

THROUGHPUT OPTIMIZATION IN MIMO NETWORKS

A Thesis
Presented to
The Academic Faculty

by

Ramya Srinivasan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2011

THROUGHPUT OPTIMIZATION IN MIMO NETWORKS

Approved by:

Professor Douglas M. Blough, Advisor
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Raghupathy Sivakumar
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Mary Ann Ingram
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Bonnie H. Ferri
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Professor Mostafa Ammar
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Date Approved: 08-11-2011

ACKNOWLEDGEMENTS

I am grateful to my friends and family- Ritu Srivastava, Selcuk Uluagac, Doug Strait, Abhinav Dalal, Rajeev Gadre, Amisha Manek, Rhetta Meryl, the Straits- Tom, Alison and Benjamin, Apurva Mohan, Luis Miguel Cortés-Peña, David Bauer, Brian Hayes, Daniel Lertpratchya, Mike Sun, Peter Kingston (Good and Evil), Greg Philips, Eliana Christina, Moazzam Khan, Yusun Chang, Pan Zhao, Amit Warke, James Sakalaukus, Aditi Kulkarni, Sanjana Rao, Shobhana Jayashankar, Sudu Mama, Shammi Prasad, Meera Simharajan, Simha Srishylam, Sujata Saraf, Vikram Iyengar, Vidya Aunty, Mohan Mama, Nate Spadafore, Mihaela Luncian, Tom Robbins, Sophia Fischer, Yonca Toker, Alper Akanser, Nisa Fachry, Val Jones and last but not least, Rodney and Tascha (dogs - the only beings aside from one's parents, who are capable of unconditional love!).

My heartfelt thanks to Ritu, my childhood friend - a wonderful girl; to Doug for his advice, friendship and support; to Abhinav, for sticking by me during all my ups and downs and for pushing me to apply myself and succeed; to Selcuk for being kind, funny, slightly eccentric and very helpful; to Amisha for her constant friendship and to Rajeev for being a caring, fun-loving and generous person. Special thanks to David Bauer for being a good friend and a helpful labmate and for assisting me at various points with solving programming problems that I ran into.

I dedicate this work to my family. Although at times I have distanced myself from them, I could never have successfully completed the PhD program without the love and support of my Mother, Father and Sister. I am indebted to my Father for encouraging me in my academic pursuits, to my Mother for her constant affection, cheery disposition and optimism, and to my little Sister for always being a devoted

friend, especially when times were rough, and for helping me with innumerable things (particularly when I leave things for the last minute!).

Finally, I am very grateful to Professor Blough for his excellent and diligent guidance in helping develop my scientific and technical skills. However, the example he set through his fineness of character is, in the grand scheme of things, the more valuable lesson that I bring away with me from this experience.

The greatest challenge to any thinker is stating the problem in a way that will allow a solution.

- Bertrand Russell

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iii
LIST OF TABLES	ix
LIST OF FIGURES	x
SUMMARY	xii
I INTRODUCTION	1
1.1 MIMO Enabled Networks	2
1.2 MIMO Network Throughput Optimization	4
1.3 Physical Layer and Network Layer Models	5
1.4 Problem Statement	7
1.4.1 Feasibility	8
1.4.2 Stream Allocation	8
1.4.3 One Shot Link/Stream Scheduling	9
1.5 Contributions	9
1.6 Thesis Organization	11
II BACKGROUND AND SYSTEM MODEL	13
2.1 Basic MIMO Principles	13
2.2 Channel State Information and Interference Suppression	16
2.3 MIMO Degrees of Freedom Model	20
2.4 Network Model	22
2.4.1 Interference Model	22
2.4.2 Rates and Streams	24
2.5 Summary of Modeling Assumptions	26
III RELATED WORK	27
3.1 MIMO Communication Theory	27
3.1.1 Interference handling	28
3.1.2 Information theoretic analysis and upper bounds	33

3.1.3	Degrees-of-Freedom (DOF) approach	34
3.2	MIMO network wide throughput optimization	35
3.2.1	Stream Allocation/Scheduling	36
3.2.2	Cross Layer Optimization of MIMO WMNs, MIMO-aware Routing and MAC-based Approaches	39
3.3	Feasibility	40
IV	PROBLEM DEFINITION	42
4.1	Feasibility	42
4.2	Stream Allocation	44
4.3	One-Shot Link/Stream Scheduling	45
V	FEASIBILITY: THEORETICAL RESULTS	47
5.1	Matrix Formulation For MIMO Feasibility Checking	48
5.2	Results on Single Collision Domain	50
5.2.1	Spatial Multiplexing with IC: A Matrix Formulation	51
5.2.2	Lagrange Multiplier Method of Optimization: SR+SM	54
5.2.3	Structure of the Matrices at Optimal Point: SR+SM	58
5.2.4	Handling Primary Interference	59
5.2.5	Optimal total number of streams in an SCD with Uniform Rate Model	59
5.2.6	An Example	61
5.2.7	Discussion	62
5.3	Complexity of MIMO Feasibility	64
5.3.1	Receiver-Side Suppression	64
5.3.2	Maximum Antenna Array Size $K = 2$	66
5.3.3	Maximum Antenna Array Size $K=3$	69
5.4	Conclusion	78
5.4.1	Conclusions on SCD	78
5.4.2	Conclusions on Multi-hop Networks	79

VI FEASIBILITY: EXPERIMENTAL RESULTS	80
6.1 Feasibility Heuristics	80
6.1.1 Simple Greedy and Extended Greedy	80
6.1.2 Accuracy of Greedy Heuristics for Uniform Antenna Arrays .	81
6.1.3 Accuracy of Greedy Heuristics for Non-uniform Antenna Arrays	82
6.1.4 isFeasible3 Heuristic for $K \leq 3$	83
6.1.5 Accuracy of isFeasible	84
6.2 Conclusion	84
VII STREAM ALLOCATION AND ONE-SHOT LINK SCHEDULING	88
7.1 Stream Allocation Algorithm: StreamMaxRate	89
7.2 One-Shot Link Scheduling Algorithms	92
7.3 Stream Allocation and One-Shot Scheduling Results for Single Collision Domain	93
7.3.1 Simulation Set-up	93
7.3.2 Simulation Results	94
7.4 Stream Allocation Algorithm: MultihopMaxThrpt	96
7.5 Stream Allocation Results For Multi-hop Networks	99
7.5.1 Simulation Set-up	99
7.5.2 Simulation Results: Uniform Array Size	100
7.5.3 Simulation Results: Non-uniform Array Size	102
VIII PARTIAL RESULTS AND FUTURE DIRECTIONS	108
8.1 Stream Allocation: Independent Connected Component Partitioning	108
8.1.1 PartitionedStreamMaxRate Algorithm	109
8.1.2 Simulation Set-Up	110
8.1.3 Simulation Results	112
8.1.4 Drawbacks Of PartitionedStreamMaxRate	114
8.2 Complexity of Verifying Feasibility for Arbitrary MIMO Networks .	115
8.2.1 Open Problem	116

8.2.2	Efficient Feasibility Check for $K = 4$	116
8.3	Integrating Physical Layer Operations with the Networking Layer . .	117
8.4	Practical Applications of this Work	117
IX	CONCLUSION	120
	Bibliography	122

LIST OF TABLES

1	Link rates of a sample link (in <i>Mb/sec</i>) for different numbers of DOFs and streams ($k_i^t = k_i^r = 4$)	25
2	No.of active streams with SR only and with SR+SM ($k = 8$)	58
3	Output of the Lagrange multiplier optimization method.	62
4	Average running times (per sample) for 24 links	106

LIST OF FIGURES

1	An Infrastructure/Backbone Wireless Mesh Network.	3
2	Field with $N = 2 - 18$ nodes uniformly distributed. variable-rate model, best antenna(red); uniform-rate model, best antenna(pink); variable-rate model, random antenna(blue); uniform-rate model, random antenna(cyan)	7
3	Spatial Multiplexing on a MIMO Link	15
4	Interference Alignment	31
5	Beamforming and Suppression	32
6	Three link example where every node must use some DOFs for interference suppression.	37
7	Interference Cancellation between Two Links l_i and l_j	57
8	Interference Cancellation Assignment for $l = 7$	58
9	(a) Communication graph in a single collision domain (b) Maximum matching of the communication graph	61
10	Optimal total number of streams achievable with $l = 7$ links and increasing number of antenna elements, with SR only, SM only, and SM+SR MIMO systems.	63
11	Illustration of definition: $DO(v, v')$	72
12	Algorithm Extended Greedy	81
13	Failure Rates of Simple Greedy and Extended Greedy Heuristics: SCD, uniform array size	82
14	Failure Rates of Simple Greedy and Extended Greedy Heuristics: SCD, non-uniform array size	83
15	Failure Rates of Simple Greedy and Extended Greedy Heuristics: SCD, array size= 4 or 8	84
16	Algorithm isFeasible	85
17	Failure rates of feasibility heuristics: $cgd = 5.5$	86
18	Failure rates of feasibility heuristics: $cgd = 7.5$	86
19	Initialization Procedure for Algorithm StreamMaxRate	91
20	Algorithm StreamMaxRate	91

21	Update Rule for Algorithm StreamMaxRate	92
22	Throughput vs number of nodes for randomly selected matchings . .	94
23	Throughput vs number of active links for two different methods of selecting matchings	96
24	Algorithm MultihopMaxThrpt	99
25	Throughput vs no. of nodes for randomly selected matchings:cgd = 2.5	103
26	Throughput vs no. of nodes for randomly selected matchings:cgd = 5.5	103
27	Throughput vs no. of nodes for randomly selected matchings:cgd = 7.5	104
28	Throughput vs number of nodes for randomly selected matchings: cgd = 2.5; Variable K case: K values can be anywhere between 2 and 6 .	104
29	Throughput vs number of nodes for randomly selected matchings: cgd = 5.5; Variable K case: K values can be anywhere between 2 and 6 .	105
30	Throughput vs no. of nodes for randomly selected matchings: $K=3$, cgd=5.5	106
31	PartitionedStreamMaxRate Algorithm	110
32	PartitionedSMR: neighbors of connected components identified and common neighbors removed.	111
33	PartitionedSMR: maximal IS of neighbors selected to augment partitions	111
34	findRoot procedure	111
35	Type 4 grid: links of length equal to one hop length and two hop lengths	113
36	Throughput and running time from PartitionedSMR for type 4 grid as a function of number of partitions	113
37	Throughput and running time from PartitionedSMR for type 4 grid as a function of the number of partitions	114
38	Throughput and running time from PartitionedSMR for type 4 grid as a function of the number of partitions	115

SUMMARY

Enabling multi-hop wireless mesh networks with multi-input multi-output (MIMO) functionality boosts network throughput by transmitting over multiple orthogonal spatial channels (spatial multiplexing) and by performing interference cancellation, to allow links within interference range to be concurrently active. Furthermore, if the channel is in a deep fade, then multiple antenna elements at the transmitter and/or receiver can be used to transmit a single stream, thereby improving signal quality (diversity gain).

However, there is a fundamental trade-off between boosting individual link performance and reducing interference, which must be modeled in the process of optimizing network throughput. This is called the *diversity-multiplexing-interference suppression trade-off*. Optimizing network throughput therefore, requires optimizing the trade-off between the amounts of diversity employed on each link, the number of streams multiplexed on each link and the number of interfering links allowed to be simultaneously active in the network.

We present a set of efficient heuristics for one-shot link scheduling and stream allocation that approximately solve the problem of optimizing network throughput in a single time slot. We identify the fundamental problem of verifying the *feasibility* of a given stream allocation. The problems of general link scheduling and stream allocation are very closely related to the problem of verifying feasibility.

We present a set of efficient heuristic feasibility tests which can be easily incorporated into practical scheduling schemes. We show for some special MIMO network scenarios that feasibility is of polynomial complexity. However, we conjecture that in general, this problem, which is a variation of Boolean Satisfiability, is NP-Complete.

CHAPTER I

INTRODUCTION

The fundamental limit to the capacity of a wireless channel is shown by Shannon's capacity equation, to be imposed by the signal-to-noise ratio (SNR) of the channel and the bandwidth of the channel [4]. Achieving capacities that approach Shannon's theoretical limit is the main challenge that wireless communication technologies face.

The capacity of a single-input single-output channel corrupted by an additive white Gaussian noise (AWGN), at a level of SNR denoted by ρ , having a bandwidth B and a normalized, random channel gain h can be written as

$$C = B \log_2(1 + \rho|h^2|) \text{ Bits/Sec}$$

Telecommunication systems are continually striving to improve performance and spectral efficiency. Efforts in this direction include OFDM, CDMA, space-time coding, opportunistic communication and MIMO (multi-input multi-output), to name a few. The most promising of these is MIMO, where transmitters and/or receivers are equipped with multiple antenna elements.

The traditional way to achieve higher data rates is by increasing the signal bandwidth. Unfortunately, increasing signal bandwidth of a communications channel by increasing the symbol rate of a modulated carrier increases its susceptibility to multipath fading. Solutions like OFDM partially remedy this problem.

