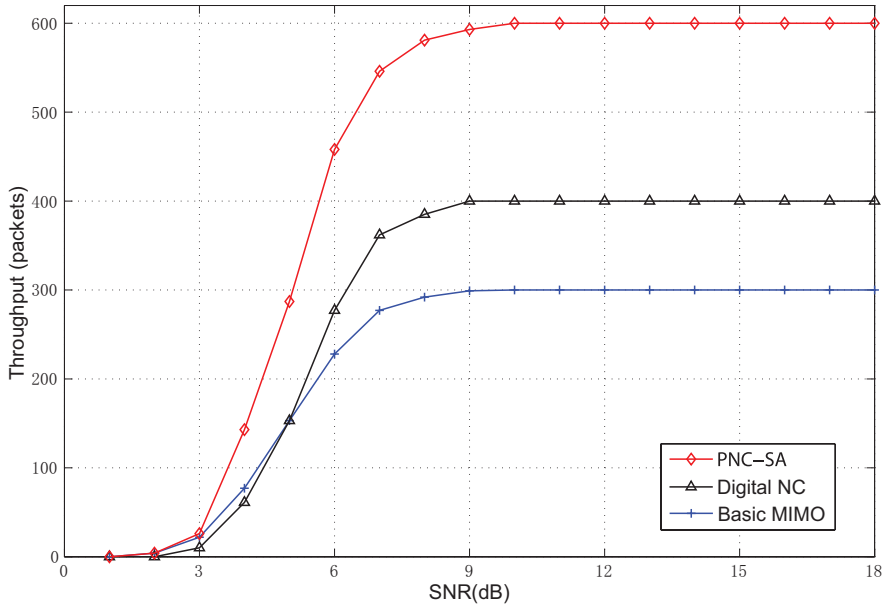

Figure 3.9: Packet-level throughput for information exchange, PNC-SA *vs.* DNC *vs.* MIMO alone.

the worst. The main reason is that DNC needs to succeed in all transmissions in 3 time slots for successful packet reception and decoding, while PNC-SA and MIMO only need 2 time slots each.

3.6.2 PNC-SA for Unicast Routing

PNC-SA for Cross Unicasts

Fig. 3.10 depicts two unicast sessions, from S_1 to T_1 and from S_2 to T_2, whose routes intersect at a relay. Each sender cannot directly reach its intended receiver, and needs to resort to the help of the relay node in the middle.

With PNC-SA, the two senders can transmit simultaneously. They align the signals for reception at the relay, such that x_1 is aligned with x_3, and x_2 with x_4. The relay decodes and broadcasts x_1+x_3 and x_2+x_4. Only 3 transmissions in 2 time slots are required. T_1 can first decode x_3 and x_4 overhead from S_2, and then combine them with

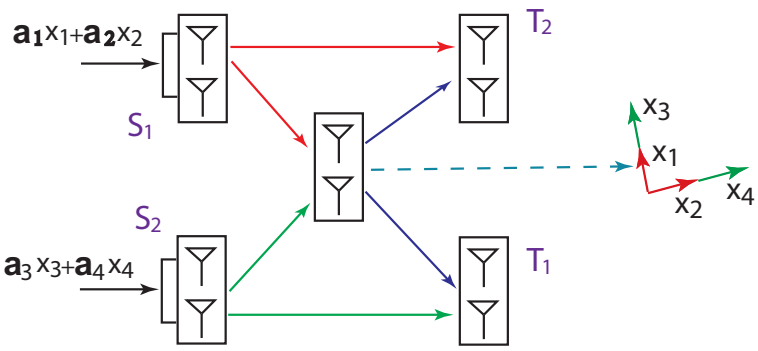

Figure 3.10: PNC-SA with PNC performed at the relay node in the middle.

x_1+x_3 and x_2+x_4 to recover x_1 and x_2. T_2 recovers x_3 and x_4 similarly.

Without any coding, it takes 4 transmissions in 4 time slots to send 2 packets in each session: each sender transmits once (using both antennas), and the relay transmits twice. With DNC, it takes 3 transmissions in 3 time slots — the relay can transmit just once, broadcasting two encoded packets.

The PNC-SA precoding optimization discussed in Sec. 3.2.1 still applies here. SA enables PNC in this MIMO network, and PNC further enables demodulate-and-forward at the relay, which provides an alternative to amplify-and-forward for cooperative communication [29]. In general multi-session unicast routing, such a cross-unicast topology can be applied as a gadget, embedded into larger unicast sessions [19].

Fig. 3.11 shows packet-level throughput comparison between PNC-SA and a basic MIMO solution. Again the network is run for 200 time slots, with the same node transmission capacity and packet lengths as previously assumed. At high SNR, the throughput gap between PNC-SA and MIMO is a factor of 2, confirming the analysis above. As SNR decreases, however, MIMO catches up with PNC-SA and eventually outperforms, due to its better SNR-BER performance. This suggests that a good design of error-correction code in combination with PNC-SA is important at the low SNR regime.

The Zig-Zag Unicast Flow: PNC Meets DNC

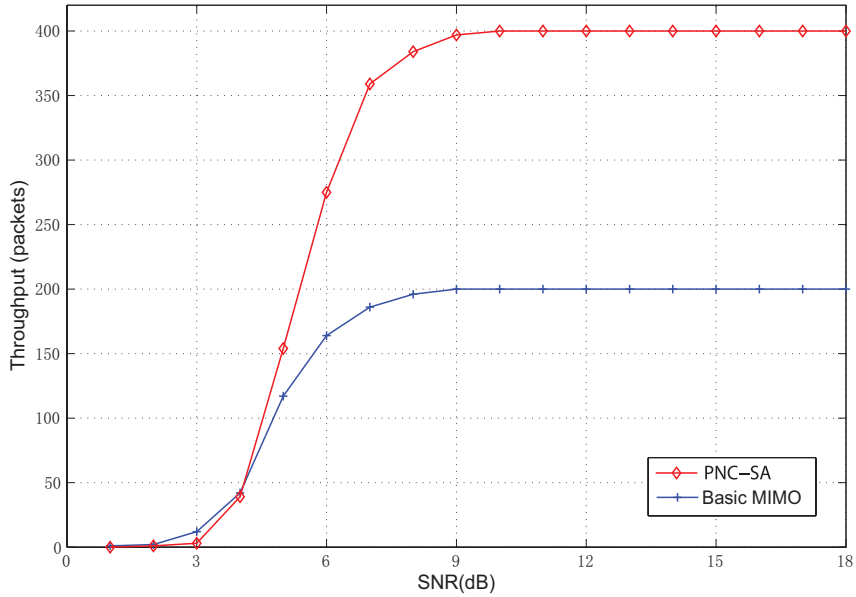

Figure 3.11: Packet-level throughput for cross unicasts, PNC-SA *vs.* MIMO alone.

Existing literature on the application of network coding in wireless routing often focuses on identifying local gadgets, such as the Alice-and-Bob topology and the cross-unicast topology [19, 41]. These gadgets usually involve multiple unicast sessions with reverse or crossing routes. It is often believed that network coding provides little benefit to a single unicast session with lossless links [11, 20]. We present an application of PNC-SA, where PNC and DNC work in concert to enable a new, efficient wireless unicast routing algorithm.

Consider a large wireless mesh network, with two antennas per sensor. We wish to transfer information from the top of the network to the bottom [9]. What multi-hop unicast routing scheme can we use, to achieve a high throughput? Fig. 3.12 illustrates a PNC-SA based solution: a *zig-zag unicast flow*.

The zigzag solution routes k parallel data streams side by side, employing k nodes for transmission at each row (k=3 in Fig. 3.12). The resulting unicast flow exhibits a zigzag topology. The following theorem shows that the packets at each row can be used

to recover the $2k$ original packets.

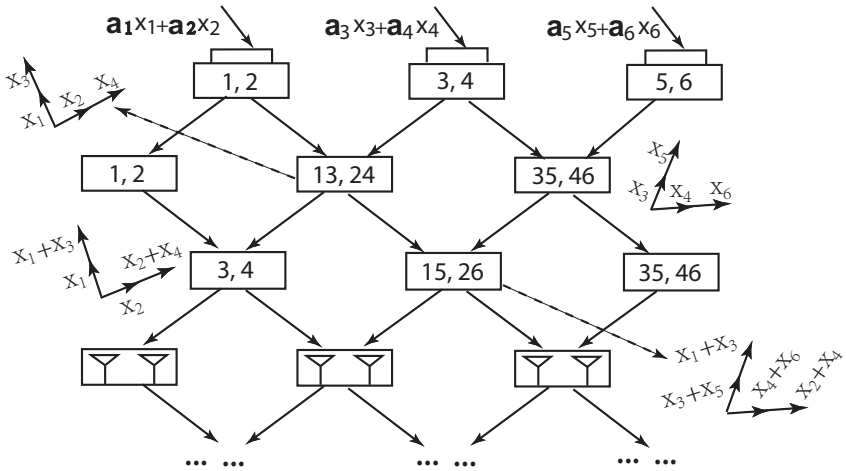

Figure 3.12: The zig-zag unicast flow using PNC-SA. Here 35, 46 in a node represents x_3+x_5 and x_4+x_6. The first row transmits 6 packets simultaneously. The signals are aligned at the second row for demodulating (x_1, x_2), (x_1+x_3, x_2+x_4) and (x_3+x_5, x_4+x_6). In the odd (even) rows, the left-most (right-most) node receives from one sender in the previous row only, without PNC.

Theorem 3.3. At each row in the zigzag unicast flow, the $2k$ *data packets are linearly independent, and can be used to recover the original packets* $x_1, \ldots, x_{2k}$.

Proof: We prove the theorem using a row-by-row induction. As the basis, the $2k$ packets at the first row are the original ones, and are independent. Assume the packets at row i, $y_1, \ldots, y_{2k}$, are independent. Number the nodes in each row from left to right. Without loss of generality, assume the left-most node (node 1) in row $i+1$ receives packets without PNC coding. Packets at node 1 in row $i+1$ are y_1 and y_2. Packets at node 2 in row $i+1$ are y_1+y_3 and y_2+y_4 and can be used to further recover y_3 and y_4. Similarly, each node $j \in [2 \ldots k]$ in row $i+1$ possesses packets that can be used to further recover y_{2j-1} and y_{2j}. In conclusion, packets at row $i+1$ can be used to recover all packets in row i. Since the latter are linearly independent, so are the former. □

The table below lists the packets received by nodes at each row, for $k = 3$. The intra-row linear independence can be verified. It is also interesting to observe that after

every 7 rows, the 6 data packets in routing return to uncoded form.

row	node 1	node 2	node 3
0	x_1, x_2	x_3, x_4	x_5, x_6
1	x_1, x_2	x_1+x_3, x_2+x_4	x_3+x_5, x_4+x_6
2	x_3, x_4	x_1+x_5, x_2+x_6	x_3+x_5, x_4+x_6
3	x_3, x_4	$x_1+x_3+x_5$, $x_2+x_4+x_6$	x_1+x_3, x_2+x_4
4	x_1+x_5, x_2+x_6	x_5, x_6	x_1+x_3, x_2+x_4
5	x_1+x_5, x_2+x_6	x_1, x_2	$x_1+x_3+x_5$, $x_2+x_4+x_6$
6	x_5, x_6	x_3+x_5, x_4+x_6	$x_1+x_3+x_5$, $x_2+x_4+x_6$
7	x_5, x_6	x_3, x_4	x_1, x_2

Compared to a basic single-chain unicast solution, the zigzag flow represents a $\times k$ throughput gain. Unlike traditional multi-path wireless routing, the k parallel data streams in the zigzag flow do not need to be spatially far apart to avoid interference, and is in that sense more practical to deploy. The rationale behind the zigzag structure guarantees that a node at the border obtains data without PNC, which can be used to bootstrap the decoding process along that row.