On the other hand, the underlying concept of MIMO communication is that the effective SNR of the system can be increased by transmitting independent data streams simultaneously on orthogonal spatial dimensions over the same physical channel (frequency spectrum). This is called *spatial multiplexing*. A rich scattering environment provides independent transmission paths from each transmit antenna to each receive

antenna. A MIMO link with k_t transmit antennas, k_r receive antennas and with a $k_r \times k_t$ channel normalized matrix H and unit bandwidth has a capacity equal to

$$C = \log_2 \left[\det \left(I_{kr \times kt} + \frac{\rho}{N} HH^* \right) \right] \text{ Bits/Sec}$$

where ρ , is the average SNR at any receiving antenna. For a large number $k_t = k_r = K$ of antennas and a channel with rich multipath, the average value C_{av} of this capacity increases linearly with M .

$$C_{av} \approx K \log_2(1 + \rho|h^2|) \text{ Bits/Sec} \quad (1)$$

Therefore in theory, with an ideal full rank channel matrix, arbitrarily large capacities can be realized if a large number of antenna elements can be employed.

1.1 MIMO Enabled Networks

Current telecommunication standards such as WiMax, Wifi, HSPA, 4G and wireless mesh networks (WMNs) are increasingly adopting MIMO functionality.¹ In particular, wireless mesh networks are emerging as a revolutionary technology that enables seamless connectivity across large areas, entire cities even.

Wireless mesh networks are a subclass of ad hoc networks with the distinguishing feature being that the routing nodes are stationary, unlike in wireless ad hoc networks (MANET's). WMNs lend themselves to self-configuration, self-healing, self-management and self-optimization with the added benefit of not having to deal with rapidly changing channel conditions owing to the stationariness of the routers (see Figure 1).

In the discussion above, the spatial multiplexing (SM) capability of MIMO was seen to deliver enormous capacity gains. It involves exploiting the spatial dimensions

¹Interestingly, even wired technologies such as ITU-T G.9963 are adopting MIMO capabilities.

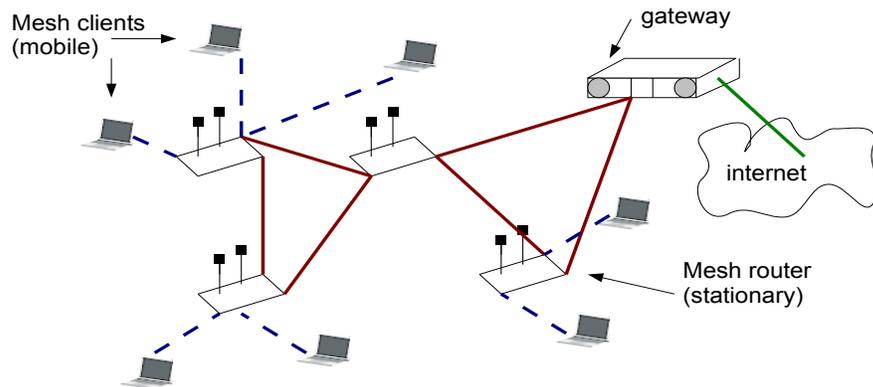


Figure 1: An Infrastructure/Backbone Wireless Mesh Network.

of the communication link by using channel information at the receiver (and transmitter) to recover independent data streams sent on the MIMO channel. A major issue to be dealt with in wireless networks is the capacity reduction due to interference between concurrently transmitting links within interference range of one another [6]. In addition to spatial multiplexing, another primary advantage of MIMO communication is the ability to cancel interference between simultaneously active interfering links. With adequate channel state information, signal processing at the transmitter and/or receiver of a MIMO link can be used to decode each of the transmitted streams on the given link while canceling interference from surrounding interfering links.

Therefore, equipping routers in a wireless mesh network with multiple antennas greatly boosts network throughput through 1) increasing the rates on individual links by allowing multiple parallel data streams to be multiplexed over orthogonal spatial channels and through 2) allowing links within interference range to be simultaneously active by performing interference cancellation (IC).

The past two decades have seen an extensive amount of research activity in the areas of MIMO channel capacity characterizations, MIMO channel interference handling techniques, cross-layer optimizations in interference limited wireless mesh networks with MIMO links, MIMO-aware MAC layer link scheduling.

1.2 MIMO Network Throughput Optimization

Applying spatial multiplexing on individual MIMO links increases the rates on the links by approximately a multiplicative factor equal to the number of antenna elements used. Furthermore, a MIMO array can be used to improve the bit error rate (BER) by using multiple antenna elements to transmit a single data stream. This is called *spatial-diversity gain* and is advantageous when the channels are subject to fades. Network-wide optimization of MIMO resources improves the throughput by a greater amount as compared to applying link-by-link optimization ([54], [49], [89], [8]). A relatively small amount of work has been done on network-wide optimization of MIMO systems. In this thesis, we consider the problem of maximizing throughput in an arbitrary multi-hop MIMO network in a single time slot. One of the key factors that distinguishes the MIMO network optimization problem from optimization of traditional single-antenna systems is the capability of MIMO links to perform IC. Eliminating interference allows greater spatial reuse, which increases the overall capacity of a wireless network. However, the use of MIMO resources for interference suppression by a link reduces the resources available to maximize the link's individual capacity through spatial multiplexing and/or its BER through spatial diversity [45]. Thus, there is a fundamental trade-off between boosting individual link performance and reducing interference, which must be modeled in the process of optimizing network throughput. This is called the *diversity-multiplexing-interference suppression trade-off* on which very little work has been done [65, 68]. In this thesis, it is precisely this three-way trade-off that we model and characterize. We do this while accounting

for variable-rate streams on individual MIMO links, as well as across different links (see section 1.3).

1.3 Physical Layer and Network Layer Models

A relatively small amount of work has been done on network-wide optimization of MIMO systems. Characterizing a complete MIMO communication system with simple, yet accurate, abstractions for the complex behaviors of MIMO links is necessary for the tractability of a network level optimization. The majority of MIMO networks research has used a “degrees of freedom” (DOF) model [23], which accounts for only two capabilities of a MIMO link: spatial multiplexing and interference suppression [5]. Antenna elements provide DOFs that can be divided arbitrarily between spatial multiplexing and interference cancellation (see Chapter 3 for detailed discussion). This model does not account for data rates that vary with SINR, i.e. the data rate is assumed to be fixed and there is a single SINR threshold above which, the fixed data rate is achieved on the link and, below which, no communication is possible. This is called the *uniform rate model*. Therefore, for MIMO networks research to blossom, new models are necessary that can more completely capture the complex and multi-faceted operation of MIMO.

In this work, we have remedied this shortcoming by employing a model ([45, 61]) that accounts for the fact that when a MIMO link (with channel matrix H), multiplexes a number s of streams such that $s < \min(k_t, k_r)$, then the transmitter can allocate power to the s strongest channel modes. The receiver correspondingly sets its weight vector so that the s streams each benefit from a gain equal to the s largest eigen-values of HH^* , where H^* is the complex conjugate of H (see Chapter 7 in [5]). These operations require both the transmitter and the receiver to have complete knowledge of the channel. These eigen-values or equivalently, stream gains, are not equal and, for moderate-to-low SNR, they can have quite large disparities, even in the

presence of interference [67]. Therefore the incremental aggregate capacity of a MIMO link becomes progressively smaller as the number of spatially multiplexed streams is increased. This is called the *variable rate model* [54, 50]. The Figure 2 shows how employing the variable-rate model enables us to achieve a higher throughput as compared to when the uniform-rate model is used.

The channel is modeled as an idealized rich scattering static environment, which corresponds to a quasi-static flat Rayleigh fading channel model. Therefore, the channel has i.i.d. complex, zero mean, unit variance elements as described by [58]. The gain of each channel matrix is calculated using Friis transmission equation and the log-distance path-loss model with a path-loss exponent of 3 ([59, 70, 71, 2]). We assume channel state information is available to the transmitters and therefore include optimal power allocation in our rate calculations. The data rate is calculated from Shannon’s capacity formula using the optimal power allocation [60, 5].

As discussed above, the data rate on a MIMO link is a function of the channel between the transmitter t and the receiver r , the number of streams s multiplexed on the link, the number of DOFs, $ADOF_t$, available at t for multiplexing s streams and the number of DOFs, $ADOF_r$, available at r for multiplexing s streams. This data rate is modeled as a rate function $R(t, r, ADOF_t, ADOF_r, s)$. *No assumption is made on the rate function*, that is, it can be any arbitrary function.

For the purpose of simulation however, we chose to approximate the data rate as follows. We first perform antenna selection and then find the optimal data rate as described above. Suppose the transmitter uses t antenna elements for transmission of s streams and that the receiver uses r antenna elements for reception. We perform best eigen-value selection by picking the t transmit elements and the r receive elements that maximize the data rate of the link. We then calculate the rates for the case of $1 \leq s \leq \min(t, r)$ streams by allocating power through the best s eigen-modes of the $t \times r$ channel.

The variable-rate model is the basis for our analytical and experimental results throughout this thesis. We tackle the MIMO network throughput optimization problem by 1) modeling the spatial multiplexing-interference suppression trade-off based on the degrees-of-freedom² approach and by 2) modeling the spatial-diversity gain based on the variable rate concept. The problem to be solved in this thesis is defined in the next section.

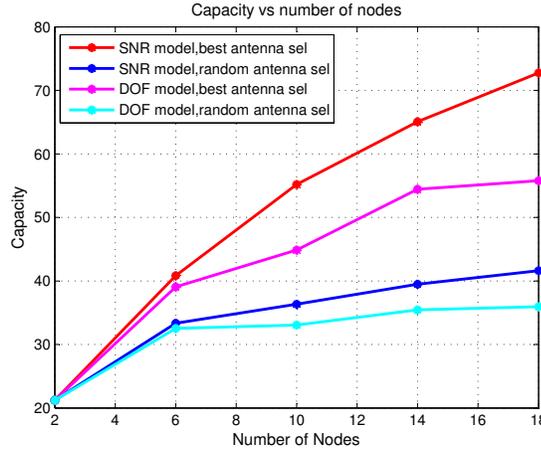


Figure 2: Field with $N = 2 - 18$ nodes uniformly distributed. variable-rate model, best antenna(red); uniform-rate model, best antenna(pink); variable-rate model, random antenna(blue); uniform-rate model, random antenna(cyan)

1.4 Problem Statement

The problem at large, is to determine which links of a MIMO network, when scheduled together concurrently, will maximize the aggregate throughput. This is a version of network scheduling in a MIMO setting where the goal is to maximize the throughput delivered in a single scheduling slot. We identify three inter-related problems discussed below [54].

²The DOF model was briefly discussed above and is described in detail in Section 2.3.

1.4.1 Feasibility

The problem of feasibility is to determine if a set of links can be concurrently active such that all individual transmissions are successful, under a given interference model.

In the case of non-MIMO systems, this problem reduces to determining whether there exists a pair of links in the given set that interfere with one another. Since the number of link pairs is polynomial, checking feasibility in non-MIMO systems is a polynomial time operation.

On the other hand, the interference suppression capability of MIMO links makes the problem of checking feasibility highly complex. For transmitters and receivers to both have interference handling capability, channel information must be known at both ends. This is the assumption we make in this thesis. Thus, the problem of determining whether all interference in an arbitrary MIMO network can be eliminated, is an important one³.

1.4.2 Stream Allocation

In this thesis, we assume that each MIMO node is equipped with a single radio. This means that a node can participate in only one transmission at a time, either as transmitter or as receiver. A set of communication links is said to be *primary-interference-free* if a node that is assigned to be a transmitter is not also assigned to be a receiver in the same time slot and vice-versa. Links that are primary-interference-free have the potential to be scheduled concurrently. The stream allocation problem is to find an optimal stream vector over a given set of primary-interference-free links. An optimal stream vector is defined as a feasible stream vector with maximum aggregate transmission rate.

³However, in the case that channel information is available only at the receivers, the problem of feasibility in the MIMO network reduces to a polynomial time operation. We prove this result in Section 5.3.

1.4.3 One Shot Link/Stream Scheduling

In the stream allocation problem, a set of primary-interference-free links is given. However, in classical one-shot link scheduling, the problem is to determine which links, when scheduled together concurrently, will maximize the aggregate rate. In other words, this is a version of the general link scheduling problem in which the goal is to maximize the data throughput in a single scheduling slot.

1.5 Contributions

The primary contributions of this thesis are

1. A mathematical model that enables an elegant analysis of the MIMO feasibility problem is developed. Feasibility checking is shown to be a variation of the Boolean satisfiability problem.
2. An analytical expression for the optimal total number of streams of a MIMO single collision domain (network where all active links mutually interfere) is derived. This optimal is achieved by optimizing the spatial multiplexing-interference suppression trade-off.
3. Several special cases of the MIMO network scenarios are identified, for which verifying feasibility can be done in polynomial time. These are for arbitrary multi-hop networks a) when interference suppression is done only on the receiver side but not on the transmitter side and b) when the maximum array size of all nodes is at most equal to two.
4. A necessary and sufficient condition for feasibility is derived for the specific case of arbitrary symmetric multi-hop networks for which the maximum array size of all nodes is equal to three and all links carry one stream.

5. We find that the problem of verifying feasibility is a variation of Boolean satisfiability and is very likely to be NP-complete. Since most link scheduling algorithms require the repeated testing for feasibility of stream allocation vectors, and since feasibility is such a complex problem in the MIMO setting, efficient heuristic tests for feasibility are needed. We have designed three efficient heuristic feasibility tests which can be incorporated within practical MIMO scheduling algorithms.
6. Four heuristics for the optimal stream allocation problem are developed and their performance is compared against the absolute optimal (obtained by brute force calculations). These heuristics are an important contribution because of the following. In general, existing stream allocation and link scheduling algorithms are iterative procedures which perform a feasibility test in each iteration. While this is computationally tractable for single-antenna systems, in the case of MIMO systems, the approach becomes impractical owing to the complexity of checking feasibility. Our stream allocation heuristics provide a practical solution to this. They are designed so as to employ the efficient feasibility test(s) that we developed, as a sub-routine in each iteration. The stream allocation heuristics are flexible in that they can be used with any feasibility test in general. These procedures are accurate and computationally efficient approximations to solving stream allocation problem for throughput maximization. Although we have not studied the general link scheduling problem in this thesis (only one-shot link scheduling was studied), our stream allocation heuristics can be used as a starting point in developing a link schedule.
7. From the study of the stream allocation algorithms, we made the important observation that their performance is very sensitive to the accuracy of the feasibility tests that they employ. This is a direct motivation for developing accurate

MIMO-network feasibility tests - a subject that has not received adequate attention from the research community.

1.6 Thesis Organization

The remainder of this thesis is organized as follows.

- Chapter 2 provides some necessary background material on the basic principles of MIMO communication. The underlying system and network models are developed and the set of modeling assumptions made are listed and justified.
- Chapter 3 discusses important prior work related to MIMO communication theory; techniques for interference suppression; approaches towards solving the network throughput optimization problem such as stream allocation, link scheduling, routing, stream control and cross-layer optimizations.
- Chapter 4 develops a detailed mathematical formulation of the problems to be solved, namely, the problems of verifying feasibility, optimal stream allocation and one-shot link scheduling for throughput optimization.
- Chapter 5 presents a mathematical framework for casting the problem of feasibility as well as analytical results on feasibility for both the MIMO single collision domain and for arbitrary multi-hop MIMO networks. These results include analytical expressions for optimal throughput/optimal stream allocation and complexity analyses of verifying feasibility in various network scenarios.
- Chapter 6 develops heuristic methods of verifying feasibility, presents experimental results and performance evaluation of different feasibility tests.
- Chapter 7 develops various approximate methods of optimizing one-shot network throughput by stream allocation and link scheduling. Performance of

these methods in terms of accuracy and computational cost is studied through experimental results.

- Chapter 8 addresses work that was done during the course of writing this thesis, that delivered promising results but which still has to be refined before being reliably applied. Specific challenges that must be overcome in this respect are discussed. Interesting open problems that resulted from our investigations during the course of this thesis and recommendations for future work are discussed.
- Chapter 9 A summary of the conclusions drawn from this research work is discussed.

CHAPTER II

BACKGROUND AND SYSTEM MODEL

A wireless network wherein the deployed nodes are equipped with multiple antenna elements is called a MIMO network. MIMO capabilities, provided by the antenna elements present at each end of a MIMO link, include delivering array gain, diversity gain, spatial multiplexing, and performing interference cancellation. The goal of this research is to use the MIMO resources available on individual links to optimize overall network performance, rather than to separately optimize link-by-link performance, which can lead to sub-optimal network performance. This chapter describes the principles of MIMO communication and establishes the system model employed in this research work.

2.1 Basic MIMO Principles

MIMO links are equipped with digital adaptive array antennas at both ends. A MIMO communication link consists of k_t transmit antennas and k_r receive antennas. The components of a MIMO system are 1) a transmitter with a space-time modulator that maps bits to space-time codewords, 2) a matrix propagation channel H that is a function of the wireless environment, and 3) a space-time receiver that uses an estimate of the propagation channel to decide on the transmitted bit stream. A practical assumption that is made in the literature is that the transmission bandwidth is much smaller than the coherence bandwidth of the channel and that the block length is much smaller than the coherence time. In this case the propagation channel H from the transmitter to the receiver can be described as a $k_r \times k_t$ matrix whose k^{th} column describes the complex scalar coefficients from the k^{th} transmit antenna to each receive antenna. In ideal, rich scattering environments, H is of full rank i.e.

the channel between any given transmit antenna and receive antenna is independent from the channel between all other transmit-receive antenna pairs. In this case, the transmitter and receiver antenna arrays can be adaptively used to boost the link rate by spatial multiplexing independent data streams. In addition to the spatial multiplexing gain, MIMO links can also be used to deliver array gain and diversity gain in the wireless channel and to eliminate interference in the surrounding region [1].

Array gain occurs in an array receiver when the desired signal parts of each antenna element add coherently while noise adds incoherently. Thus, the signal strength is roughly multiplied by the number of antenna elements k_r , while the noise strength is not substantially increased. This results in a factor of k_r increase in signal-to-noise ratio (SNR) after signal combination, compared to the average SNR at an individual antenna element.

Spatial multiplexing gain results when a transmitter transmits multiple streams at the same time using its multiple antenna elements. In the simplest case, the transmitter sends a different stream on each antenna element (up to k_t streams). In this case, each element at the receiver will receive a superposition of the transmitted streams. However, if the receiver has estimated the channel conditions through reception of a training sequence, it knows how the signals from different transmit antenna elements are affected by the channel and it can reconstruct the original streams. In this manner, up to $\min(k_t, k_r)$ streams can be spatially multiplexed on a single MIMO link (see Equation 1). This is illustrated in Figure 3.

In contrast to spatial multiplexing where the aim is to boost throughput, the aim of *spatial-diversity gain* is to enhance reliability by minimizing the channel fluctuations due to small scale fading. The core idea behind diversity gain is to send and/or receive multiple copies of a single data stream ([1, 3]). When fading is uncorrelated between all pairs of antenna elements, the probability that the channel between every transmit-receive antenna pair is in a deep fade is very small. In the ideal case, when

In the case that the channel is perfectly known at the transmitter, then transmit diversity can be employed to improve reliability by an order of as much as k_t . If the receiver has one antenna element, the resulting capacity is given by

$$C = \log_2 \left(1 + \frac{E}{\sigma^2} \sum_{n=1}^{k_t} |h_n|^2 \right) \quad (3)$$

where h_n is unit gain channel between the n^{th} transmit antenna element and the receiver. The optimal method of obtaining a diversity order of k_t is through *water-filling*.

Finally, when the channel is perfectly known at both the transmitter (of size k_t) and the receiver (of size k_r), the capacity is given by

$$C = \log_2 \left(1 + \frac{E}{\sigma^2} \sum_{n=1}^{k_t} \sum_{m=1}^{k_r} |h_{mn}|^2 \right) \quad (4)$$

Each of the gains discussed above leads to increased throughput on an individual MIMO link. Array gain and diversity gain increase SNR on the link, which in turn leads to a higher achievable data rate. This is illustrated by Shannon's classic formula in Equation 2.1 and is achieved in practice by link-layer space-time coding techniques. Spatial multiplexing gain allows multiple data streams to be transmitted concurrently on a link, which can potentially multiply the data rate by the number of concurrent streams (see Equation 1). While optimization of performance on a single link has been extensively investigated, only a relatively few papers have yet considered how to make use of these various capabilities to optimize overall MIMO network performance.

2.2 Channel State Information and Interference Suppression

With a MIMO link, interference cancellation (IC) between concurrently transmitting links can be done by the transmitter or by the receiver or both. In the former case, we say that the transmitter nulls interference at receivers, while in the latter case, we say

that the receiver suppresses interference from transmitters. To completely eliminate interference requires channel state information (CSI). Receivers can measure channels during transmission of probe sequences in order to collect CSI necessary both for interference suppression and for performance optimization of the channel. In many cases, the receiver feeds back the channel state information (CSI) to the transmitter, and the transmitter and receiver jointly perform the appropriate transformation by using the singular value decomposition of the channel matrix. When the receiver feeds back CSI to the transmitter, this is referred to as closed-loop MIMO, and when only the receiver has CSI, this is referred to as open-loop MIMO. In general, closed-loop MIMO performs better but has an overhead associated with the exchange of CSI.

Suppose there are two concurrently transmitting links $l_1 = (t_1, r_1)$ and $l_2 = (t_2, r_2)$. Consider the interference generated by l_1 on l_2 . One approach is for the interference suppression to be done by the receiver r_2 . Suppose t_1 has k_{t_1} DOFs and its link carries s_1 streams and r_2 has k_{r_2} DOFs and its link carries s_2 streams. If $k_{r_2} - s_2 \geq s_1$, i.e. if the total number of streams being transmitted on the two links does not exceed the number of antenna elements at r_2 , then r_2 can separately decode all of the streams from both t_1 and t_2 , and simply discard the streams from t_1 . This effectively eliminates interference from l_1 . However, in order to do this, it is very important to note that r_2 must possess CSI for both link l_2 and the unused link $l_3 = (t_1, r_2)$. This implies that r_2 successfully received a training sequence from t_1 in order to perform a channel measurement on link l_3 . Thus, in general a receiver can cancel interference from transmitters for which it is inside the transmission range. This is called “receiver-side suppression”, where r_2 suppresses the transmission from t_1 without t_1 needing to modify the way it transmits to its intended receiver. Typically, it is assumed that there is a range, referred to as the interference range, which is greater than the transmission range up to which transmitting nodes can interfere. A common assumption is that the interference range is twice the transmission range. Thus, in

most cases, a receiver is only capable of canceling interference from some transmitters (the ones for which it is inside the transmission range) but not for others (the ones for which it is outside the transmission range but within the interference range).

Another approach is for the interference suppression to be done on the transmitter side. In the above example, it might be possible for t_1 to null its signals at r_2 , so that its transmission has no effect on that receiver. However, in order for t_1 to do this, it must know the CSI for the non-existent link $l_4 = (t_1, r_2)$. In general, t_1 can null its signals at r_2 if it has the appropriate CSI and if $k_{t1} - s_1 \geq s_2$, i.e. if the total number of streams being transmitted on the two links does not exceed the number of antenna elements at t_1 . This is called “transmitter-side nulling” and it allows r_2 to use all of its antenna elements for processing its desired signal without any consideration of the transmission coming from t_1 . Since there is no natural feed-back mechanism to get the CSI of l_4 to t_1 , even if r_2 is able to measure l_4 , the potential use of this type of interference cancellation is much more limited than receiver-side suppression.

An example of MIMO interference cancellation is done by the zero forcing (ZF) beam-former, which is a linear technique. The ZF beamformer [1] is given by the $k_r \times k_t$ matrix $C = (H^*H)^{-1}H^*$, where k_t is the number of antenna elements at transmitter, k_r is the number of antenna elements at receiver, H is the $k_r \times k_t$ channel matrix, and H^* is its conjugate transpose. Matrix C is used to either pre-process the transmit signal at the transmitter end, or to post-process the receive signal at the receiver end. The former is a case of the transmitter nulling interference it generates at receivers, while in the latter case, it is a case of the receiver suppressing interference from transmitters.

Consider the following three key benefits to using transmit and receive arrays in a communication link:

1. The ability to mitigate interference.
2. The ability to spatially multiplex several data streams onto the MIMO channel.

3. The ability to mitigate small scale fading (a.k.a. spatial diversity).

These benefits cannot be fully realized simultaneously through linear processing. An antenna array (either transmitter or receiver array) with linear processing that mitigates interference has a diminished capacity in the number of spatially multiplexed streams that it can decode and in its ability to combat fading, and vice-versa. For a transmit or receive array, this trade-off is summarized by the conservation theorem, which states (see pg.548, [1])

$$\text{diversity order} = M - N_S - N_I + 1 \geq 1$$

Here, M is the number of antenna elements in the array (either transmitter or receiver array), N_S is the number of spatially multiplexed streams supported by the MIMO link, and N_I is the number of interferers (interfering transmit antenna elements) that must be suppressed by the array if it is a receiver or the number of interfering receive antenna elements at which its signal must be nulled if it is a transmitter. Note that the number of antenna elements that must be used to handle interference is dependent on the number of streams being transmitted on the interfering (or interfered-with) link, rather than the number of streams being transmitted on the array's own MIMO link.

Since the diversity order should be at least one, we have

$$M \geq N_S + N_I$$

The size of an antenna array therefore must be at least equal to the sum of the number of data streams that it supports and the number of interfering streams that it mitigates. This corresponds to the so-called degrees of freedom (DOF) model [8, 52], wherein *antenna elements provide DOFs that can be divided arbitrarily between spatial multiplexing and interference cancellation*. Note that in the DOF model, the term “degree of freedom” is used to refer to an antenna element on either the transmitter

or receiver, which can be used to support multiplexing of a stream on the link or cancellation of interference with another link. This usage of the term is different from some other work, e.g. [25, 38], in which the number of DOFs refers to the number of independent streams that can be supported.