3.6.3 PNC-SA for Multicast/Broadcast Routing

Network coding is naturally well-suited for multicast and broadcast routing in wireless networks. The local broadcast nature of omnidirectional antennas is well suited for simultaneously transmitting an encoded packet to multiple receivers. PNC-SA extends such benefit of DNC to information dissemination in MIMO networks.

Multi-Sender Multicast

Fig. 3.13 depicts a multi-sender multicast in an 8-node MIMO network. The 3 top nodes are senders, and the 3 bottom nodes are receivers. Each sender wishes to multicast to all receivers. As another natural fusion of PNC and DNC, the application of PNC-SA

here doubles the achievable multicast throughput.

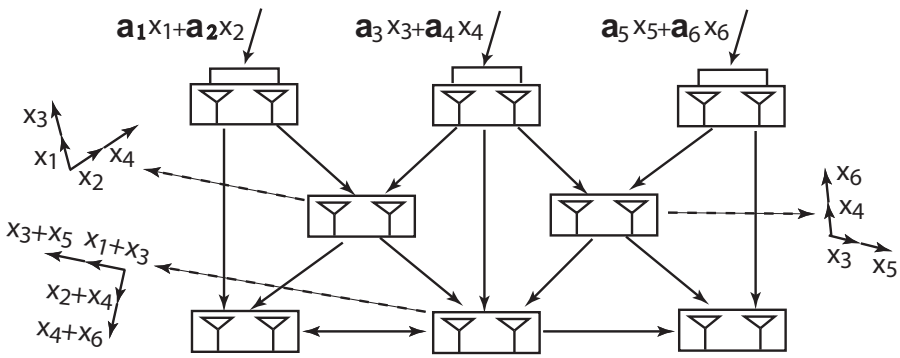

Figure 3.13: Multicast from top layer to bottom layer. PNC-SA doubles throughput.

With PNC-SA, 6 packets can be multicast to all receivers in 4 time slots. (i) The three senders align their six signals at the two relays in the middle, such that they can successfully demodulate $\{x_1{+}x_3,\ x_2{+}x_4\}$ and $\{x_3 + x_5,\ x_4 + x_6\}$, respectively. At the same time, the three receivers obtain $\{x_1,\ x_2\}$, $\{x_3,\ x_4\}$ and $\{x_5,\ x_6\}$, respectively. (ii) The two relays transmit $x_1{+}x_3$, $x_2{+}x_4$, respectively, simultaneously. Their signals are aligned so that the middle receiver can demodulate $x_1 + x_3 + x_3 + x_5 = x_1 + x_5$ and $x_2 + x_4 + x_4 + x_6 = x_2 + x_6$. From left to right, the three receivers accumulate $\{x_1,\ x_2,\ x_3,\ x_4\}$, $\{x_3,\ x_4,\ x_1 + x_5,\ x_2 + x_6\}$, and $\{x_3,\ x_4,\ x_5,\ x_6\}$, respectively. (iii) The middle receiver broadcasts $x_1{+}x_5$ and $x_2{+}x_6$, so that the other two receivers can now recover all 6 packets via DNC decoding. (iv) The left receiver transmits x_1 and x_2 to the middle receiver, who can now decode all 6 original packets too.

Using a straightforward multicast scheme without network coding, we need 7 time slots instead. x_1 and x_2 require 3 broadcasts to reach all receivers, the same for x_5 and x_6. x_3 and x_4 require two broadcasts. Among these 8 broadcast transmissions, only two can be scheduled concurrently, resulting in a total of 7 time slots. With DNC, the number of time slots required is between that of PNC-SA and a no coding solution, at 5.

Fig. 3.14 shows packet-level throughput achieved by PNC-SA, DNC, and MIMO. The network is simulated for 140 time slots, with identical node transmission capacity

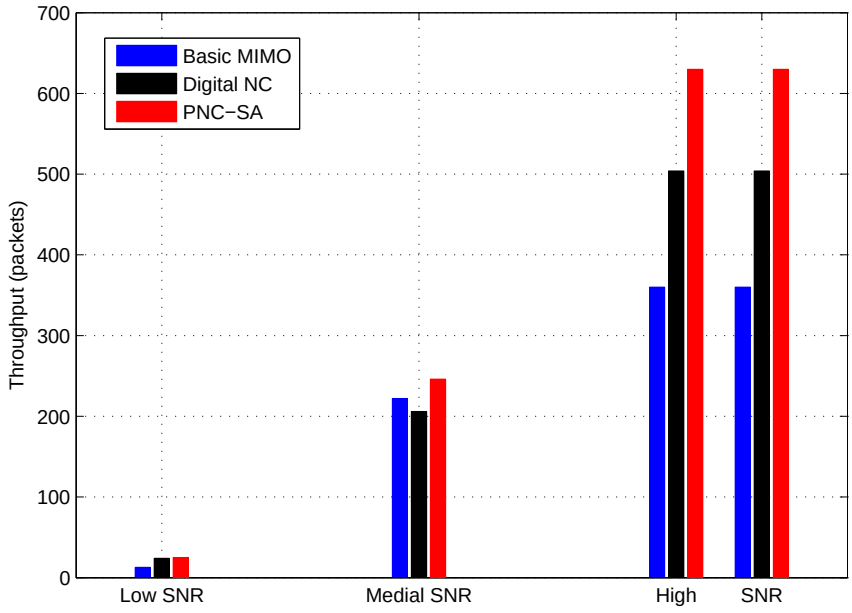

Figure 3.14: Packet-level throughput for multicast, PNC-SA *vs.* DNC *vs.* MIMO alone.

and packet length as previously assumed. At high SNR, PNC-SA again demonstrates a marked throughput gain. DNC slightly leads MIMO at high SNR, but becomes inferior when SNR decreases due to its relatively worse SNR-BER performance.

Cascading SA for Multi-hop Broadcast

In this final application, we show that SA can be applied independently, without coupling with PNC. When signals of distinct packets are aligned to the same direction, PNC demodulation is required; when signals of the same packet are aligned, normal demodulation suffices.

In Fig. 3.15, the sender at the top wishes to broadcast to the entire network, with m rows. Each node has 2 antennas. The source data is divided into 2 packets, x_1 and x_2. The goal is to finish broadcast routing in as few time slots as possible.

The SA solution is rather simple: have each row of nodes transmit concurrently, and disseminate the data item in $m - 1$ rounds. Signals are aligned for reception at inner

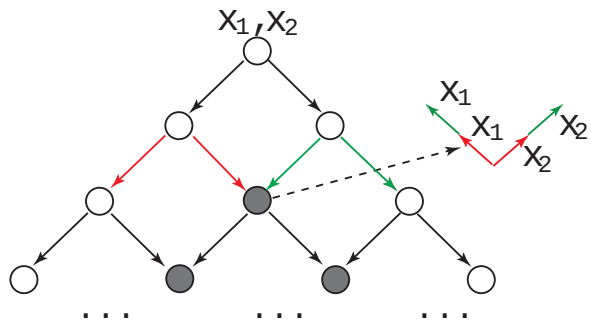

Figure 3.15: Cascading signal alignment for multi-hop broadcast. Note that the two x_1's reinforce each other, since we apply normal BPSK instead of PNC demodulation. Signals are aligned at dark nodes.

nodes in black. The two signals for x_1 (x_2) augment each other, yielding a power gain. For the k nodes at row k, SA is applied in a cascading fashion: we can first decide the precoding vector for the left-most node. Consequently, all other precoding vectors at the same row are determined. Each node aligns its signal according to its neighbor on the left. The number of time slots, $m-1$, is the minimum possible, since under any routing scheme, data can propagate only one row per time slot.

A non-SA solution schedules individual transmissions to avoid interference. It not only takes at least $m-1$ time slots, but also requires a complex scheduling algorithm, in contrast to the simple row-by-row structure of SA. For the same BER, SA does not consume significantly more energy, even by having all nodes except the bottom row transmit. This can be verified by checking the following facts (assume each node transmits with power P in the non-SA solution). (i) In the optimal non-SA solution, each transmission, with power P, covers ≤ 2 nodes. (ii) With SA, each transmission covers > 1 nodes on average. (iii) With SA, for the same BER, only border nodes in white need to transmit at power P. Inner nodes in black can transmit at roughly $P/2$ due to the MISO power gain. (iv) Border nodes only represent a $O(1/m)$ fraction of the network.

3.7 Summary

Signal alignment (SA) was introduced in this chapter. We showed that PNC-SA, SA coupled with PNC, can open new design spaces for routing in MIMO wireless networks, and can hence augment the network capacity region. The design of PNC-SA has been inspired by recent advances in PNC and IA research, yet PNC-SA can better exploit the spatial diversity and precoding opportunities of a MIMO network, for achieving higher throughput. We studied the new problem of optimal precoding introduced by PNC-SA, formulated it into a vector programming problem, and designed a solution for maximizing SNR at the receiver. The SNR-BER performance of PNC-SA was then analyzed. In the second half of this chapter, the analysis of the throughput in a $N \times N \times M$ system was provided. Moreover, general applications of both PNC-SA and SA alone were demonstrated, in various multi-hop MIMO routing scenarios, including information exchange, unicast, and multicast/broadcast. Throughput gain of up to a factor of 2 was observed, compared to simple solutions without coding.

In the next chapter, we will look at another application of PNC in NanoNets. We will introduce a pair-to-pair routing paradigm, Buddy Routing, for both unicast and multicast in multi-hop wireless networks consisting of extremely small nodes, such as NanoNets.

Chapter 4

Buddy Routing: A Routing Paradigm for NanoNets Based on Physical Layer Network Coding

Nanonetworks (NanoNets) represent an emerging type of wireless sensor networks. A nanonetwork consists of nanomachines (nanonodes) — wireless nodes at extremely small form factors, on the order of micrometers or nanometers. Recent developments in nanotechnology allow tiny components to communicate and compute, introducing new applications in the biomedical field, industrial goods, and other areas. This chapter aims to present the first routing/MAC protocol design tailored for multi-hop NanoNets, by utilizing physical layer network coding (PNC) for pair-to-pair routing that breaks through the frugal nodal power limitation at nanonodes. Sec. 4.1 compares two enabling technologies, in terms of multi-hop transmission throughput and single-hop BER. Sec. 4.2 presents theoretical analysis of a BR route in terms of capacity and power consumption. We present our BR algorithms and simulation results in Sec. 4.3 for unicast, and in Sec. 4.4 for multicast. Sec. 4.5 summarizes this chapter.

4.1 Enabling Buddy Routing: PNC *vs.* Amplify&Forward

In Sec. 2.4, we already mentioned the operation process for the pair-to-pair forwarding gadget that is depicted in Fig. 4.1. The pair-to-pair forwarding mechanism underlying Buddy Routing can be enabled by either PNC or Amplify&Forward (A&F). The main difference between PNC and A&F lies in the intra-pair transmission to the Rx node from its buddy: in A&F, it transmits an amplified version of the received analog signal $h_{11}a_1x_1 + h_{21}a_2x_2$: in PNC, the buddy transmits a digital version of $x_1 + x_2$.

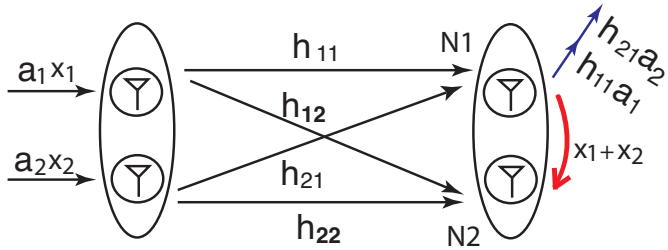

Figure 4.1: Pair-to-pair based buddy forwarding enabled by PNC.