There is thus an evident symmetry between transmitters and receivers in terms of the usage of available degrees of freedom in supporting spatially multiplexed streams and in mitigating interference (assuming that perfect CSI of communication and interfering links is available at all transmitters and receivers.). This is seen by considering two interfering MIMO links each carrying s_1 and s_2 data streams respectively. Suppose that the transmitter of link 1, T_1 , nulls itself at the receiver of link 2, R_2 . In order to avoid putting any energy from T_1 at the interfering s_2 elements of R_2 , T_1 requires s_2 degrees of freedom to project its transmit vector into the space that is orthogonal to the space spanned by the MRC weights of each of the s_2 selected antennas at R_2 . The constraint on the size of T_1 is therefore $k_{t_1} \geq s_1 + s_2$. On the other hand, suppose that R_2 suppresses the s_1 data streams from T_1 . This would impose s_1 constraints on the receive vector. The constraint on the size of R_2 is therefore $k_{r_2} \geq s_1 + s_2$. Hence, irrespective of whether interference suppression is done by a transmitter or a receiver, the same constraint on the array size is felt. The DOF model is formalized in the following section.

2.3 MIMO Degrees of Freedom Model

The use of MIMO antenna elements is typically modeled with degrees of freedom (DOFs). The DOFs available on a link characterize the number of independent streams that can be spatially multiplexed on the link, the amount of interference that can be suppressed by the link, and a trade-off between these two capabilities. A node with k antenna elements has up to k DOFs, which it can use for multiplexing and/or interference suppression. In the absence of interference, a link with k_t DOFs

at the transmitter and k_r DOFs at the receiver can support up to $\min(k_t, k_r)$ spatially multiplexed streams. If DOFs are used for interference suppression, then the number of streams that can be supported on a link is reduced.

The *full degrees-of-freedom model*, used for example in [56, 49, 54], allows both transmitters and receivers to use antenna elements arbitrarily to perform spatial multiplexing and interference cancellation in a manner that accounts for the trade-off between these two functionalities. The trade-off rule requires that an antenna element of equivalently, a degree-of-freedom that is used for spatial multiplexing is no longer available to perform interference cancellation. Simply stated, MIMO resources can be divided according to the number of DOFs allocated to multiplex data streams and the number of DOFs used to perform interference cancellation. So, a transmitter t with k_t antenna elements can spatially multiplex s_0 streams on its link and create nulls at a set of interfering receivers, as long as $s_0 + \sum_{j \text{ is a receiver}} a_j s_j \leq k_t$, where s_j is the number of streams being received by the j^{th} receiver and $a_j = 1$ if the transmitter creates s_j nulls at that receiver and $a_j = 0$ otherwise. Similarly, a receiver r with k_r antenna elements can receive up to s_0 streams and cancel s_j streams coming from an interfering transmitter j , so long as $s_0 + \sum_{j \text{ is a transmitter}} a_j s_j \leq k_r$, where $a_j = 1$ if the receiver cancels interference from transmitter j and $a_j = 0$ otherwise.

In general, a node i with k DOFs, spatially multiplexes s_i streams on its link and suppresses interference with other nodes $j_1 \dots j_n$ carrying s_{j_l} streams respectively, if and only if the following inequality is satisfied.

$$s_i + \sum_{l=1}^n s_{j_l} \leq k$$

Note from the earlier discussion that the full degrees-of-freedom model assumes that *every transmitter and receiver has CSI about its own link and any links it can interfere with* (for a transmitter) *or be interfered by* (for a receiver). The *receiver*

degrees-of-freedom model, used for example in [9, 44], is similar to the full degrees-of-freedom model except that it assumes that only receivers can do interference cancellation. Thus, in the receiver degrees-of-freedom model only the receivers need to collect CSI, although they need to have CSI both for their own links and for every possible interfering link that could be used concurrently with their own. Finally, the DOF models implicitly assume a binary interference model, where one link either completely interferes with another (in which case interference suppression is necessary for the two links to be used concurrently) or there is zero interference between the links.

2.4 Network Model

A heterogeneous network is assumed where the antenna arrays of different nodes can be different sizes. A distinguishing characteristic of our work is that we account for variable stream rates, both on different links and on the same link. We model this with a rate function, which maps links to achievable rates based on their available MIMO resources. This will be discussed later in this section.

2.4.1 Interference Model

We consider multi-hop networks consisting of a set of links $L = \{l_1, l_2, \dots, l_m\}$, where each l_i is a transmitter-receiver pair (t_i, r_i) . Any wireless propagation model can be used to compute the set of links, given a set of node positions, or the set of links can simply be given as input. We assume that all links use the same wireless channel. The primary limitation on concurrent wireless transmissions in the same channel is interference.

Primary interference is a basic constraint on concurrency of transmissions, which

dictates that each node can participate in one transmission at a time, either as transmitter or as receiver². A set of links is said to be primary-interference-free if and only if every node in the network appears at most once in the set, that is to say, a node that is designated to be a transmitter for one link in a given time slot cannot also be designated to be the receiver on another link in the same time slot. Wireless interference models are distinguished by their assumptions on secondary interference, which specify how two different wireless links made up of four distinct nodes interfere with each other. For secondary interference, we adopt a simple binary interference model wherein two links either interfere completely or not at all. Thus, we define a directed *conflict graph* $CG = (L, E)$, where the vertex set L is the set of links to be scheduled, and directed edge $(l_i, l_j) \in E$ if transmitter t_i causes interference at receiver r_j . Again, we do not assume any specific underlying interference model, e.g. interference is not necessarily specified by a simple interference range nor do the conflicts have to be symmetric. We simply assume that the conflicts between links are known. The conflict graph can be computed from the set of links and the node positions by specifying an underlying (binary) interference model or the conflict graph can simply be given as input.

Admittedly, modeling interference as a binary phenomenon is a simplification of reality, and using more complex SINR-based interference models as is done in some non-MIMO wireless scheduling (e.g., [12]) would be preferable. Etkin and Tse introduced the concept of generalized degrees of freedom in [56], which accounts for relative signal strengths, unlike the conventional DOF model which assumes all signals to be of equal strength. Huang, Cadambe and Jafar employed the GDOF model in [29] to study the sum capacity of the symmetric X-channel under different interference levels. Karmakar and Varanasi extended this model to a MIMO setting in [35] where they characterize the capacity of a Gaussian interference MIMO channel. Again, while

²Here, we assume each node is equipped with only a single radio.

such an approach is tractable for centralized and symmetric networks, applying SINR considerations to arbitrary interference networks is very complex and remains an open problem. Novel PHY layer MIMO models would have to be developed to address this problem. For this reason, modeling interference as a binary phenomenon is a common assumption made by researchers in MIMO networking and communication ([9, 8]) and we adopt the same assumption in this thesis.

2.4.2 Rates and Streams

The data rate on an individual MIMO link is determined by the characteristics of the channel in between the transmitter and receiver, the numbers of DOFs used at the transmitter and the receiver for multiplexing, and the number of multiplexed streams. We model this with a rate function $R(t_i, r_i, ADOF_{t_i}, ADOF_{r_i}, s_i)$, which gives the rate on the link (t_i, r_i) when $ADOF_{t_i}$ DOFs are available at t_i DOFs and $ADOF_{r_i}$ are available at r_i and s_i independent streams are spatially multiplexed on the link. Note that this rate is computed in absence of interference, since in our approach interference is always completely removed by allocating MIMO resources and performing interference suppression. We do not make any assumption on the rate function, i.e. it can be an arbitrary function. Note that, if the channel between t_i and r_i is random (as is the case with Rayleigh fading channels, for example), the rate on the link is also a random variable. In this case, we interpret R as the expected data rate, which can also be thought of as the long-term rate on the link if its channel characteristics change dynamically and at random.

Note that the rate functions as specified above are approximations to the actual rates, which depend on the instantaneous channel characteristics and the MIMO weights that are chosen by the transmitter and receiver. While we believe those rates to be reasonable approximations, which capture the essence of the diversity-multiplexing-suppression trade-off, if more accurate techniques for approximating

data rates are developed, they could easily be plugged into our problem formulation.

Table 1 shows representative rates for one link used in our later evaluations. The amount of interference generated by a link l is dependent on the number of streams that are spatially multiplexed on l (every other link that wishes to suppress interference on or from l must allocate one DOF for every stream on l). Therefore, while allocating more streams on a link l increases the aggregate rate on that specific link, more radio resources (DOFs) need to be used in the network to cancel the interference generated by the transmitter of link l . However, while interference increases linearly with the number of streams, there is a law-of-diminishing-returns effect on data rate as the number of streams increases. Note in the table that going from 1 to 4 streams increases the link data rate by at most a factor of $\frac{119.25}{56.31} \approx 2.12$. Thus, even if a link has extra DOFs available, the overall network performance might benefit if the link uses those DOFs for array and diversity gains, rather than for increasing the number of streams it transmits. For example, a link with the characteristics in Table 1 that has 3 DOFs available at both transmitter and receiver could use those DOFs to transmit 2 streams, thereby achieving almost as high a rate as with 3 streams but generating less interference in the network. Clearly, it is critical to know the shape of the rate function when trying to optimize network performance.

Table 1: Link rates of a sample link (in Mb/sec) for different numbers of DOFs and streams ($k_i^t = k_i^r = 4$)

s	(t, r)			
	(1,1)	(2,2)	(3,3)	(4,4)
1	56.31	76.64	85.01	87.71
2	–	82.25	104.96	116.26
3	–	–	105.06	119.25
4	–	–	–	119.25

2.5 Summary of Modeling Assumptions

Assumptions used in this thesis are as follows:

1. Perfect CSI of communication and interfering links is available at all transmitters and receivers.
2. Transmitters and receivers are both capable of interference cancellation.
3. Interference cancellation is coordinated such that, for any link l_1 interfering with another link l_2 , either the transmitter of l_1 nulls its signal at receiver elements of l_2 or the receiver of l_2 suppresses the signal from the transmitter of l_1 , but not both.

CHAPTER III

RELATED WORK

3.1 MIMO Communication Theory

Using multiple antennas at both ends of wireless links (MIMO) is known as a unique solution to achieve a variety of performance gains. For power and bandwidth limited wireless systems, this opens up another dimension - space that can be exploited in a similar way as time and frequency. This allows both transmitters and receivers to use antenna elements arbitrarily to achieve an increase in the overall throughput by spatial multiplexing and/or interference cancellation, up to the limits imposed by the numbers of antenna elements they possess.

Interference management is one of the main challenges in wireless networks in which multiple transmissions occur concurrently over a common medium. Extensive experimental and theoretical research has been ongoing over the past decade on handling interference in MIMO networks. Researchers have been studying the MIMO Gaussian interference channels from an *information theoretic* perspective. The capacity region of the two user Gaussian interference channel with multiple antenna nodes is studied in [16][18]. The achievable rate region of multiple input single output (MISO) interference channel is studied in [37] where the authors treat it as strong interference and in [36] where the authors treat interference as noise. The capacity region of the single-input multiple-output (SIMO) is studied in [39].

While capacity characterizations have been found for centralized networks (Gaussian MIMO multiple access and broadcast networks [84, 90]), similar capacity characterizations for most distributed communication scenarios (e.g. interference networks)

remain long standing open problems. In the absence of precise capacity characterizations, researchers have pursued asymptotic and/or approximate capacity characterizations. A promising approach in this direction is the *degrees of freedom characterization* of wireless networks. The degrees of freedom (also referred to as multiplexing gain [2] or capacity prelog) of a network approximates the capacity of a network as

$$C(SNR) = d \log(SNR) + o(\log(SNR))$$

Here, d is the number of degrees of freedom of the network. The degrees of freedom (DoF) of a wireless interference network represents the number of interference-free signaling- dimensions in the network. $C(SNR)$ represents the capacity of a network as a function of the signal to noise ratio (SNR).

The following subsections detail related prior work done on MIMO interference handling, capacity characterizations of MIMO systems through information theoretic approaches and as well as through the degrees-of-freedom approach.

3.1.1 Interference handling

At a high level, the different interference management approaches used in practice and their information theoretic basis may be summarized as follows.

- **Decoding:** If the interference is strong, then the interfering signal can be decoded along with the desired signal. There is a price to be paid in doing this because, although decoding the interference improves the rate of the desired signal, ensuring the decodability of the interfering signals limits the other users' rates. Although it is theoretically somewhat supported by the capacity results on strong-interference-scenarios in the context of the two-user interference channel [37], the extension of these results to more than two users is not straightforward in general.
- **Treating interference as noise:** In this scheme, users adjust their transmission power and treat the interference produced by every other user as noise. From an

engineering stand point, it is natural and makes sense to treat *weak* interference as noise and simply apply single user encoding/decoding.

- Interference avoidance or interference cancellation (IC): Users coordinate their transmissions by orthogonalizing their signals in time or in frequency or in space.

In this thesis, all desired and interfering signals are assumed to be of comparable strength and interference is handled through interference avoidance. The primary issue of IC is balancing the need for high received signal power for each user against the interference produced by the signal at other receivers. Several different IC algorithms exist. While the basic idea behind these is the same, namely the use of channel state information (CSI) to predict and then counteract the interference produced at each node in the network, they achieve different performance objectives. Typical performance criteria include zero-interference transmission, minimum transmit power subject to a minimum signal-to-interference plus noise ratio at each receiver, or maximum throughput subject to a given transmit power constraint. Two commonly implemented linear processing techniques used as part of IC schemes are the

- Minimum mean squared error (MMSE) beam-former - MMSE does not null those interferers which are below the noise floor, it merely ignores them [43].
- Zero forcing (ZF) beam-former - ZF instead completely nulls all interferers irrespective of their strengths.

These techniques can be applied at the transmitter or receiver array, provided CSI is available.

For the greater part, existing literature assumes CSI to be available only at the receivers. Transmitters possess no CSI. In other cases, CSI is assumed at the receivers, with transmitters possessing CSI only of the specific communication link(s) they are associated with. This is a suitable model for cellular networks where fast channel

dynamics and node mobility make it impractical for channel information to be fed back at the desired rate. Our model however, deviates from this trend by assuming perfect CSI of all communication links as well as of all interfering links at every receiver and transmitter. This is a reasonable assumption, e.g., for the backbone of a wireless mesh network, where nodes are fixed and channel conditions do not change rapidly. Periodic measurement and sharing of channel states by receiver nodes and feedback to every transmitter node is thus a feasible system design. In this thesis, it is assumed that neither the transmitters nor the receivers cooperate in signaling. In other words, each transmitter is unaware of the data of the other transmitter. Similarly, each receiver is unaware of the signal received by the other receiver. The main objective of such non-cooperative signaling schemes is the design of the filters at the transmitter and receiver arrays so as to exploit the structure of the channel matrices to maximize the system's available DOFs. A generic method of achieving this is by interference alignment.

As defined in [33], “*Interference alignment refers to the consolidation of multiple interfering signals into a small subspace at each receiver so that the number of interference-free dimensions remaining for the desired signal can be maximized.*”

The potential for overlapping interference spaces was first pointed out by Maddah-Ali et al in [28]. They can be considered to be the pioneers of the spatial-interference-alignment technique. Their method was improved by Jafar in [30]. Within the class of signal vector space interference alignment schemes - alignment in frequency or space or time - alignment in the spatial dimension through multiple antennas (MIMO) is found to be more robust to practical limitations such as frequency offsets, compared to other types of alignment [34]. There are a number of schemes that realize interference alignment, each of which have certain merits and demerits but are all instances of the same conceptualization.

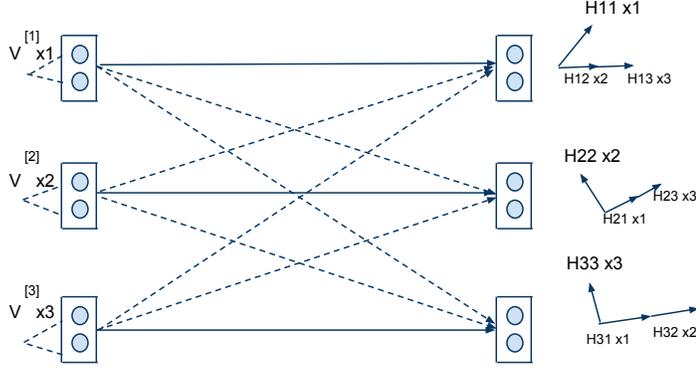


Figure 4: Interference Alignment

One such solution is to combine interference-alignment-precoding with zero forcing. This is illustrated in Figure 4 and is briefly described here. The description is borrowed from the the work of Yetis et al [33]. Consider a K -user MIMO system where the size of the transmitter array of the k^{th} user is $M^{[k]}$ and the size of its receiver array is $N^{[k]}$. The received signal at the k^{th} receiver is $Y^{[k]} = \sum_{l=1}^k H^{[kl]} X^{[l]} + Z^{[k]}$ where $Y^{[k]}$ and $Z^{[k]}$ are the $N^{[k]} \times 1$ received signal vector and the zero mean unit variance circularly symmetric additive white Gaussian noise vector (AWGN) at the k^{th} receiver respectively. $X^{[l]}$ is the $M^{[l]} \times 1$ signal vector transmitted from the l^{th} transmitter and $H^{[kl]}$ is the $N^{[k]} \times M^{[l]}$ matrix of channel coefficients between the l^{th} transmitter and and the k^{th} receiver. In interference alignment precoding, the transmitted signal from the k^{th} user is $X^{[k]} = V^{[k]} \tilde{X}^{[k]}$ where $\tilde{X}^{[k]}$ is a $d^{[k]} \times 1$ vector that denotes $d^{[k]}$ independently encoded streams transmitted from the k^{th} user. The $M^{[k]} \times d^{[k]}$ precoding (beamforming) filters $V^{[k]}$ are designed to maximize the overlap of interference signal subspaces at each receiver while ensuring that the desired signal vectors at each receiver are linearly independent of the interference subspace. Therefore, each receiver can zero-force all the interference signals without zero-forcing any of the desired signals. The zero-forcing filters at the receiver are denoted by $U^{[k]}$.

The transmit and receive filters are designed such that the following conditions are simultaneously satisfied.

$$U^{[k]\dagger} H^{[kj]} V^{[j]} = 0 \quad j \neq k \text{ and}$$

$$\text{rank}(U^{[k]\dagger} H^{[kk]} V^{[j]}) = d^{[k]} \quad \forall k \in \{1 \dots K\}$$

Another solution is to perform beamforming at the transmitter of the k^{th} user, ($\forall k = \{1 \dots K\}$) such that a null is steered towards a subset $\{R_k\}$ of the interfering receivers. The remaining receivers which still experience interference from the k^{th} transmitter perform MMSE or zero-forcing or successive-interference-cancellation to suppress this interference, and thereby decode the desired signals. This is illustrated in Figure 5. Both interference alignment and beamforming-suppression achieve the same performance goals in theory and differ in their implementation.

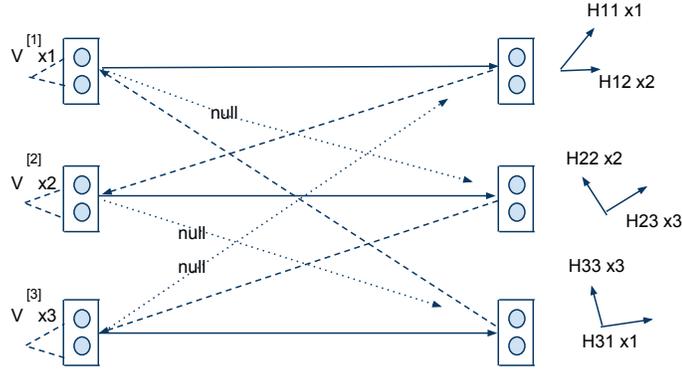


Figure 5: Beamforming and Suppression

Yet another approach that is being investigated currently in our lab is to perform a network-wide optimization by an iterative MMSE-based algorithm which produces weight assignments at all transmitters and receivers such that the ratio of the received (desired) signal strength to the interference is maximized at every receiver. The weight

vector of every transmitter (receiver) is dependent on the weight vectors of all other transmitters and receivers [94].

3.1.2 Information theoretic analysis and upper bounds

There is a rich collection of work on the information theoretic capacity characterization of Gaussian interference channels [37, 55, 16, 17, 18, 22, 47, 74]. These efforts have produced an extensive set of interesting results that successfully address various aspects of the problem. The capacity region of the two user Gaussian interference channel with MIMO capability is studied by Jafar and Chen in [16][18]. A special case of the Han-Kobayashi scheme [55] is shown by Etkin and Tse in [56] to achieve the capacity of the two-user interference channel within one bit. Furthermore, Etkin and Tse [56] provide a generalized degrees of freedom (GDOF) characterization that identifies different operational regimes for the two-user interference channel.

While a vast body of literature has been devoted to the investigation of MIMO channel capacity in typical, centralized configurations such as one-to-one, one-to-all, all-to-one, etc., the issue of characterizing capacity of MIMO-equipped networks has been approached only recently. We on the other hand, are interested in network-wide optimization. The difficulty in achieving such characterizations is that usage of MIMO links introduces additional optimization parameters into an already very complex optimization problem involving routing, transmit power control, and scheduling in the most general formulation.

Given the complexity of this problem, researchers have been pursuing approximate capacity characterizations by adopting the degrees-of-freedom approach. The following section addresses the study of MIMO Gaussian interference channels from a degrees of freedom perspective.

3.1.3 Degrees-of-Freedom (DOF) approach

The degrees of freedom (DoF) of wireless interference networks represent the number of interference-free signaling- dimensions in the network.

The degrees of freedom of the two user MIMO gaussian interference channel with M_1, M_2 antennas at the transmitters and N_1, N_2 antennas at their respective receivers was shown to be $\min\{M_1 + M_2, N_1 + N_2, \max(M_1, N_2), \max(M_2, N_1)\}$ by Jafar and Fakhreddin in [23]. For optimal wireless *network* design, the natural question is whether the insights from the two-user interference channel generalize to interference channel scenarios with *more* than two users. Unfortunately, for more than two users, even degrees of freedom characterizations are not known. In [26], Cadambe and Jafar show that the sum capacity for the K user interference channel with $M \geq 1$ antennas at each node scales linearly with the number of users when the technique of interference alignment is used. They characterize the capacity as $C(SNR) = \frac{KM}{2} \log(SNR) + o(\log(SNR))$. Thus, this channel has $\frac{KM}{2}$ degrees of freedom i.e. it loses half its DOFs due to performing interference cancellation. This result was generalized by Gou et. al. in [24] to a K user MIMO Gaussian interference channel with M antennas at each transmitter and N antennas at each receiver. They assume that the ratio $\frac{\max(M,N)}{\min(M,N)} = R$ is equal to an integer which includes MISO and SIMO interference channels as special cases. They provide both the innerbound (achievability) and outerbound (converse) of the total number of degrees of freedom for this channel and show that the total number of degrees of freedom is equal to $\min(M, N)K$ if $K \leq R$ and $\min(M, N)\frac{R}{R+1}K$ if $K > R$. Their achievable scheme is based on interference alignment. Since each user can achieve $\min(M, N)$ degrees of freedom without interference, their result implies that there is no loss of DOFs when $K \leq R$ and only a fraction $\frac{1}{R+1}$ of DOFs are lost due to interference cancellation when $K > R$. However, these results require full cooperation between all transmitters and full cooperation between all receivers. In fact, the authors state that for this K user

system, when the number of interferers in the system is above a threshold, which depends on the number of transmit and receive antennas, cooperation provides no benefit from the DOF point of view for almost all channel realizations. In this thesis (and all our related publications) only the *non-cooperative MIMO* case is considered.

3.2 MIMO network wide throughput optimization

Improving the capacity of ad-hoc wireless networks to approach the Shannon limit is one of the most important problems of network information theory. Although centralized networks such as Gaussian MIMO multiple access (many-to-one) and broadcast networks (one-to-many) lend themselves to tractable capacity analysis, most arbitrary, distributed communication scenarios such as MIMO interference networks largely remain as open problems. For these reasons, researchers have typically introduced some simplification in the models and/or problem formulation in an attempt to enable them to characterize the degrees of freedom of various distributed wireless networks.

Approaches to this problem heretofore include *bounding* network throughput; characterizing the benefits of cross-layer optimizations in interference-limited MIMO-equipped wireless mesh networks; and characterizing optimal throughput performance for highly restricted network scenarios.

The problem that this thesis is centered on is that of MIMO network-wide throughput optimization. We do not consider cooperative MIMO ([89]) using antenna arrays distributed across multiple nodes (Cooperative MIMO requires tight synchronization [87] between transmissions on different nodes, which provides extreme challenges for high data rate communications.). We consider the use of co-ordinated interference suppression (either the transmitter nulls itself at an interfering receiver or the receiver suppresses interference from that transmitter, but not both) which involves multiple links MIMO but does not require synchronization of communications [15].

3.2.1 Stream Allocation/Scheduling

The work [8] of Hamadou, Shin and Jiang is closely related to ours. Here, the authors study the throughput of a multi-hop wireless MIMO network. Similarly to our work, they assume that perfect CSI is available at both receivers and transmitters. However, although they include constraints that account for both transmitter-side and receiver-side suppression, these are based on *necessary but not sufficient feasibility conditions*. They cast the problem of optimal throughput determination as an integer linear programming (ILP) formulation, whose solution is upper bounded by the corresponding linear programming (LP) formulation. This upper bound is used to numerically characterize throughput performance under different MIMO usages (spatial reuse, spatial multiplexing, and their combination), subject to fairness constraints on allocation of scarce wireless capacity among mobile clients. In section 4.3 we show that our analytical results qualitatively confirm the main findings of the numerical evaluation reported in [8] in terms of the relative throughput gains when spatial reuse is combined with spatial multiplexing for the case of a single collision domain. Differently from the results reported here, our experimental results are exact rather than upper bounds. Also, a considerable part of our work on the subject is very analytical and we were even able to provide a simple closed-form expression for the optimal throughput in the restricted case of a single collision domain. Hamdaoui, Shin, and Maiya extend the problem formulation to include multiple frequency bands [40].