In this section, we compare these two enabling technologies in terms of multi-hop throughput potential (Sec. 4.1.1), single-hop BER (Sec. 4.1.2), and protocol overhead. Note that during the data transmission along the BR route, channel state information is needed at each hop to perform precoding. The discussion of CSI in nanonetworks is similar to the analysis for MIMO networks in Sec. 3.1.

4.1.1 A&F *vs.* PNC : Multi-hop Buddy Routing

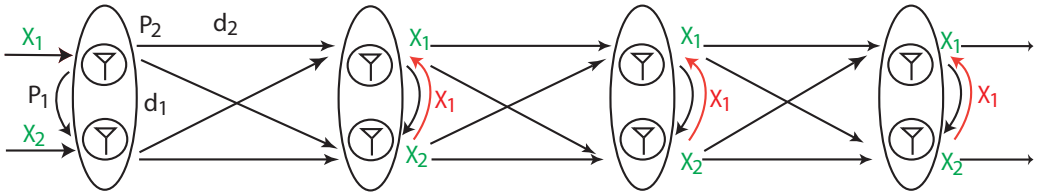

Figure 4.2: BR Transmissions in a multi-hop unicast route enabled by A&F.

Fig. 4.2 depicts the pipeline operation of a BR route enabled by A&F. We highlight that, in order to prepare for the pair-to-pair transmission, intra-pair sharing of a half-packet is required at each hop (labelled by the red arrow). This is an extra step of transmission that does not exist in the PNC-enabled BR route. As a result, an extra time slot is required for scheduling such intra-pair half-packet sharing, leading to a lower end-to-end data throughput.

In contrast, Fig. 4.3 shows the pipeline operation of a multi-hop route based on pair-to-pair forwarding, enabled by PNC. Except at the source pair, there is no need for

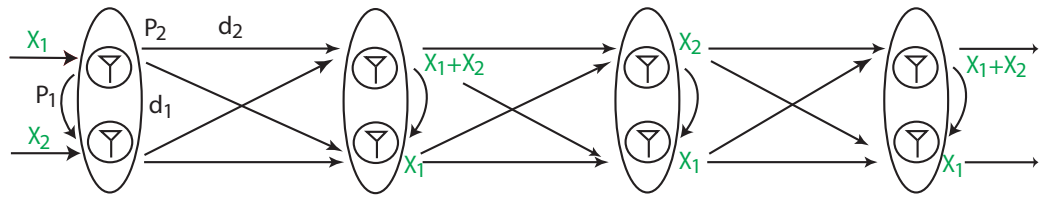

Figure 4.3: BR transmission in a multi-hop unicast route enabled by PNC.

half-packet sharing in subsequent buddy pairs for subsequent pair-to-pair transmission. The top receiver has already demodulated a digital half-packet (labeled in figure) that can be directly used. As a result, all short hop (intra-pair) transmissions can happen simultaneously along the entire BR route, without incurring severe interference.

4.1.2 A&F *vs.* PNC : One-hop BER

We first analyze the BER performance of A&F, and then compare with the BER of PNC. We ignore the BER for the Tx node to share x_1 with its buddy, since it is the same for both schemes, and is relatively small, due to the short distance.

1) BER of Amplify&Forward

The analysis of BER performance for A&F with ML detection is similar to that of a basic 2×2 MIMO link. $N2$ can decode x_1 and x_2 after receiving the amplified signal from $N1$. The vector error rate of A&F is:

$$Pr_s(A\&F) = \frac{1}{4}\sum_{i=1}^{4}\sum_{k=1k\neq i}^{4} Q\left(\sqrt{\frac{\phi_{ik}\rho_A}{2}}\right),$$

where $\phi_{ik} = (\mathbf{l_i} - \mathbf{l_k})^T(\mathbf{l_i} - \mathbf{l_k})$, $\mathbf{l_i}$ and $\mathbf{l_k}$ are two possible spatial source vectors, and $i, k \in \{1, \ldots, 4\}$. ρ_A is SNR at the receiver side. Then the BER of A&F can be approximated as:

$$Pr_b(A\&F) \approx Pr_b(A\&F)/2.$$

During joint ML decoding at $N1$, two SNR values are involved, the SNR for the pair-to-pair transmission, and the SNR to receive the amplified signal. Cor-

respondingly, we plot two BER lines in the simulation: 'A&F-upper' assumes the pair-to-pair BER, 'A&F' assumes the average of the two SNR values.

2) BER of PNC

For the one-hop gadget in Fig. 4.1, the BER performance of PNC can be analyzed in two phases. In phase one, we study the BER at $N1$, for decoding x_1+x_2. In phase two, we study the BER at $N2$ for decoding x_1 and x_2, assuming an adapted version of Maximum-Likelihood (ML) detection [42, 43].

BER at $N1$. $N1$ can demodulate x_1+x_2 by applying ML detection and PNC mapping. Let $\mathbf{m} = x_1 + x_2$, which is in the $\{-2, 0, 2\}$ domain according to PNC mapping under BPSK modulation. Let $\mathbf{m_i}$ and $\mathbf{m_k}$ be two possible transmit vectors, with $i, k \in \{1, 2, 3\}$ being indices to $\{-2, 0, 2\}$. Assuming that $\mathbf{m_i}$ is received, the probability that $N1$ incorrectly outputs $\mathbf{m_k}$ is:

$$Pr(\mathbf{m_i} \rightarrow \mathbf{m_k}) = Q\left(\sqrt{\frac{d_{ik}^2}{2\sigma_{PNC-SA}^2}}\right) = Q\left(\sqrt{\frac{\phi'_{ik}\rho_1}{2}}\right),$$

where $\phi'_{ik} = (\mathbf{m_i} - \mathbf{m_k})^T(\mathbf{m_i} - \mathbf{m_k})$, and ρ_1 is the received SNR at N1. Function Q computes the area under the tail of a Gaussian PDF.

The ternary values in $\{-2, 0, 2\}$ appear in $\mathbf{m}$ with probabilities of 25%, 50%, and 25%, respectively, assuming 0 and 1 are equally likely in the original data packet. $Pr(\mathbf{m_i} \rightarrow \mathbf{m_k}) = 0$ when both $\mathbf{m_i}$ and $\mathbf{m_k}$ are in $(\pm 2, \pm 2)^T$. In other words, judging -2 to be $+2$ or vice versa does not lead to an error in $x_1 + x_2$. $N1$ wishes to demodulate the digital bits $x_1 + x_2$. The average vector error probability, which is also the bit error rate, for $x_1 + x_2$ is

$$Pr_s(x_1 + x_2) = Pr_b(x_1 + x_2)$$

$$= 2P(\mathbf{m_1})Pr(\mathbf{m_1} \rightarrow \mathbf{m_2}) + P(\mathbf{m_2})\sum_{i \neq 2} Pr(\mathbf{m_2} \rightarrow \mathbf{m_i})$$

BER at $N2$. We apply adapted ML, a detection scheme tailored for collaborating PNC receivers recently proposed by us [42], to decode x_1 and x_2. Before applying the normal min-distance criterion in ML, it first filters out the enumerated vectors that are not in agreement with the known values for x_1+x_2, to reduce the computational complexity. Using 16QAM modulation, there are 16 such vectors, with dimension 2×1. $\mathbf{n_i}$ and $\mathbf{n_k}$ are two distinct vectors among the sixteen. Let Λ_c and Λ_w denote the events that N2 receives the correct and wrong data in $x_1 + x_2$ from $N1$, respectively. The average vector error probability when $x_1 + x_2$ is correct is:

$$Pr_s(\mathbf{n}|\Lambda_c) = \frac{1}{16}\sum_{i=1}^{16}\sum_{k=1k\neq i}^{16} Q\left(\sqrt{\frac{\phi''_{ik}\rho_2}{10}}\right).$$

Here $\phi''_{ik} = (\mathbf{n_i} - \mathbf{n_k})^T(\mathbf{n_i} - \mathbf{n_k})$, and ρ_2 is the received SNR at node 2. In the constellation graph with ML decoding, when noise exceeds the decision threshold, only 1 bit will be in error. Thus, the approximate BER can be computed as:

$$Pr_b(\mathbf{n}|\Lambda_c) \approx Pr_s(\mathbf{n}|\Lambda_c)/4.$$

We next analyze the case that x_1+x_2 transmitted from $N1$ contains an error. We have $Pr_b(\mathbf{n}) = Pr_b(\mathbf{n}|\Lambda_c)Pr_b(\Lambda_c)+Pr_b(\mathbf{n}|\Lambda_w)Pr_b(x_1+x_2)$. When information from $N1$ is wrong, $N2$ outputs a wrong vector with probability 1, *i.e.*, $Pr_b(\mathbf{n}|\Lambda_w) = 1$. Therefore the vector error rate of the overall PNC-based scheme is:

$$Pr_b(\mathbf{n}) = Pr_b(\mathbf{n}|\Lambda_c)(1 - Pr_b(x_1 + x_2)) + Pr_b(x_1 + x_2).$$

3) Numerical result of BER

Fig. 4.4 shows the numerical result based on the BER analysis of A&F and PNC. We can observe that the BER of PNC is almost the same as, but slightly worse than, that of A&F, under the same SNR at the receiver side. A small price in BER is paid by the PNC scheme, for involving two steps of demodulation.

To conclude, a PNC-enabled BR route and an A&F enabled BR route have comparable BER performance, while the former leads to a more efficient pipeline operation and

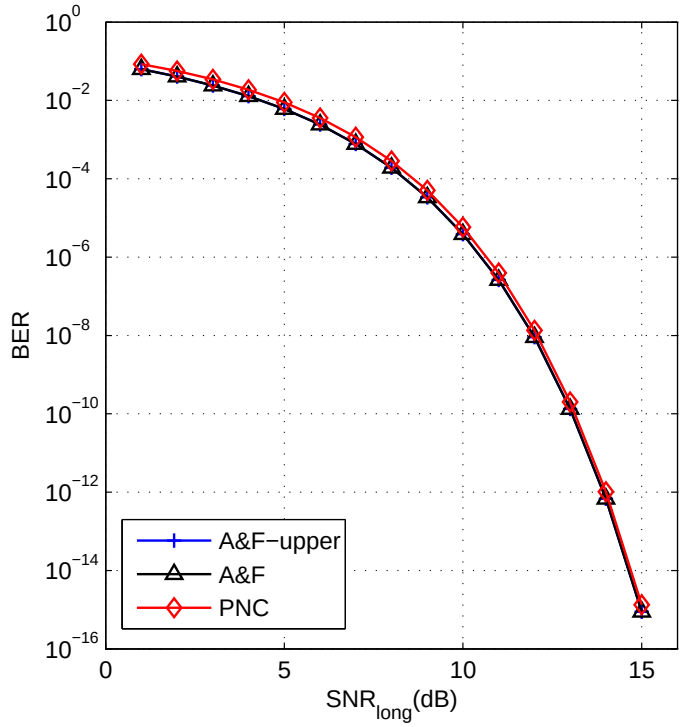

Figure 4.4: PNC *vs.* virtual MIMO, ignoring error in collaborative steps.

a higher end-to-end throughput. In the rest of this chapter, we focus on PNC as the enabling technology of BR routing. While the original proposal of PNC requires extra overhead in symbol-level node synchronization, recent advances show that asynchronous PNC with only packet-level synchronization (required in the TDMA MAC underlying both PNC-based and A&F based schemes) can achieve similar performance, especially when channel coding is appropriately designed [22].