Perhaps the most closely related work to ours is the important work of Liu, Shi, and Hou [41]. To our knowledge, their paper presented the first complete solution to a MIMO network optimization problem. By complete solution, we mean that the result of the optimization corresponds to a set of zero forcing equations for every node that is guaranteed to be solvable and suppresses all interference between links. Their approach, where links are successively scheduled such that each link is responsible for suppressing interference with all links scheduled before it. This procedure results

in a feasible link schedule that ensures all interference can be removed. However in doing so, the approach precludes some feasible link schedules and is, therefore, sub-optimal, in general.² We instead model interference suppression based on the degrees of freedom (DOF) model. The 3-link network in Figure 6 with every link carrying two streams is an example of a situation where the DOF model provides a solution ensuring that all interference is suppressed, while the OBIC approach of [41] will not find a solution. In the solution shown, the dashed lines represent interference suppression, every node uses two DOFs for suppression, and this is the only way to suppress all interference. Since, in the approach of [41], the first node chosen will never use any DOFs for suppression, that approach will not find a solution for this case. Whether a zero forcing solution always exists for any interference suppression assignment given by the DOF model is an open problem. However, we have iteratively solved the zero forcing equations of the Figure 6 example for large numbers of random link combinations and never failed to find a solution.

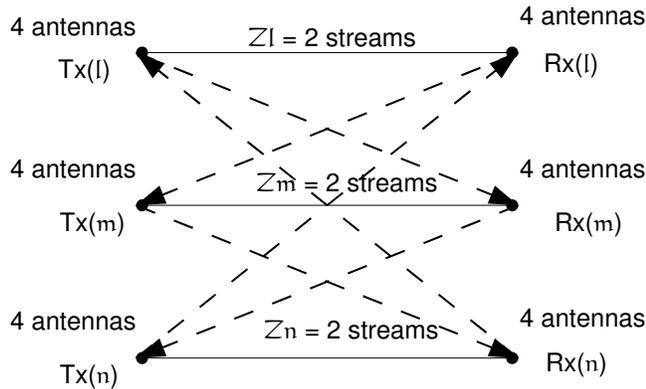


Figure 6: Three link example where every node must use some DOFs for interference suppression.

A few other papers have attempted at characterizing the optimal throughput

achieved in quite restricted network scenarios [64, 82]. For instance in [82], the author considers a single-hop network and addresses the maximum throughput characterization by casting it as a joint scheduling and MIMO stream allocation problem, and characterizing the optimal solution for the case of two interfering links.

Another important difference between our work and most prior work is the following. Prior work in this area makes the simplifying assumption that link rates increase linearly with the number of streams on a link¹. In contrast to this, we use rate functions on a link, which specifically consider the dependence of channel capacity on the numbers of DOFs used for transmitting and receiving. This allows array and diversity gains to be factored into MIMO resource allocation decisions. It models the “law of diminishing returns” of spatial multiplexing, wherein performance increases become smaller and smaller as the number of spatially multiplexed streams is increased on a given link. Therefore, our variable rate model accounts for sub-linear increase of aggregate rate with the number of streams on a link. This is necessary to fully exploit the diversity-multiplexing-suppression trade-off.

A few papers have considered non-linear aggregate rates when attempting to characterize throughput of MIMO-equipped networks, by including so-called variable rate stream control in the problem formulation. Variable-rate stream control for CSMA-based MAC layer has been discussed in [44, 88]. Our results, and other optimization-based approaches, target TDMA-based MACs. The approaches of [47] and [48] consider variable rate stream control but only provide upper bounds complemented with feasible heuristic approaches [47], or simply heuristic solutions [48]. Thus, to the best of our knowledge, our work in this thesis and in [49] and [50], represented the first attempt to solve a multi-hop MIMO network throughput optimization problem in terms

¹Traditional DOF models do not account for data rates that vary with SINR, i.e. the data rate is assumed to be fixed and there is a single SINR threshold where, above this threshold, the fixed data rate is achieved on the link and, below this threshold, no communication is possible; this means that the beneficial impacts of array gain and diversity gain are ignored by these models.

of the diversity-multiplexing-interference suppression trade-off while accounting for a variable rate stream control model.

3.2.2 Cross Layer Optimization of MIMO WMNs, MIMO-aware Routing and MAC-based Approaches

The vast amount of work on integrating MIMO with the networking stack has focused on the MAC layer. Some of this work, e.g. [44, 62], has considered MAC layer techniques to allocate streams to links that optimize performance in a local sense. [44] does not consider network-wide resource allocation to optimize performance.

An interesting work in this area is [9] (see also [10]) where the authors characterize the benefits of cross-layer optimizations in interference limited wireless mesh networks with MIMO links. They formulate a framework where data routing at the protocol layer, link scheduling at the MAC layer and stream control at the physical layer can be jointly optimized for throughput maximization in the presence of interference, and then develop an efficient algorithm to solve the resulting throughput optimization problem subject to fairness constraints. This approach provides only a bound to the achievable network throughput.

In [7], Chu and Wang address some of the same problems that we do. Their work aims to improve the network wide throughput and transmission quality in MIMO-based ad hoc networks by proposing an integrated scheduling by jointly considering traffic demands, service requirements, network load, multiuser diversity, and channel conditions. While this work accounts for variable rate streams, it models interference-suppression only at the receiver side, i.e. transmitters do not perform any interference suppression. Moreover, exact optimal solutions are not evaluated under this framework. Work on MIMO broadcast includes [75], [76], where multiuser interference is canceled at the transmitter by Dirty Paper Coding, which is of theoretical importance but is considered impractical due to high complexity. Prior work on interference cancellation (IC) of multiuser MIMO systems has mainly focused on the uplink [80], [81].

However, because of the need for inexpensive mobile units with low complexity of realization, closed loop MIMO systems have been studied where CSI is known at the transmitter of the base station. In related work, multi-user precoder designs [83], [19] can serve multiple mobile units over the same frequency in such a way that co-channel interference is mitigated.

Although a number of papers present formal network optimization problem formulations [[8], [9], [48], [53], [52]], they introduce some simplification in the models and/or problem formulation. Sundaresan, et al., give a probabilistic throughput formulation but do not explicitly consider interference constraints [52]. Chu and Wang [48] include constraints that are sufficient but not necessary for feasibility and they assume receiver-side suppression only. The recent work on MIMO-aware routing by Chu and Wang [51] mathematically formulates a multi-commodity flow problem, but they only consider receiver-side suppression and do not provide details of how the stream rate function is modeled. Liu, Hou, and Sherali jointly consider the problems of power control and routing with a maximum throughput objective through a network flow formulation [53]. Interference constraints are not considered. Mumey, Tang, and Hahn give an approximation algorithm for joint stream control and scheduling, where only receiver-side interference suppression is considered.

3.3 Feasibility

The problems of stream allocation and link scheduling for throughput optimization are closely tied to the problem of solving feasibility. One of the major findings of our research was that in a MIMO setting, with the exception of certain special cases, feasibility checking is a very expensive operation. However, many existing iterative scheduling algorithms assume that feasibility can be determined efficiently ([10], [11], [12], [13]). In these cases, the scalability with increasing network size would probably be quite limited.

Therefore, feasibility is a fundamentally important problem to be researched. Very little work has been done on this subject. The biggest contributions of this thesis are in providing analytical and experimental results towards better understanding feasibility. In fact, it was only after publishing our first three papers on feasibility ([49],[50],[54]), that work done on this subject by other researchers began to appear. The most substantive of these is [33] by Yetis, Gou, Jafar and Kayran. This work considers the problem of checking feasibility of interference alignment in signal vector space, based only on beamforming, for the K -user MIMO interference channel. They relate the feasibility issue to the problem of determining the solvability of a multivariate polynomial system. They show that a *necessary* condition for an assignment to be feasible is that the system of equations describing the assignment is *proper* (i.e. the number of equations does not exceed the number of variables). That is to say, if the system is proper, then it is *likely* to be feasible. They find a lower bound for the throughput (number of streams) in a K -user single collision domain (the authors of [33] refer to this as the K -user interference channel). They *conjecture* that this lower bound is also an upper bound and therefore they conjecture it to be optimal (see pg 4,5 of [33]). In our first work [49], we had considered the same problem and had derived the optimal value of throughput for a K -user single collision domain with uniform antenna array sizes, and this matches with the result conjectured to be true in [33]. They further extend their analysis to the case where every link in the single collision domain has receiver array size equal to M and transmitter array size equal to N (this is a slight loosening of the uniform array size requirement).

CHAPTER IV

PROBLEM DEFINITION

The problem at large, is to determine which links of a MIMO network, when scheduled together concurrently, will maximize the aggregate throughput. This is a version of network scheduling in a MIMO setting where the goal is to maximize the throughput delivered in a single scheduling slot. We identify three inter-related problems discussed below.

4.1 Feasibility

The problem of feasibility is to determine if a set of transmissions can be undertaken concurrently such that all individual transmissions are successful, under a given interference model. Many existing scheduling algorithms assume that feasibility can be determined efficiently[[10], [11], [12], [13]]. These algorithms iteratively add transmissions to slots in a frame, checking a slot for feasibility whenever a new transmission is proposed to be added to a slot by the algorithm. Thus, feasibility is considered to be a very simple operation that can be repeated many times during execution of the scheduling algorithm. One of the interesting findings of this research is that with MIMO links, in certain cases, feasibility checking becomes a very expensive operation. In these cases, the scalability with increasing network size could be quite limited for this iterative schedule construction with repeated feasibility checking approach. Thus, alternative scheduling approaches might have to be developed for the most general MIMO link scheduling problems.

Without MIMO links, checking feasibility of a set of transmissions amounts to simply checking the transmissions against the given interference model. However, MIMO links have the capability to suppress interference. Thus, whether a set of

transmissions is feasible depends both on the interference model and on how the links choose to use their degrees of freedom (DOFs) in suppressing interference. As detailed earlier, the number of DOFs necessary for the transmitter of a link i to suppress interference on the receiver of a link j is given by the number of streams carried on link j . (The same is true if the receiver of i is suppressing interference from the transmitter of j .) Thus, to determine feasibility, we must know both which links are transmitting *and* how many streams are carried on each of the links.

Let the set of links under consideration be denoted by $L = \{(t_1, r_1), \dots, (t_m, r_m)\}$. Let $S = [s_1, \dots, s_m]$ be the stream allocation vector, i.e. the number of streams carried by (t_i, r_i) is $s_i > 0$. Let $K^t = [k_1^t, \dots, k_m^t]$ be the vector of (effective) DOFs of the transmitters, Similarly, let $K^r = [k_1^r, \dots, k_m^r]$ be the vector of (effective) DOFs of the receivers. Let $a_{ij}^t = 1$ if the transmitter of link i suppresses interference on the receiver of link j and let $a_{ij}^t = 0$, otherwise. Furthermore, let $a_{ii}^t = 1$, for all i . Denote by A^t the matrix of a_{ij}^t values. Similarly, let $a_{ij}^r = 1$ if the receiver of link i suppresses interference from the transmitter of link j , let $a_{ij}^r = 0$, otherwise, and let $a_{ii}^r = 1$, for all i . Denote by A^r the matrix of a_{ij}^r values.

The feasibility problem is defined as follows:

Input: A set $L = \{(t_1, r_1), \dots, (t_m, r_m)\}$ of links, a stream allocation vector S for L , and a conflict graph $G_c = (L, E_c)$.

Output: **True** if S and L are feasible and **False** otherwise. S and L are defined to be feasible if L is free of primary interference and there exist A^t and A^r such that:

1. $A^t S \leq K^t$,
2. $A^r S \leq K^r$, and
3. for all $i \neq j$ such that $(l_i, l_j) \in E_c$, $a_{ij}^t + a_{ji}^r \geq 1$.

Conditions 1 and 2 ensure that a node does not use more DOFs than it has

available. For each t_i , we have that:

$$s_i + \sum_{j:j \neq i \text{ and } (l_i, l_j) \in E_c} a_{ij}^t s_j \leq k_i^t$$

In other words, t_i uses s_i DOFs for spatially multiplexing its streams and uses s_j DOFs for every receiver on which it suppresses its interference, and the total of these values cannot exceed the size of t_i 's antenna array. Condition 1 states this inequality in matrix form over all transmitters and Condition 2 is the equivalent for receivers. Condition 3 ensures that all interference is cancelled, i.e. for every pair of links i and j where i interferes with j , either the transmitter of i or the receiver of j (or both) suppresses the interference from i to j .

Note that the cases of receiver-side suppression only and transmitter-side suppression only can easily be handled by this general problem statement. For receiver-side suppression only, a_{ij}^t is set to zero for all $i \neq j$, and for transmitter-side suppression only, a_{ij}^r is set to zero for all $i \neq j$.

Note also that Conditions 1 and 2 above are a variation of the basic DOF inequality in which a set of boolean variables, the a_{ij}^t 's and the a_{ij}^r 's, are included. These boolean variables indicate which nodes are suppressing interference on which other nodes. Thus, one way of stating the feasibility problem is to ask the question: "Does there exist an assignment of interference suppressions to nodes (a_{ij}^t and a_{ij}^r values) that satisfy the DOF inequalities at every node and together suppress all interference?". This formulation makes it clear that the feasibility problem is a special type of Boolean satisfiability problem.