4.2 Theoretical Analysis

4.2.1 System model and parameters

We consider a multi-hop BR route as shown in Fig. 4.5. Let $d_1 = \alpha d_2$, $P_1 = \beta P_2$. $0 < \alpha, \beta \leq 1$. For ease of analysis, we assume in this section that the distance d_1 of each pair-to-pair hop is the same, and the inter-node distance d_2 is the same in each pair.

We can synchronize nodes in the network, and schedule two types of time slots: long slots and short slots. In each long time slot, the long hop pair-to-pair transmissions happen simultaneously every three hops, for mitigating interference (following the two-hop interference range in the protocol interference model [31]). Therefore, three long time slots are required: t_{11}, t_{12} , and t_{13}. Every $(3k+1)$-st long hop transmits in slot t_{11}, every $(3k+2)$-nd long hop transmits in slot t_{12}, and every $(3k+3)$-rd long hop transmits in slot t_{13}. During short time slot t_2, all the intra-pair short hops transmit simultaneously.

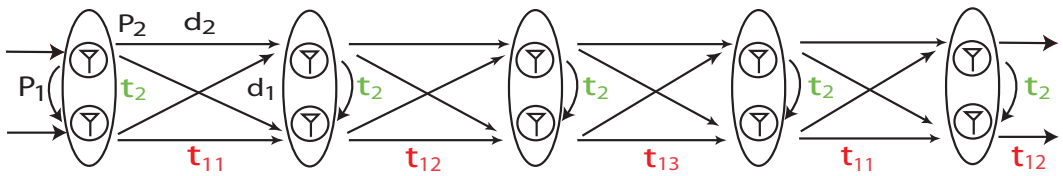

Figure 4.5: BR System Model.

4.2.2 Capacity of a BR Route

To analyze the end-to-end routing capacity of a BR route, we first compute SNR_{short} and SNR_{long}, BER values in the short and long transmissions, respectively.

Assume the path loss exponent is 3 [28], and the distance between a wireless Tx node and Rx node is d. Then the power available at the receiving antenna can be expressed by the power for the transmitting antenna and distance, which is $P_r = P_t/d^3$. Considering interference from immediate neighboring pairs along the BR path, the SNR of the short hop can be approximated as:

$$SNR_{short} = \frac{P_1/d_1^3}{\sigma^2 + 2 \times P_1/d_2^3} \qquad (4.1)$$

Here σ^2 is the intensity of additive white Gaussian noise. Considering interference from the closest two pairs that transmit concurrently in the BR TDMA scheme, the SNR of the long hop can be approximated as:

$$SNR_{long} = \frac{2 \times P_2/d_2^3}{\sigma^2 + 2 \times P_2/(2d_2)^3} \qquad (4.2)$$

According to the Shannon-Hartley Theorem, the capacity of a wireless link l is

$$C_l = B_l \log_2(1 + SNR_l),$$

where C_l is the channel capacity in *bps* and B_l is the bandwidth of the channel in Hertz. The capacity of a k-hop BR route is the bottleneck capacity among all the long (inter-pair) and short (intra-pair) transmissions, at each hop i:

$$C_{BR} = \min\{C_{long-i}, C_{short-i} | 1 \leq i \leq k\}$$

Capacity at very high SNR. We first simulate the BR route capacity with noise ignored. Fig. 4.6 shows that the BR route capacity decreases when $d_1/d_2 > 0.39$. On the other hand, the ratio between P_1 and P_2 has no significant effect on the capacity. In this set of numerical results, $B = 100KHz$, $P2 = 100\mu$W, $d_2 = 50$dm.

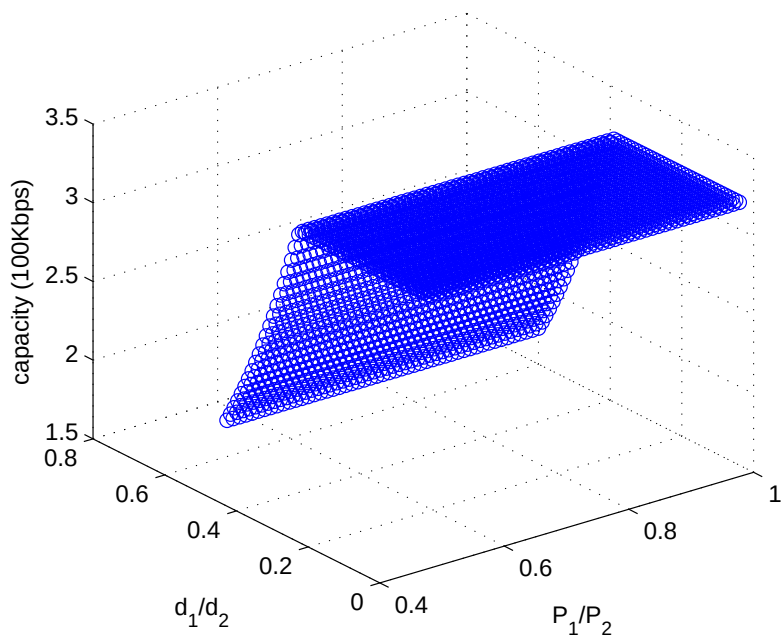

Figure 4.6: BR route capacity with different values for P_1/P_2 and d_1/d_2.

Without background noise, with constant P_2 and d_2, inter-pair link capacity is constant and does not depend on P_1/P_2. When $\alpha = d_1/d_2 < 0.39$, the bottleneck of the BR

route lies in the inter-pair transmissions. When $\alpha > 0.39$, the bottleneck becomes the intra-pair links, whose capacity decreases as d_1 increases.

Capacity with noise considered. We next analyze the capacity of a BR route with noise considered. Fig. 4.7 shows a decreasing trend of the BR route capacity as noise grows. In this set of numerical results, noise intensity varies from 0 to $4 \times 10^{-6}W$, $P_2 = 100\mu$W, $d_2 = 50$dm, $d_1 = 5$dm. The bottleneck resides in the inter-pair transmissions, and changes in $\beta = P_1/P_2$ have no influence on capacity.

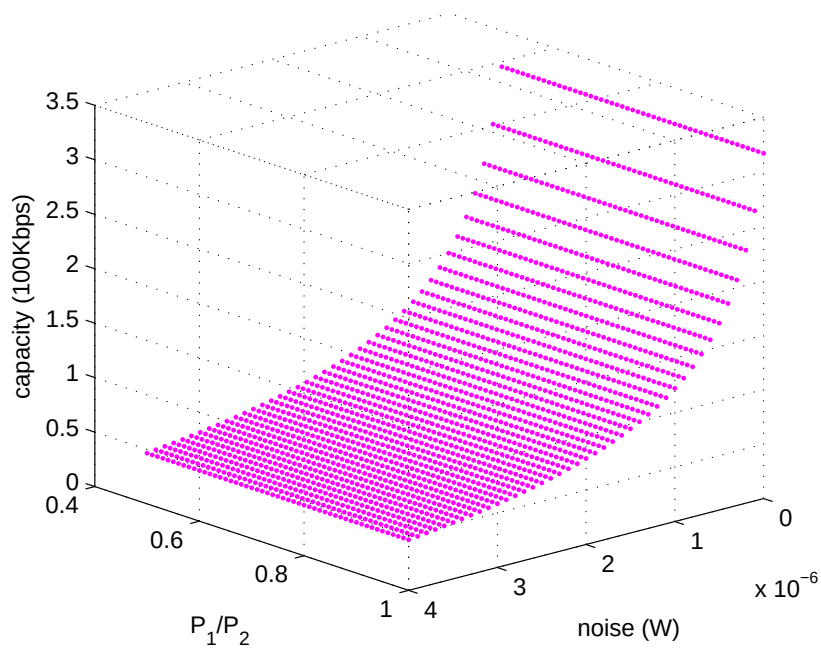

Figure 4.7: Capacity with the effect of noise, $\alpha = 0.1$. BR route bottleneck exists in inter-pair transmissions, P_1/P_2 is irrelevant.

The short hop becomes a bottleneck when $SNR_{short} < SNR_{long}$. Substituting (4.1) and (4.2) into this inequality, we obtain the equivalent condition of

$$\sigma^2 < \gamma, \text{and } \alpha < (\frac{\beta}{2})^{1/3};$$
$$\text{or} \quad \sigma^2 > \gamma, \text{and } \alpha > (\frac{\beta}{2})^{1/3},$$

where $\gamma = \frac{(16-\alpha^{-3})\frac{P_2}{d_2^3}}{4\alpha^{-3}-\frac{8}{\beta}}$.

For the numerical results in Fig. 4.8, σ^2 varies from 0 to $4 \times 10^{-7}W$, $P2 = 100\mu$W, $d_2 = 50$dm, $d_1 = 30$dm. Under such parameter settings, the bottleneck switches to the intra-pair links. Overall BR capacity decreases gradually as the noise level escalates.

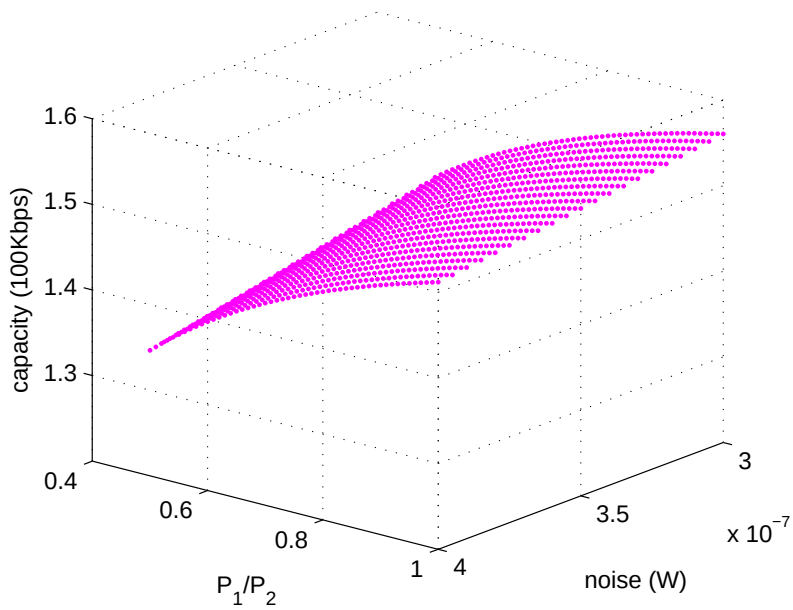

Figure 4.8: Capacity with the effect of noise, $\alpha = 0.6$. BR route bottleneck exists in intra-pair links, P_1/P_2 is relevant.

From Fig. 4.8, we can see that as P_1 increases, the BR route capacity increases. However, for the same amount of information routed, the total power consumption along the entire BR route increases. We therefore face a fundamental tradeoff between capacity and energy efficiency.

4.2.3 Power Consumption: BR *vs.* Point-to-Point Routing

Next, we compare the energy consumption, for routing the same amount of information, between Buddy Routing and traditional point-to-point schemes. Again, we assume that BPSK and 16QAM are selected for modulation in the long and short BR transmissions,

respectively. For point-to-point routing, a single node relays the data packet at each hop, using BPSK modulation. Let t be the time duration for one antenna to transmit one packet with BPSK modulation, and k be the number of (long) hops from the source to the destination. At each hop, the energy consumption ratio between BR and point-to-point routing is

$$\frac{2P_2\frac{t}{2} + 2P_1\frac{t}{8}}{P_2 t} = 1 + \frac{P_1}{4P_2}$$

The ratio of total energy consumption along the entire route is

$$\frac{k(2P_2\frac{t}{2}) + (k+1)(P_1\frac{t}{8})}{kP_2 t} = 1 + \frac{(k+1)P_1}{8kP_2}$$

Fig. 4.9 plots the energy consumption ratio computed above, with $P2 = 100\mu$W, $d_1 = 5$dm, $\alpha = 0.1, d_2 = 50$dm; $k = [2, 4, 8, 12, 30, 50, 100]$ (each corresponding to a line in the figure). The energy consumption ratio decreases when P_1 is smaller, while the value of k doesn't have a great influence on the ratio. Overall, the extra power consumption overhead caused by BR is mostly below 20%, and further decreases to below 5% when $P_1/P_2 < 0.5$. Such a consumption can be well justified by the potential capacity gain of a factor of 2.

4.3 Buddy Routing: Unicast

In this section, we complete the design of a routing/MAC protocol suite, by applying Buddy Routing for unicast in multi-hop wireless networks consisting of extremely power constrained devices, such as NanoNets and smart dust [17]. We describe the overall routing solution, as well as a tailored power and MAC optimization module in Sec. 4.3.1, and present simulation results in Sec. 4.3.2.

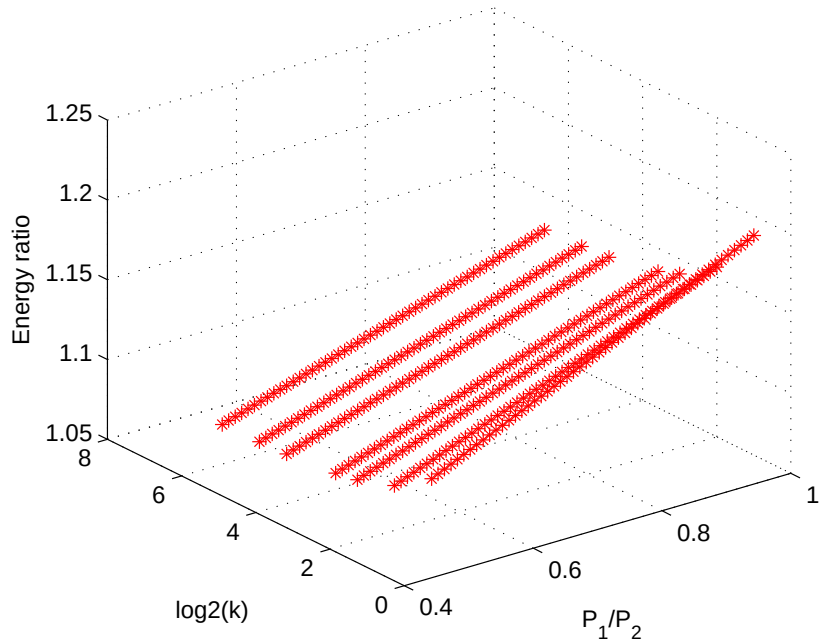

Figure 4.9: Energy consumption ratio of the entire unicast route: BR *vs* point-to-point routing.

4.3.1 The BR Algorithms for Unicast

Table 4.1 presents the algorithms for BR unicast. Here r_b (radius of smallest circle in Fig. 4.10) is the maximum distance between a pair of buddy nodes, r_{min} (medium circle) and r_{max} (large circle) are the minimum and maximum allowed distances between two neighbor buddy pairs, respectively.

The idea behind BR unicast routing is to extend the well-known greedy geographical routing algorithm [18], which is known for its light-weight and fully distributed nature, from the point-to-point domain to the pair-to-pair domain. At each step in the iterative forwarding process, the algorithm looks for a next-hop pair between the two co-axial circles of radius r_{max} and r_{min}, which is closest to the destination. The routing algorithm assumes a relatively dense network, such that the search for a buddy within a pair and the search for a next-hop pair of buddies can succeed. If the network density does not meet such a desired property, a hybrid route that combines pair-to-pair BR routing and

Table 4.1: BR Unicast Algorithms: Routing & MAC Optimization

1. Pair-to-pair greedy geographic unicast routing

find closest neighbor u of source
$pair = \{source, u\}$
while destination $\notin pair$ **do**
  if $dist(pair, destination) \leq r_{max}$:
  find closest neighbor v of destination
  $pair_{next} = \{destination, v\}$
  else:
  find $pair_{next}$, such that $r_{min} \leq dist(pair, pair_{next}) \leq r_{max}$
  and $dist(pair_{next}, destination)$ as small as possible
  end if
  PNC-based pair-to-pair packet transmission: $pair \rightarrow pair_{next}$
  $pair = pair_{next}$
end while

2. Iterative MAC layer optimization

$\delta \leftarrow 1$
while $\delta > \epsilon$:
  2.1. adjust time slot lengths in t_{11}, t_{12}, t_{13} and t_2
  — so that the capacity in each time slot is equal
  2.2. inter-pair power optimization
  — adjust P_2 of bottleneck long BR hop & neighbor pairs
  — achieve equal capacity at bottleneck link & 2 neibghbor links
  2.3. intra-pair power optimization
  — adjust P_1 in bottleneck short BR pair & neighbor pairs
  — achieve equal capacity at bottleneck pair & 2 neibghbor pairs
  — $\delta \leftarrow$ increment in end-to-end capacity due to 2.1-2.3
end while

traditional point-to-point routing can be used instead.

We now consider the complexity of the BR algorithms, for application in a NanoNet. The iterative power refinement is based on simple computation and neighbor communication only. The TDMA MAC is known for its low overhead, when compared to random access based protocols. The greedy geographical routing is stateless and light-weight. However, obtaining and maintaining location information at nanonodes may constitute a considerable overhead, if the NanoNet consists of mobile nodes. Our current design of BR is therefore more suitable for a relatively static network environment. Lastly, while the original proposal of PNC requires symbol level synchronization and accurate estimation

of channel state information, such requirements are relaxed in the latest developments of asynchronous physical layer network coding [22].

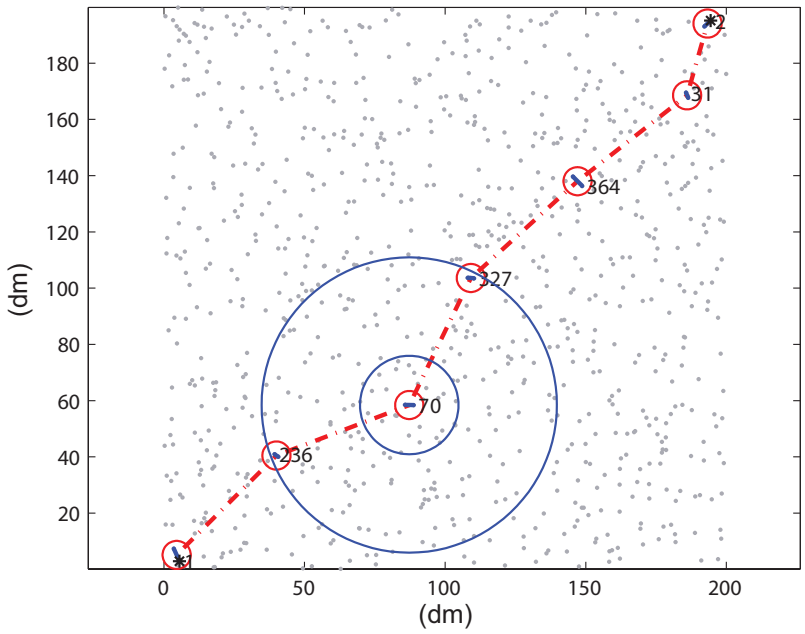

Figure 4.10: BR unicast based on pair-to-pair greedy geographical routing.

Fig. 4.10 depicts a multi-hop unicast route found by the BR unicast routing algorithm. We have generated 800 random nodes in a square region of 200dm×200dm using Matlab. Node 1 is the source while Node 2 is the destination, and they are each marked with an asterisk. If the distance between two nodes is less than or equal to 5dm (r_b), these two nodes can form one pair which is linked by one short blue line and ringed with a red circle. We set r_{min} = 20dm and r_{max} = 50dm. The two blue co-axial circles in Fig. 4.10 indicate when the third pair (Pair 70) looks for the next hop towards the destination, it can only search the area in the ring.

We have further enhanced the algorithm in Table 4.1 with a number of extra functionalities. First, in the case that the last pair of buddies in the BR route (excluding the destination pair) is too close to the destination, it will be discarded and replaced by a

new pair such that the distance between the new pair and the destination is just larger than r_{min}. Second, we further implemented the planar face routing module [18] to enable the greedy geographic routing algorithm to be able to route around a large area devoid of wireless nodes, as shown in Fig. 4.11.

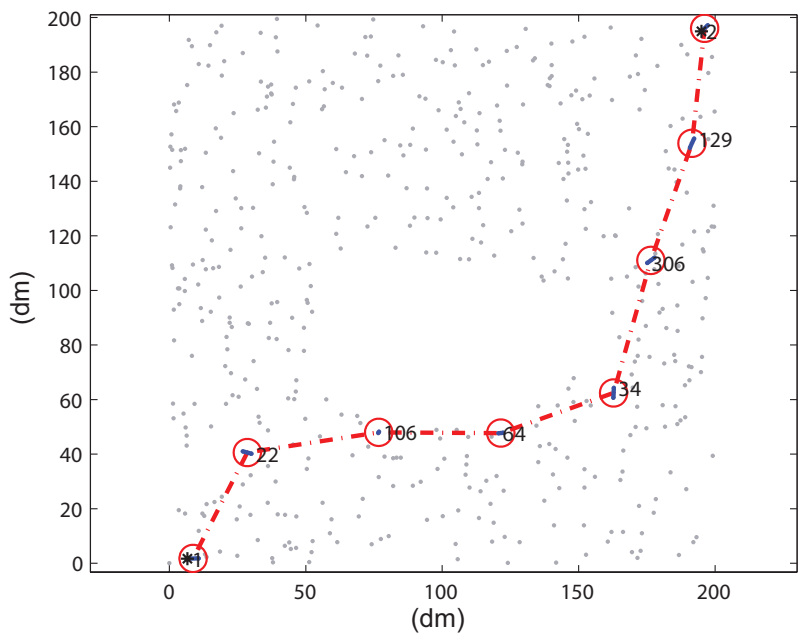

Figure 4.11: BR unicast with Greedy Routing, with planar face routing implemented.

4.3.2 Simulation Results: BR Unicast

In this subsection, we show some numerical results on the route capacity of BR unicast. We set $r_b = 5$dm, $r_{min} = 20$dm, and $r_{max} = 50$dm in all three sets of experiments.

Fig. 4.12 depicts the effectiveness of the MAC optimization module in part 2 of Table 4.1. In this set of simulations, 700 nodes are deployed in the network, each with maximum Tx power of 120μW. The end-to-end capacity of the BR route monotonically increases, and stabilizes after five rounds. The increment in each round is more or less random, and is not monotonic. End-to-end throughput is more than doubled after the iterative power/MAC optimization.