4.2 Stream Allocation

The achievable rate $R(t_i, r_i, \text{ADOF}_{t_i}, \text{ADOF}_{r_i})$ on a link depends on the numbers of available degrees of freedom (DOFs) at both ends of the link. The number of available DOFs at t_i is given by $k_i^t - ks_{t_i}$, where ks_{t_i} denotes the number of DOFs that t_i uses to suppress its streams on receivers other than r_i . Note that $ks_{t_i} = \sum_{i \neq j} a_{ij}^t s_j$, i.e. ks_{t_i}

is the sum of the numbers of streams received by all receivers on which t_i suppresses its interference. Similarly, the number of available DOFs at r_i is given by $k_i^r - ks_{r_i}$, where $ks_{r_i} = \sum_{i \neq j} a_{ij}^r s_j$.

The stream allocation problem, formally defined below, is to find an optimal stream vector, given a set of links that could be scheduled concurrently. An optimal stream vector is defined as a feasible stream vector with maximum aggregate transmission rate. Links that could be scheduled concurrently means that the links are free of primary interference.

Input: A set $L = \{(t_1, r_1), \dots, (t_m, r_m)\}$ of primary-interference-free links, DOF vectors K^t and K^r , and rate function $R(t_i, r_i, \text{ADOF}_{t_i}, \text{ADOF}_{r_i})$.

Output: A stream allocation vector S and matrices A^t and A^r that make S feasible, where (S, A^t, A^r) has maximum aggregate rate over all feasible stream vectors.

4.3 One-Shot Link/Stream Scheduling

In the stream allocation problem, a set of primary-interference-free links is given. However, in classical one-shot link scheduling, the problem is to determine which links, when scheduled together concurrently, will maximize the aggregate rate. In other words, this is a version of link scheduling in which the goal is to squeeze as much out of a single scheduling slot as possible. Repeatedly scheduling a maximum-rate set of links over and over will yield a maximum throughput solution. However, such a schedule obviously does not meet any fairness criteria and, therefore, this approach cannot be considered a solution to an overall network scheduling problem. Nevertheless, one-shot scheduling algorithms can be adapted in various ways to address fairness and can therefore still form a core component of an overall scheduling approach.

We can generalize the stream allocation problem into a one-shot scheduling problem. In this case, we simply need to start with an arbitrary set of links (rather than being given a primary-interference-free set of links) while maintaining the same goal

of maximizing the aggregate rate. Thus, in this situation, the problem can be defined as follows:

Input: An arbitrary set $L = \{(t_1, r_1), \dots, (t_m, r_m)\}$ of links, DOF vectors K^t and K^r , and rate function $R(t_i, r_i, \text{ADOF}_{t_i}, \text{ADOF}_{r_i})$.

Output: A set of primary-interference-free links L_{pif} , a stream assignment vector S for L_{pif} , and matrices A^t and A^r that make S feasible, where (L_{pif}, S, A^t, A^r) has maximum aggregate rate over all sets of primary-interference-free links and feasible stream vectors.

The three problems we consider herein can be summarized as follows. In feasibility, a set of primary-interference-free links and a stream assignment vector are given and the problem is to determine if there is a DOF assignment that will allow the links to successfully receive the given numbers of streams when they are transmitted concurrently. In stream allocation, a set of primary-interference-free links is given and the problem is to determine a stream assignment vector (and DOF assignments) that maximizes the achievable aggregate rate on the links. Finally, for one-shot link/stream scheduling, an arbitrary set of links is given and the problem is to determine a set of primary-interference-free links and a stream assignment vector that maximize the achievable aggregate rate. In the following sections, we will consider these problems individually and we will build solutions to these problems on top of one another to ultimately produce a solution to the one-shot scheduling problem.

CHAPTER V

FEASIBILITY: THEORETICAL RESULTS

As discussed in Section 4.1, the problem of verifying feasibility over a MIMO network is a special type of Boolean satisfiability problem and in general, it is highly complex. Link scheduling and optimal stream allocation are closely related to checking feasibility, thus making it an fundamentally important problem. Analytical characterizations of feasibility over different MIMO network scenarios is the subject of this chapter. In section 5.2, results on a single collision domain (SCD) are derived. All links that are concurrently active in a single collision domain interfere with one another i.e. for any two active links l_i and l_j , the transmitter of l_i interferes with the receiver of l_j and the transmitter of l_j interferes with the receiver of l_i . In section 5.3, arbitrary multi-hop MIMO networks are considered. The following analyses are done in this chapter: (1) An analytical expression for the maximum number of streams that can be scheduled in a single collision domain in one time-slot is derived. This is done by evaluating the the optimal stream allocation vector. (2) The complexity of the problem of verifying feasibility is studied for a number of specific Multi-hop MIMO network scenarios. In Chapter 6, the results and insights obtained from these theoretical analyses are used to develop efficient heuristics for checking feasibility.

Before developing the system model, the notation introduced in Section 4.1 is

recapitulated here and some additional quantities are defined.

$K^t = [k_1^t, \dots, k_l^t]$ is the vector of (effective) DOFs of the transmitters

$K^r = [k_1^r, \dots, k_l^r]$ is the vector of (effective) DOFs of the receivers

$l =$ number of active links

$\mathbf{s} = [s_1 \dots s_l]$ is the $l \times 1$ stream allocation vector containing the number of data streams carried by each link

$w(A) =$ weight of matrix or vector A (sum of all entries)

$\mathbf{s}_m =$ optimal \mathbf{s} vector (having maximum weight)

$W_T =$ total amount of work (number of links on which IC is performed) done by all transmitters

$W_R =$ total amount of work done by all receivers

5.1 Matrix Formulation For MIMO Feasibility Checking

Interference suppression is modeled by the full degrees-of-freedom model, which was described in Section 2.3. The full DOF model allows both transmitters and receivers to handle interference suppression. A transmitter (or receiver) node i with k DOFs can spatially multiplex s_i streams on its link and null (or suppress) interference at the receivers (or transmitters) of a set links denoted by L_i if and only if $s_i + \sum_{j \in L_i} s_j \leq k$.

Consider a multi-hop MIMO network with a set L of l links and represented by a conflict graph is $G_c = (L, E_c)$. For any two links $l_i, l_j \in L$, $G_c(i, j) = 1$ if the transmission on l_i interferes with the receiver of link l_j . Similarly, $G_c(j, i) = 1$ if the transmission on l_j interferes with the receiver of link l_i . For a stream allocation vector \mathbf{s} to be feasible over L , interference between every pair of links must be removed. Therefore for the transmitter side, we have

$$s_i + \sum_{j \in L_i} s_j \leq k_i^t \quad \forall i \in L \quad (5)$$

where L_i is the set of links at which the transmitter of link i nulls itself. Rewrite this as

$$A_1 \mathbf{s} \leq K^t \quad (6)$$

Similarly, for the receiver side, we have

$$s_i + \sum_{j \in M_i} s_j \leq k_i^t \quad \forall i \in L \quad (7)$$

where M_i is the set of links whose transmissions the receiver of link i suppresses. Rewrite this as

$$A_2 \mathbf{s} \leq K^r \quad (8)$$

The transmitter-side matrix A_1 and the receiver-side matrix A_2 are binary, square matrices such that $A_1(i, j) = 1$ if the transmitter of link i nulls itself at the receiver of link j , and similarly, $A_2(i, j) = 1$ if the receiver of link i suppresses the transmission from on link j . A_1 and A_2 , however, are related. Any choice of A_1 completely determines A_2 and vice-versa. The relation is

$$\begin{cases} A_2(j, i) = 1 - A_1(i, j) \text{ if } G_c(i, j) = 1, \forall i \neq j \\ A_1(i, j) = 0; A_2(j, i) = 0 \text{ if } G_c(i, j) = 0, \forall i \neq j \\ A_1(i, i) = 1; A_2(i, i) = 1 \quad \forall i \end{cases} \quad (9)$$

Equation 9 follows from the fact that if transmitter i nulls itself at receiver j , then receiver j need not suppress the signal from transmitter i (coordinated interference cancellation). We therefore have

$$A_1 = G_c + I - A_2^T \quad (10)$$

Therefore, for the stream allocation vector \mathbf{s} to be feasible, the following three Conditions must hold.

$$\begin{cases} A_1 \mathbf{s} \leq K^t \\ A_2 \mathbf{s} \leq K^r \\ A_1 = G_c + 1 - A_2^T \end{cases} \quad (11)$$

5.2 Results on Single Collision Domain

Here, we consider the case when all links are in the same collision domain. By the same collision domain, we mean that any two links that are being used simultaneously will each cause the other's transmission to fail unless interference between them is canceled. We make some assumptions in our analysis of the single collision domain. These are

1. Primary interference has been eliminated, i.e. the set of links with data to transmit is free of primary interference.
2. All links have identical transmit and receive arrays, in terms of their sizes and signal processing capabilities i.e. $K_i^t = K_i^r = k \forall i \in L$
3. All links have the same rate. Moreover, different streams on any given link have equal rates. This is the *uniform rate model*.

The assumption of the uniform rate model implies that maximizing the throughput in the SCD is equivalent to maximizing the total number of streams that are concurrently scheduled.

Details of the following analyses are reported in this section:

1. When both spatial reuse and spatial multiplexing are performed together, we obtain an analytical expression for the optimal total number of streams as a function of the number of simultaneously active links.
2. Moreover, at this optimal point, we determine how the work of interference cancellation is distributed among all the transmitters and receivers.
3. The result in item 1 allows us to compute the maximum size of an SCD for a given value of k . This is the case of *spatial reuse only* (SRO), i.e. each link carries only one data stream and no multiplexing is done.

4. The result in item 1 also trivially shows that in the case that the size of the SCD equals one, i.e. when only one link is active, the optimal number of streams equals k . This is the case of *spatial multiplexing only* (SMO), i.e. no spatial reuse is done.

The next three sub-sections are devoted to deriving the optimal total number of streams in a SCD with l links (for odd values of l). When it is possible to use both spatial reuse and spatial multiplexing with MIMO links (SR+SM), the optimal way to use the MIMO DOFs is not obvious. The transmitters and receivers of a given link could use their DOFs to multiplex several streams, thereby increasing the link data rate of that link, or they could use their DOFs to cancel interference, thereby allowing more links to be simultaneously active. We will show that the optimal total number of streams equals $\frac{2kl}{l+1}$, and this is achieved when the work of interference cancellation is equally distributed among all transmitters and receivers.

5.2.1 Spatial Multiplexing with IC: A Matrix Formulation

Interference cancellation between $\frac{l(l-1)}{2}$ pairs of links ($l(l-1)$ cancellations total) must be done. Thus,

$$W_T + W_R = l(l-1)$$

For the transmitter side, we have

$$s_i + \sum_{j \in L_i} s_j \leq k$$

where L_i is the set of links at which the transmitter of link i nulls itself. Rewrite this as

$$A_1 \mathbf{s} \leq \mathbf{k} \text{ where } l \leq w(A_1) \leq l^2$$

Similarly, for the receiver side, we have

$$s_i + \sum_{j \in M_i} s_j \leq k$$

M_i is the set of links whose transmissions the receiver of link i suppresses. Rewrite this as

$$A_2 \mathbf{s} \leq \mathbf{k} \text{ where } l \leq w(A_2) \leq l^2$$

Note that $w(A_1) = W_T + l$ and $w(A_2) = W_R + l$. The solution \mathbf{s} must satisfy

$$A_1 \mathbf{s} \leq \mathbf{k} \text{ and } A_2 \mathbf{s} \leq \mathbf{k}$$

The optimal solution is the vector with maximum weight subject to the constraints $A_1 \mathbf{s}_m \leq \mathbf{k}$ and $A_2 \mathbf{s}_m \leq \mathbf{k}$, i.e.

$$\mathbf{s}_m = \max_{A_1, A_2} \{w(\mathbf{s}_m) \rightarrow \mathbf{s}_m : A_1 \mathbf{s}_m \leq \mathbf{k} \text{ and } A_2 \mathbf{s}_m \leq \mathbf{k}\}$$

A_1 and A_2 , however, are related on account of coordinated IC by

$$A_2 = I + 1 - A_1^T$$

The relation between the weights of A_1 and A_2 naturally follows from this as

$$w(A_1) + w(A_2) = 2l + l(l - 1) = l(l + 1) \quad (12)$$

The optimal solution can now be simplified as

$$\mathbf{s}_m = \max_{A_1} \{w(\mathbf{s}_m) \rightarrow \mathbf{s}_m : A_1 \mathbf{s}_m \leq \mathbf{k}, (I + 1 - A_1^T) \mathbf{s}_m \leq \mathbf{k}\}$$

Now, define the mapping f from two vectors \mathbf{s}_1 and \mathbf{s}_2 to choose the vector with minimum weight as

$$f : \{\mathbf{s}_1, \mathbf{s}_2\} \rightarrow 1\{w(\mathbf{s}_2) \geq w(\mathbf{s}_1)\} \mathbf{s}_1 + 1\{w(\mathbf{s}_1) > w(\mathbf{s}_2)\} \mathbf{s}_2$$

where the function $1(C) = 1$ if C is true and 0 otherwise. The optimal solution amounts to maximizing the minimum-weight vector of the two vectors $\mathbf{s}_1, \mathbf{s}_2$ which satisfy $A_1 \mathbf{s}_1 = \mathbf{k}$ and $A_2 \mathbf{s}_2 = \mathbf{k}$. This is expressed as

$$\mathbf{s}_m = \max_{A_1} \left\{ w(f(\mathbf{s}_1, \mathbf{s}_2)) \rightarrow f(\mathbf{s}_1, \mathbf{s}_2) : A_1 \mathbf{s}_1 = \mathbf{k} \text{ and } (I + 1 - A_1^T) \mathbf{s}_2 = \mathbf{k} \right\} \quad (13)$$

Equation 13 follows because

$$\begin{aligned}
& \text{If } w(\mathbf{s}_1) > w(\mathbf{s}_2) \\
\implies & w(A_1) < w(I + 1 - A_1^T) \\
\implies & (A_1 + A_1^T)\mathbf{1} < (I + 1)\mathbf{1} \\
\implies & (A_1 + A_1^T)\mathbf{s}_2 < (I + 1)\mathbf{s}_2 \quad (\because \mathbf{s}_2 \text{ is positive}) \\
\implies & A_1\mathbf{s}_2 - (I + 1 - A_1^T)\mathbf{s}_2 < 0 \\
\implies & A_1\mathbf{s}_2 < k
\end{aligned}$$

(and vice-versa with s_1 and s_2 interchanged.)