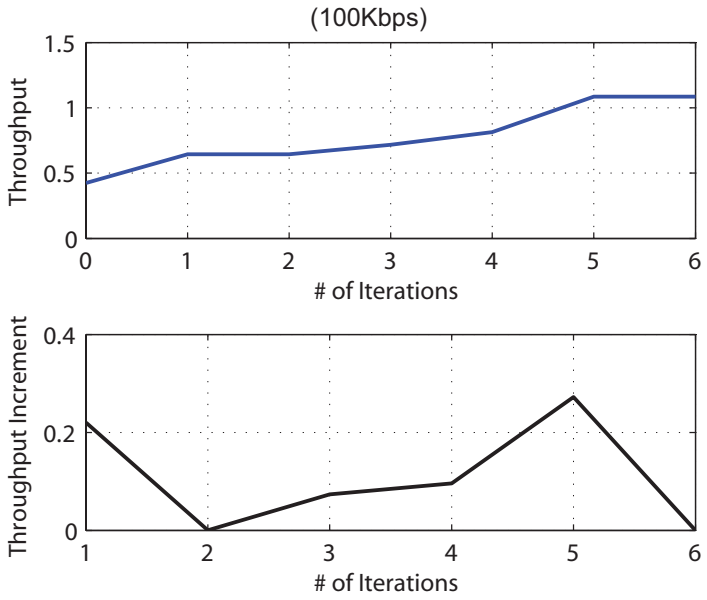

Figure 4.12: BR Unicast. Top: throughput at each round. Bottom: throughput increase at each round. Note that the throughput improvement from round 1 to round 2, although very small, is not zero.

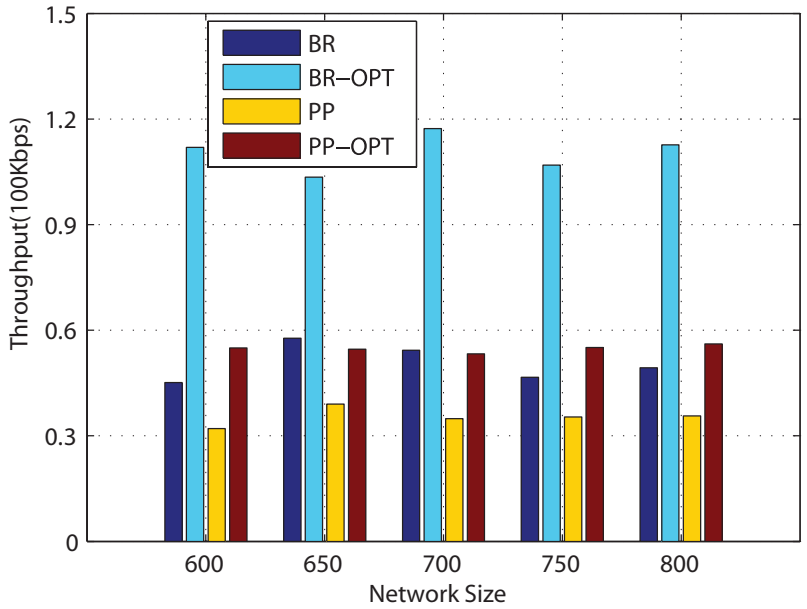

Figure 4.13: BR Unicast, end-to-end throughput comparison, with varying network sizes.

Fig. 4.13 compares the end-to-end throughput of BR with traditional point-to-point routing, both with and without MAC layer optimization, in networks of various sizes. The maximum power available for each node is 120μW. Each throughput is computed as the average of five executions of the routing algorithm in question, over different network topologies. We can see that throughput of buddy routing after optimization is almost twice that of point-to-point routing. The underlying reason for such a gain is simple yet fundamental: the BR gadget in Fig. 4.1 has twice the capacity of a point-to-point link, under equal nodal power budget. Such a significant gain in throughput can well justify the 5% to 20% overhead in power consumption observed in Sec. 4.2.3.

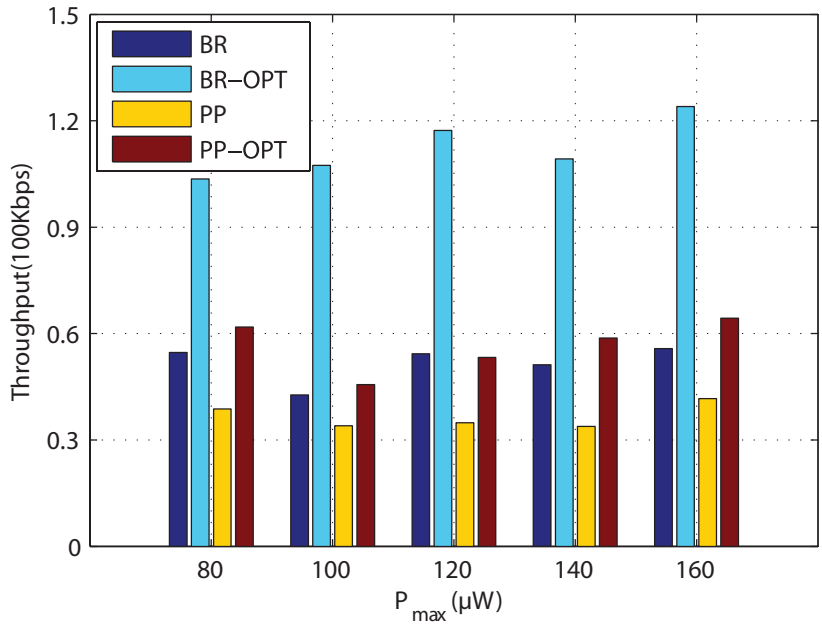

Figure 4.14: BR Unicast, end-to-end throughput comparison, with varying maximum node power.

Fig. 4.14 shows a similar throughput comparison as in Fig. 4.13, with varying maximum node power instead of varying network sizes. A similar throughput gain is observed, which appears to be insensitive to the choice of the maximum node power.

4.4 Buddy Routing: Multicast

The pair-to-pair forwarding mechanism works well in a unicast path, which does not have branches. Multicast models a class of one-to-many data dissemination, where a common data item of interest is to be transmitted to a group instead of a single destination, *e.g.*, along a multicast tree. For multi-hop multicast routing, a new challenge is to replicate a data packet from an upstream node pair to more than one pair, for supporting branching in the multicast tree. A multicast branching gadget based on PNC has been designed accordingly. We introduce this multicast BR gadget in Sec. 4.4.1, apply it to design BR multicast algorithms in Sec. 4.4.2, and perform simulation evaluations in Sec. 4.4.3.

4.4.1 The Multicast BR Gadget

As shown in Fig. 4.15, at each branching node, which has two downstream neighbour buddy pairs, we disseminate the data packet to three nodes in a collaborating group, two of which possess the entire packet (x_1 and x_2), with the third possessing half of the packet (x_1). Precoding is performed at each node as illustrated, such that the following signal alignment [41] at the top node of each node Rx pair is achieved:

$$\begin{cases} h_{11}a1 + h_{21}a_3 + h_{31}a_5 = h_{11}a_2 + h_{21}a_4 \\ h'_{11}a1 + h'_{21}a_3 + h'_{31}a_5 = h'_{11}a_2 + h'_{21}a_4 \end{cases}$$

To successfully align the perceived directions x_1 and x_2 at both the top and bottom pairs simultaneously, we need at least 5 precoding variables, if the two equations above are to have solutions. Consequently, a 3-node group is required at each branching point in the multicast tree.

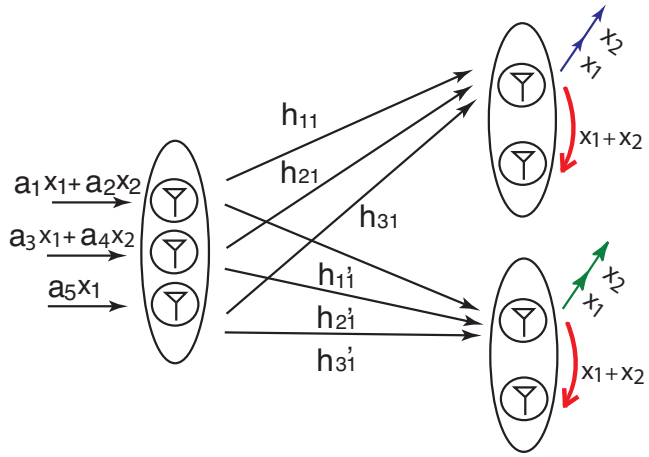

Figure 4.15: PNC gadget for simultaneous group-to-multi-group transmission, for BR multicast.

4.4.2 BR Algorithms: Multicast

The BR multicast algorithms are summarized in Table 4.2. We design a two-tier solution, where a geometric multicast tree algorithm computes the multicast tree topology at the high level (Step 1), then the BR unicast algorithm from Table 4.1 is applied at each tree branch for data forwarding (Step 2). An iterative power/MAC optimization module (Step 3) then follows, similar to the unicast case.

The geometric Steiner tree algorithm starts by including two multicast terminals in the tree, then expands the tree one terminal at a time: a new terminal with shortest total distance to two terminals in the tree is selected, and connected using a local Steiner tree. The algorithm stops when all multicast terminals are covered by the tree. The algorithm guarantees that each node in the tree has degree at most 3, therefore the one-to-two branching capability of the multicast gadget in Fig. 4.15 is always sufficient.

Fig. 4.16 (one-to-four multicast) and Fig. 4.17 (one-to-two multicast) show the multicast trees built by the geometric Steiner tree algorithm, in Step 1 of Table 4.2. There are 950 nodes in Fig. 4.16, and 600 nodes in Fig. 4.17. A 2-node group is connected into a line segment, and a 3-node group at each branching point is connected into a triangle.

Table 4.2: BR Multicast Algorithm Structure

```
1. Geometric Steiner tree construction
   find closest receiver to s, t*
   processed = {s, t*}
   active = T − {t*}
   while active ≠ {} :
     pick t from active, s.t. total distance from t
     to two closest nodes in processed is minimum
     let u, v be the two closet nodes in processed to t
     connect t to u and v through the Fermat point
     if u or v has degree 3: remove from processed set
     active ← active − {t}; processed ← processed + {t}
   end while
2. For each edge in multicast tree built in 1:
   for each node u in tree:
     if degree of u is 2: identify pair
     else: identify triple
   apply BR unicast algorithms for routing between two ends.
3. Iterative MAC layer optimization
```

We adjust the r_b to 8dm in the simulations because it is more likely to group three nodes together at the branch point. In Fig. 4.16, on the top of the right branch, when pair 93 chose the next hop, pair 457 is not the nearest one to the destination. When the greedy routing algorithm is running, if it found that the pair with the shortest distance to the destination is too close, then it will pick another pair instead.

The BR multicast algorithm also contains an iterative MAC optimization module, after routing is performed. Tx power and time slot lengths are adjusted for improving end-to-end multicast throughput. The operations here are similar to that in the unicast case. The main difference is that at a branching node group in the multicast tree, neighboring node pairs/triples along different branches of the tree are taken into consideration, when adjusting power and time slot lengths.

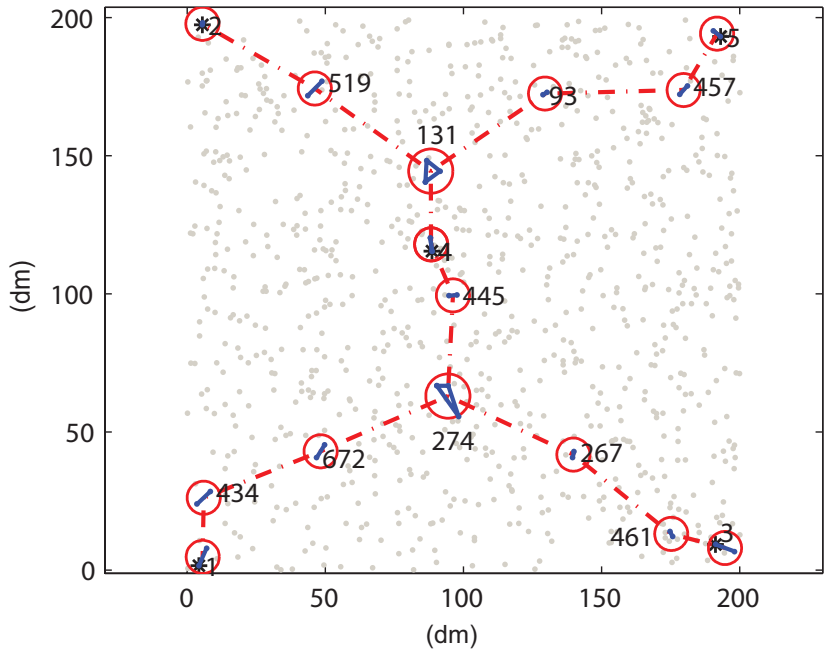

Figure 4.16: BR Multicast with geographic tree construction, one-to-four multicast.

4.4.3 Simulation Results

Fig. 4.18 shows the end-to-end multicast throughput increase during each round of the MAC layer optimization. Three out of 900 nodes in the network are multicast terminals. The maximum power available at each node is 160μW. A similar trend to that in the unicast case is observed: the multicast throughput stabilizes after a small number of rounds. The multicast throughput monotonically increases during the optimization, although the amount of improvement in each round is not monotonic.

Fig. 4.19 shows the comparison of end-to-end multicast throughput between BR multicast and point-to-point multicast, both with and without MAC layer optimization. The maximum power available at each node is 160μW. The number of terminals is 3. Network size varies from 750 to 950 nodes. The dark blue and light blue column indicate the original throughput of buddy routing and the final throughput after adjustment, while yellow

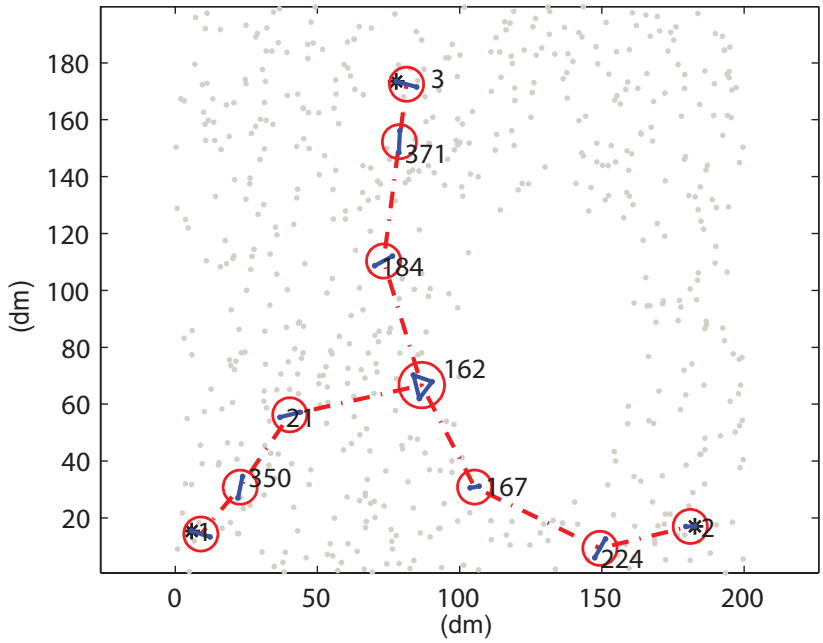

Figure 4.17: BR Multicast with geometric tree construction, one-to-two multicast in a network with large void.

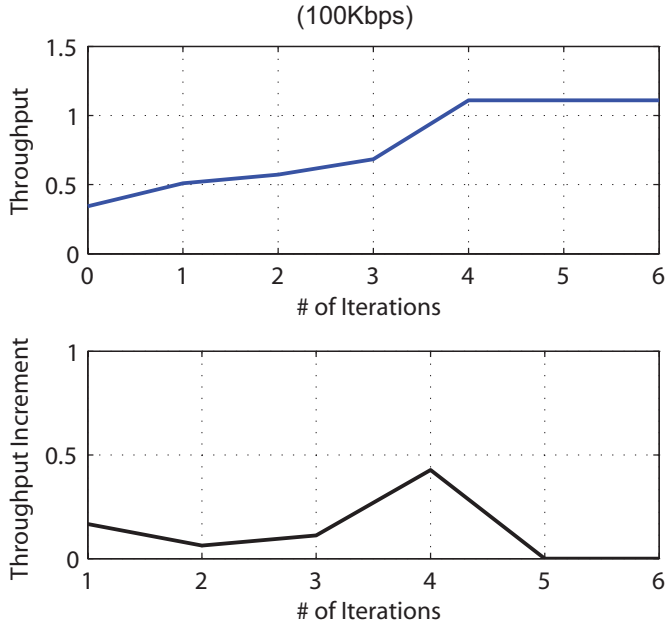

Figure 4.18: BR Multicast. Top: throughput at each round. Bottom: throughput increase at each round.

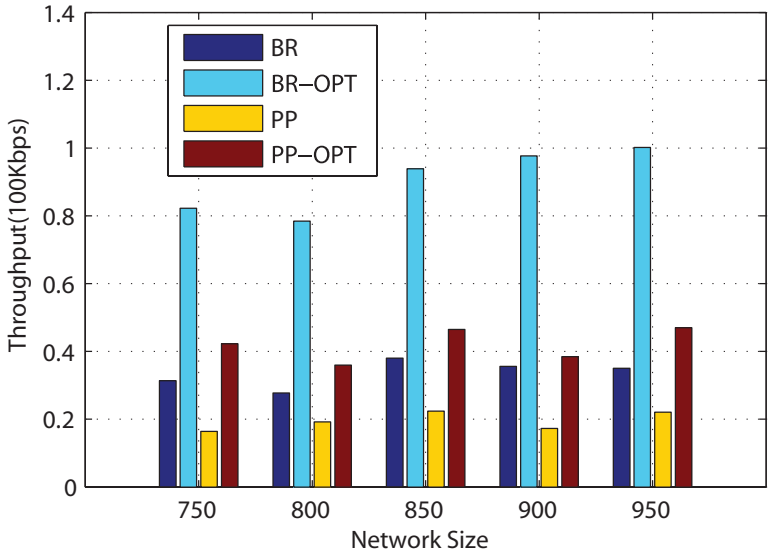

Figure 4.19: BR multicast: end-to-end multicast throughput comparison with point-to–point schemes, under different network sizes.

and red show the throughput of point to point routing. Each data point is the average of five simulation runs. We can see that the throughput of BR multicast is close to twice that of point-to-point multicast, and that the MAC layer optimization significantly improves the achievable throughput, through (a) mitigating interference at bottleneck links, and (b) intelligently adjusting Tx time slot lengths. Achievable multicast throughput appears to slightly increase as the network size grows, since more nodes in the network implies better choices are possible for tree construction and node pair/triple formation.

Fig. 4.20 shows a similar comparison of multicast throughput, but under varying maximum Tx power instead of varying network size. The throughput of BR multicast is roughly twice that of point-to-point routing. There are 900 nodes in the network, with three multicast terminals.

Fig. 4.21 provides a throughput comparison with varying sizes of the multicast group. There are 900 nodes in this network. The maximum power available at each node is 160μW. An increase in the number of multicast receivers, in the same network environ-

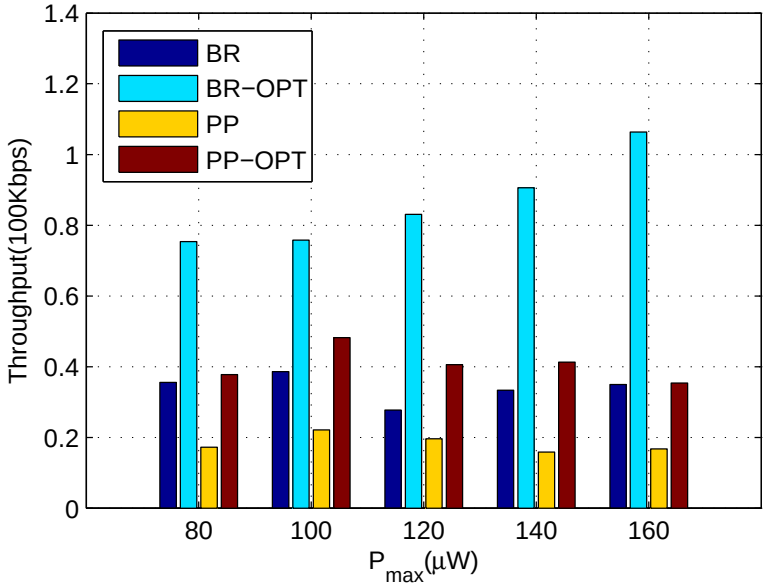

Figure 4.20: BR multicast: end-to-end throughput comparison with point-to-point schemes, under different maximum Tx power.

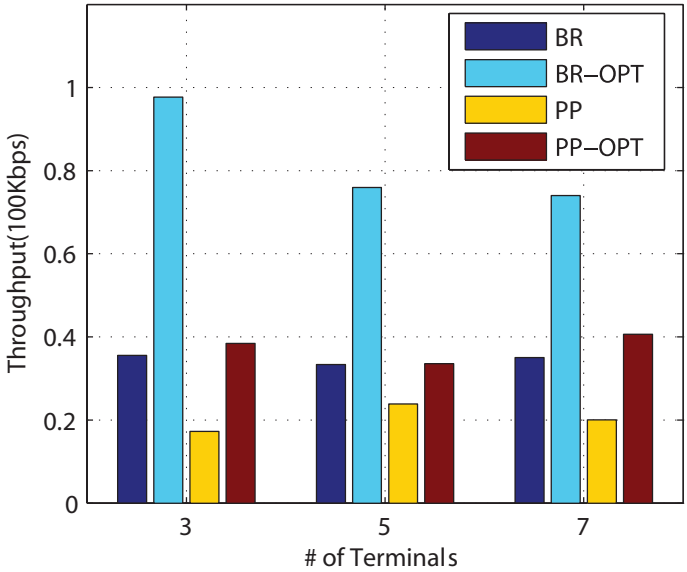

Figure 4.21: BR Multicast, end-to-end throughput comparison with growing multicast group size.

ment, usually leads to a decrease in achievable multicast throughput, since the multicast tree involves more branches that incur interference. Nonetheless, in each case, BR multicast can still manage to achieve roughly twice the throughput of point-to-point multicast.

4.5 Summary

New wireless sensor networks with extremely small and power-limited devices, as exemplified by NanoNets, are envisioned to play an important role in our future lives. We proposed a new routing paradigm tailored for such networks. Buddy Routing groups weak wireless nodes into groups for collaborative data forwarding, based on a recent technique, physical layer network coding. By paying a moderate price in energy efficiency (energy consumed per bit in end-to-end transmission), BR has a potential to break through the nodal power limit in NanoNets, substantially improving the unicast and multicast throughput, as verified by our theoretical analysis and simulation results. The ideas from BR can also be brought back to benefit normal wireless networks, since it points out a possible solution for the improvement of the throughput.

In the next chapter, we summarize the contributions in this thesis and discuss directions for the future work.

Chapter 5

Conclusion and Future Work

5.1 Thesis Summary

In this thesis, we have advocated the use of physical layer network coding (PNC) to improve the network performance in large, general, multi-hop wireless networks. Our focus has been on two distinct applications of PNC in MIMO wireless networks and NanoNets. In the former case, we have shown how PNC can be applied with signal alignment (SA), a new wireless communication technique inspired from interference alignment, for applications in MIMO wireless networks. In particular, we studied the detailed PNC-SA scheme design, and conducted BER comparison between PNC-SA and IAC. We also studied the degree of freedom of a general $N \times N \times M$ system and the applications of PNC-SA in multi-hop MIMO networks. In the case of NanoNets, we proposed Buddy Routing (BR), a PNC-enabled collaborative routing paradigm. We first compared two technologies that can realize pair-to-pair forwarding: PNC versus Amplify&Forward (A&F). We also analyzed the capacity and power consumption of BR. Furthermore, our proposed BR unicast and multicast algorithms are evaluated through simulation results, verifying that they can improve the throughput effectively. We next conclude the contributions made by this thesis.

In Chapter 3, we presented the formulation of the optimal PNC-SA precoding problem in the 2-client 2-AP MIMO system. The solution showed that the precoding vectors at the client side can be determined by the channel matrix, power budget, and alignment direction. Then demodulation at the AP side was discussed, where we can use PNC mapping with ML or ZF detection to decode the combined signal, $x_1 + x_3$ and $x_2 + x_4$,

at AP1, then apply adapted ML decoding to recover all the original packets at AP2. Through analysis, we observed the BER performance of PNC-SA was slightly better than that of IAC. In addition, we proved two theorems on the degree of freedom of PNC-SA, which indicated the maximum number of packets a $N \times N \times M$ system can deliver. Finally, we applied our PNC-SA scheme in more general transmission scenarios. Packet-level simulations demonstrated that PNC-SA outperformed digital network coding and basic MIMO at high SNR, for information exchange, unicast and multicast/broadcast applications.

In Chapter 4, we first considered two different physical layer techniques that can enable Buddy Routing, PNC and A&F. Although the BER performance of A&F was slightly better than that of PNC, A&F requires one more time slot for scheduling intra-pair half-packet sharing, resulting in lower end-to-end data throughput. Next, we focused on the capacity and power consumption of BR. We showed details on the tradeoff between capacity and energy efficiency. BR can increase the capacity by a factor of 2 while only consuming 20% extra power, compared with point-to-point routing. We presented our pair-to-pair greedy geographic unicast algorithm, and the iterative MAC layer optimization scheme that can adjust the power and time slot lengths in order to maximize throughput. We also extended the BR unicast algorithm to multicast scenarios. We designed a geometric multicast tree algorithm, with BR unicast applied at each tree branch, and iterative power and slot length optimization. Simulation results showed that the throughput of BR is almost twice that of point-to-point routing in both unicast and multicast scenarios.

5.2 Future Work

While this thesis offers a comprehensive analysis of two new PNC applications, PNC-SA and BR, there are still many interesting questions that are worth investigating in future research. We conclude the thesis by listing such problems that we have identified.

- In Chapter 3, we have studied the PNC-SA precoding formulation at the client side. Considering the alignment constraint and the power budget, we proposed the SNR-maximizing solution for the precoding vectors in Sec. 3.2. As mentioned at the end of this section, the current solution simply focuses on SNR at AP1, while we could formulate a global optimization that further considers SNR and BER at AP2. Furthermore, the optimal precoding problem can be formulated and studied for a general N-user M-antenna MIMO system.

- Software-defined radio [7] is a radio communication system where some or all of the physical layer functions are implemented by software on the computing devices. Traditional hardware based radio systems limit cross-functionality since it can only be modified by physical intervention. By contrast, software-defined radio realizes many new functions that used to be only theoretically possible. Recently, Lu *et al.* [24] implemented physical layer network coding (PNC) in a 3-node GUN radio testbed, with software-defined radio. PNC can be viewed as a special case of PNC-SA. It is interesting to implement and study the PNC-SA scheme in a software-defined radio system.

- In Chapter 4, we designed the BR unicast and multicast algorithms. In the current solution, the first step is the pair-to-pair greedy routing algorithm that finds a path from the source to the destination. Then the next step is to optimize the MAC layer for maximizing the throughput. We can see that the path is fixed after the first step. A possible direction for future research is to design a cross-layer optimization

approach, which iteratively refines both the route and the MAC layer parameters during each round.

Bibliography

[1] R. Ahlswede, N. Cai, S. Li, and R. Yeung. Network Information Flow. *IEEE Transactions on Information Theory*, 46(4):1204–1216, July 2000.

[2] I. Akyildiz, F. Brunetti, and C. Blázquez. Nanonetworks: A New Communication Paradigm. *Computer Networks*, 52(12):2260–2279, 2008.

[3] I. Akyildiz and J. Jornet. Electromagnetic Wireless Nanosensor Networks. *Nano Communication Networks*, 1(1):3–19, 2010.

[4] B. Atakan and O. Akan. Carbon Nanotube-based Nanoscale Ad Hoc Networks. *Communications Magazine, IEEE*, 48(6):129–135, 2010.

[5] J. Chitode. *Digital Communications*. Technical Publications, 2009.

[6] R. Cruz and A. Santhanam. Optimal Routing, Link Scheduling and Power Control in Multihop Wireless Networks. In *Proc. of IEEE INFOCOM*, San Franciso, CA, USA, March 30-April 3, 2003.

[7] B. Fette. Software-defined Radio, 2003. US Patent App. 10/449,597.

[8] C. Fragouli, J. Le Boudec, and J. Widmer. Network Coding: An Instant Primer. *ACM SIGCOMM Computer Communication Review*, 36(1):63–68, 2006.

[9] M. Gastpar and M. Vetterli. On the Capacity of Wireless Networks: The Relay Case. In *Proc. of IEEE INFOCOM*, New York, USA, June 23-27, 2002.

[10] S. Gollakota, S. Perli, and D. Katabi. Interference Alignment and Cancellation. In *Proc. of ACM SIGCOMM*, Barcelona, Spain, August 17-21, 2009.

[11] R. Gummadi, L. Massoulie, and R. Sreenivas. The Role of Feedback in the Choice between Routing and Coding for Wireless Unicast. In *Proc. of IEEE Symposium on Network Coding (NetCod)*, Toronto, Canada, June 9-11, 2010.

[12] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, 2002.

[13] P. Hoeher and F. Tufvesson. Channel Estimation with Superimposed Pilot Sequence. In *Proc. of IEEE Global Telecommunications Conference*, Rio de Janeiro, Brazil, December, 1999.

[14] C. Hu, X. Wang, D. Nie, and J. Zhao. Multicast Scaling Laws with Hierarchical Cooperation. In *Proc. of IEEE INFOCOM*, San Diego, CA, USA, March 15-19, 2010.

[15] K. Jensen, J. Weldon, H. Garcia, and A. Zettl. Nanotube Radio. *Nano Letters*, 7(11):3508–3511, 2007.

[16] D. Johnson and D. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. *Mobile Computing*, pages 153–181, 1996.

[17] J. Kahn, R. Katz, and K. Pister. Mobile Networking for Smart Dust. In *Proc. of ACM Mobicom*, Seattle, WA, USA, August 15-20, 1999.

[18] B. Karp and H. Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proc. of ACM Mobicom*, Boston, MA, USA, August 6-11, 2000.

[19] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft. XORs in the Air: Practical Wireless Network Coding. *IEEE/ACM Transactions on Networking*, 16(3):497–510, June 2008.

[20] Z. Li, B. Li, and L. Lau. A Constant Bound on Throughput Improvement of Multicast Network Coding in Undirected Networks. *IEEE Transactions on Information*

Theory, 55(3):997–1015, March 2009.

[21] L.Li, R. Alimi, D. Shen, H. Viswanathan, and Y. Yang. A General Algorithm for Interference Alignment and Cancellation in Wireless Networks. In *Proc. of IEEE INFOCOM*, San Diego, CA, USA, March 15-19, 2010.

[22] L. Lu and S. Liew. Asynchronous Physical-Layer Network Coding. *IEEE Transactions on Wireless Communications*, 11(2):819–831, February 2012.

[23] L. Lu, S. Liew, and S. Zhang. Optimal Decoding Algorithm for Asynchronous Physical-layer Network Coding. In *Proc. of IEEE International Conference on Communications*, Kyoto, Japan, June 05-09, 2011.

[24] L. Lu, T. Wang, S. Liew, and S. Zhang. Implementation of Physical-layer Network Coding. *Elsevier Physical Communication Journal*, 2012.

[25] A. Ozgur, O. Leveque, and D. Tse. Hierarchical Cooperation Achieves Optimal Capacity Scaling in Ad Hoc Networks. *IEEE Transactions on Information Theory*, 53(10):3549–3572, October 2007.

[26] C. Perkins and E. Royer. Ad-hoc On-demand Distance Vector Routing. In *Proc. of 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, USA, February 25-26, 1999.

[27] R. Ramanathan and R. Rosales-Hain. Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment. In *Proc. of IEEE INFOCOM*, Tel Aviv, Israel, March 26-30, 2000.

[28] T. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, 1996.

[29] A. Scaglione, D. Goeckel, and J. Laneman. Cooperative Communications in Mobile Ad Hoc Networks: Rethinking the Link Abstraction. *Distributed Antenna Systems, Auerbach Publications*, pages 87–111, 2007.

[30] K. Sengijpta. Fundamentals of Statistical Signal Processing: Estimation Theory. *Technometrics*, 37(4):465–466, 1995.

[31] Y. Shi, Y. Hou, and J. Liu. How to Correctly Use the Protocol Interference Model for Multi-hop Wireless Networks. In *Proc. of ACM Mobihoc*, New Orleans, LA, USA, May 18-21, 2009.

[32] M. Souryal and B. Vojcic. Performance of Amplify-and-forward and Decode-and-forward Relaying in Rayleigh Fading with Turbo Codes. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France, May 14-19, 2006.

[33] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, Cambridge, UK, 2005.

[34] R. Wattenhofer, L. Li, P. Bahl, and Y. Wang. Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks. In *Proc. of IEEE INFOCOM*, Anchorage, AK, USA, April 22-26, 2001.

[35] J. Weldon, K. Jensen, and A. Zettl. Nanomechanical Radio Transmitter. *Physica Status Solidi (b)*, 245(10):2323–2325, 2008.

[36] Y. Wu, P. Chou, and S. Kung. Information Exchange in Wireless Networks with Network Coding and Physical-layer Broadcast. In *Proc. of Conference on Information Sciences and Systems*, Washington, D.C, USA, March 16-18, 2005.

[37] J. Zhan, B. Nazer, M. Gastpar, and U. Erez. MIMO Compute-and-Forward. In *Proc. of IEEE International Symposium on Information Theory (ISIT)*, Seoul, Korea, June 28-July 3, 2009.

[38] S. Zhang and S. Liew. Channel Coding and Decoding in a Relay System Operated with Physical-layer Network Coding. *IEEE Journal on Selected Areas in Communications*, 27(5):788–796, 2009.

[39] S. Zhang and S. Liew. Physical-Layer Network Coding with Multiple Antennas. In *Proc. of IEEE Wireless Communications and Networking Conference*, Sydney, Australia, April 18-21, 2010.

[40] S. Zhang, S. Liew, and P. Lam. On The Synchronization of Physical-Layer Network Coding. In *Proc. of IEEE Information Theory Workshop (ITW)*, Punta del Este, Uruguay, March 13-17, 2006.

[41] S. Zhang, S. Liew, and P. Lam. Physical-Layer Network Coding. In *Proc. of ACM MobiCom*, Los Angeles, CA, USA, September 24-29, 2006.

[42] R. Zhou, Z. Li, C. Wu, and C. Williamson. Physical Layer Network Coding with Signal Alignment for MIMO Wireless Networks. In *Proc. of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Valencia, Spain, October 17-22, 2011.

[43] R. Zhou, Z. Li, C. Wu, and C. Williamson. Buddy Routing: A Routing Paradigm for NanoNets Based on Physical Layer Network Coding. In *Proc. of the 21st International Conference on Computer Communication Networks (ICCCN)*, Munich, Germany, July 30-August 2, 2012.