Now, $w(A_1) = l \implies A_1 = I^{lxl}$. This represents one extreme where the total work done by transmitters is zero, i.e. $W_T = 0$. All work is done by the receivers, i.e. $W_R = l(l - 1)$. In this case, $A_2 = \mathbf{1}$ and $w(A_2) = l^2$. At the other extreme, we have the transmitters creating nulls at every receiver ($w(A_1) = l^2, A_1 = \mathbf{1}, W_T = l(l - 1)$) and the receivers doing zero work ($w(A_2) = l, A_2 = I^{lxl}, W_R = 0$).

To find the optimal solution, we evaluate the right hand side of Equation 13 for all A_1 with weight ranging from $w(A_1) = l$ [corresponding to $W_T = 0, W_R = l(l - 1)$] to $w(A_1) = l^2$ [corresponding to $W_T = l(l - 1), W_R = 0$]. We obtain a set of $l \cdot (l - 1) + 1$ solutions, each being maximal over the class of matrices having a certain weight. The optimal solution is the maximum of this set. In practice, we do not need to sweep the weight of A_1 beyond the midpoint ($W_T = W_R = \frac{l(l-1)}{2}$) up to $W_T = l(l - 1)$ because of the following property.

Lemma 1 *The MIMO system under consideration is equivalent to its dual configuration, obtained by reversing the direction of every communication link.*

Proof: We model transmitters and receivers identically, i.e. transmitter and receiver arrays have equal numbers of antenna elements and identical signal processing capabilities. Thus, reversing the roles of transmitters and receivers and the directions

of data transfer, preserves the optimal total number of streams. The roles of A_1 and A_2 are then reversed, i.e. A_1 is the receiver side matrix and A_2 is the transmitter side matrix. ■

5.2.2 Lagrange Multiplier Method of Optimization: SR+SM

For every value ‘ w ’ of $w(A_1) \in [l, l^2]$, we apply the Lagrange Multiplier Method of optimization to find the maximal solution

$$\begin{aligned} \mathbf{s}_m^w = \max_{A_1} \left\{ w(f(\mathbf{s}_1, \mathbf{s}_2)) \rightarrow f(\mathbf{s}_1, \mathbf{s}_2) : A_1 \mathbf{s}_1 = \mathbf{k} , \right. \\ \left. (I + 1 - A_1^T) \mathbf{s}_2 = \mathbf{k} , w(A_1) = w \right\} \end{aligned} \quad (14)$$

Finally, the optimal solution is calculated as

$$\mathbf{s}_m = \max_{w(A_1)=w} \{ \mathbf{s}_m^w \} \quad (15)$$

We will see that this maximum occurs when $w(A_1) = \frac{l(l+1)}{2}$. Moreover, Equation 12 gives $w(A_2) = l(l+1) - w(A_1) = w(A_1)$. Let the weights of A_1 and A_2 be

$$\begin{aligned} w(A_1) &= l + n \text{ where } 0 \leq n \leq l \cdot (l - 1) \\ \implies w(A_2) &= l^2 - n \end{aligned}$$

We have, for the transmitter side,

$$\begin{aligned} A_1 \mathbf{s}_1 &= \mathbf{k} \\ \implies s_1 c_1^T + s_2 c_2^T + \dots + s_l c_l^T &= \mathbf{k} \end{aligned}$$

where c_i is the i^{th} column of A_1 and s_i is the i^{th} element of \mathbf{s}_1 . Denoting $w(c_i)$ by A_i^w ,

$$s_1 A_1^w + s_2 A_2^w + \dots + s_l A_l^w = kl \quad (16)$$

We also have

$$\begin{aligned}
 w(A) &= l + n \\
 \implies A_1^w + A_2^w + \dots A_l^w &= l + n
 \end{aligned} \tag{17}$$

We want to maximize the function

$$f(\mathbf{s}_1, \mathbf{A}_1) = s_1 + s_2 + \dots s_l \tag{18}$$

$$\text{where } \mathbf{s}_1 = (s_1, \dots s_l) \text{ and } \mathbf{A}_1 = (A_1^w, \dots A_l^w)$$

subject to the following two constraints:

$$\phi(\mathbf{s}_1, \mathbf{A}_1) = s_1 A_1^w + s_2 A_2^w + \dots s_l A_l^w$$

$$-kl = 0 \text{ and}$$

$$\theta(\mathbf{s}_1, \mathbf{A}_1) = A_1^w + A_2^w + \dots A_l^w$$

$$-(l + n) = 0$$

This is done by the method of Lagrange multipliers as follows. Define

$$\begin{aligned}
 F(\mathbf{s}_1, \mathbf{A}_1, \lambda, \mu) &= f(\mathbf{s}_1, \mathbf{A}_1) - \lambda \phi(\mathbf{s}_1, \mathbf{A}_1) - \mu \theta(\mathbf{s}_1, \mathbf{A}_1) \\
 &= s_1 + s_2 + \dots s_l - \lambda(s_1 A_1^w + \dots + s_l A_l^w \\
 &\quad - kl) - \mu(A_1^w + \dots + A_l^w - (l + n))
 \end{aligned}$$

Then, we solve the system

$$\frac{\delta F}{\delta s_i} = 0 \quad \forall i = 1 \dots l \tag{19}$$

$$\frac{\delta F}{\delta A_i^w} = 0 \quad \forall i = 1 \dots l \tag{20}$$

$$\phi(\mathbf{s}_1, \mathbf{A}_1) = 0 \tag{21}$$

$$\theta(\mathbf{s}_1, \mathbf{A}_1) = 0 \tag{22}$$

This gives

$$A_1^w = \dots = A_l^w = \frac{1}{\lambda} \text{ (from equation 19)}$$

$$s_1 = \dots = s_l = \frac{-\mu}{\lambda} \text{ (from equation 20)}$$

$$l \frac{-\mu}{\lambda} \frac{1}{\lambda} = kl \text{ (from equation 21)}$$

$$\implies \mu = -k\lambda^2$$

$$l \frac{1}{\lambda} = (l+n) \text{ (from equation 22)}$$

$$\implies \lambda = \frac{l}{l+n}$$

Finally, we have

$$A_1 = \dots = A_l = \left(\frac{n}{l} + 1\right) \text{ and}$$

$$s_1 = \dots = s_l = k \cdot \lambda = k \frac{l}{l+n}$$

And so, the maximum value of the function $f(\mathbf{s}_1, \mathbf{A}_1) = s_1 + s_2 + \dots + s_l$ is

$$w_m(\mathbf{s}_1, n) = k \frac{l^2}{l+n}$$

Similarly, for the receiver side, we obtain

$$w_m(\mathbf{s}_2, n) = k \frac{l^2}{l^2 - n}$$

Evaluating the optimal solution from equations 14 and 15 amounts to evaluating

$$\begin{aligned} w(\mathbf{s}_m) &= \max_n \left\{ \min \{ w_m(\mathbf{s}_1, n), w_m(\mathbf{s}_2, n) \} \right\} \\ &= k \frac{2l}{(l+1)} \end{aligned} \tag{23}$$

This maximum occurs at $n = \frac{l(l-1)}{2}$. Correspondingly, $\lambda = \frac{2}{l+1}$. Therefore

$$w(A_1) = l + n = \frac{l(l+1)}{2}$$

$$w(A_2) = l^2 - n = \frac{l(l+1)}{2}$$

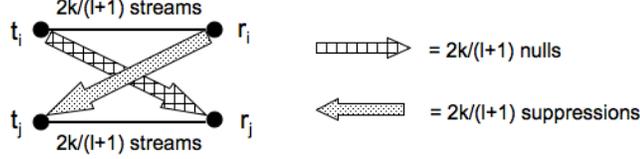


Figure 7: Interference Cancellation between Two Links l_i and l_j

Hence, optimal total number of streams \mathbf{s}_m occurs at the mid-point where $w(A_1) = w(A_2)$ i.e. transmitters and receivers share work equally. $S_{max} = w(\mathbf{s}_m) = k \frac{\binom{2l}{l}}{l+1} = \frac{2kl}{l+1}$. For this to be integral, k should be a multiple of $\frac{l+1}{2}$.

This situation is depicted in Figure 7. The total number of bi-directional interference cancellations a given link can achieve in this manner is $\frac{k - \frac{2k}{l+1}}{\frac{2k}{l+1}} = \frac{l-1}{2}$. Note that in the case under consideration, l must be odd so that this value is an integer. So, any given link can cancel interference in both directions with exactly $\frac{l-1}{2}$ other links and will completely use its DOFs to do so. Interference must be canceled between all $\binom{l}{2} = l \cdot \frac{l-1}{2}$ pairs of links. Since each link can cancel interference with $\frac{l-1}{2}$ others, the number of possible cancellations matches the number required. Our analysis cannot be completed before making the following important observation.

Note: The values of A_i^w should be integral as these are the weights of the columns of A_1 , which have ‘1’ and ‘0’ as entries. However, we have disregarded this fact and carried out the Lagrange Multiplier Method of optimization in the Real domain. At the optimal point (mid point), the value of the column weights, $\frac{1}{\lambda} = \frac{l+1}{2}$ is integral if l is odd. Furthermore, optimal values of A_i^w yielded by the Lagrange Method are integral also for those values of n which are multiples of l . For all other values of n , the column weights obtained are non-integer. Given that we have determined that the value of $\min\{w_m(\mathbf{s}_1, n), w_m(\mathbf{s}_2, n)\}$ is strictly lower than $k \frac{2l}{l+1}$ for all n other than $n = \frac{l(l-1)}{2}$, imposing the integer constraint would only further strengthen the inequality. Therefore the approach is justified, and we are safe in performing the optimization in the Real domain.

An Example: Table 2 compares the number of active streams achievable for the

Table 2: No.of active streams with SR only and with SR+SM ($k = 8$)

No. of links	1	3	5	7	9	11	13	15
No. of streams(SRO)	1	3	5	7	9	11	13	15
No. of streams (SR+SM)	8	12	12	14	14	14	14	15

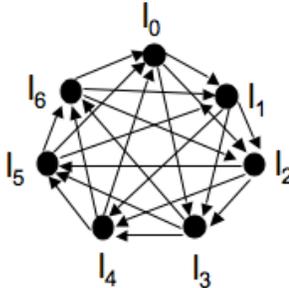


Figure 8: Interference Cancellation Assignment for $l = 7$

two scenarios: SRO and (SR+SM) when $k = 8$.

5.2.3 Structure of the Matrices at Optimal Point: SR+SM

At the optimal point, we have $w(A_1) = w(A_2)$. Moreover, the outcome of the Lagrange Method gives the weight of each row and of each column of A_1 and A_2 to be equal to $\frac{(l-1)}{2} + 1$. This result translates to our MIMO setting to mean that every transmitter and every receiver performs interference cancellation with $\frac{(l-1)}{2}$ other links. That is to say, the work of IC is equally distributed among all transmitters and receivers. Finally, the relation $A_2 = I + 1 - A_1^T$ implies $A_1 = A_2$. Hence, the transmit and receive matrices are equal at the optimal point. We can therefore write

$$\begin{aligned}
 A_1 = A_2 &\implies A_1 + A_1^T = I + 1 \implies \\
 &\begin{cases} A_1(i, j) = 1 - A_1(j, i) \quad \forall i \neq j \\ A_1(i, j) = A_2(i, j) \end{cases} \quad (24)
 \end{aligned}$$

An example of this assignment for $l = 7$ is shown in Figure 8. Every transmitter performs IC at $\frac{7-1}{2} = 3$ receivers and similarly, every receiver performs IC with 3 transmitters.

Equation set 24 is the ***Symmetry Condition***. The symmetry condition implies that if transmitter i nulls itself at receiver j , then transmitter j will not null itself at receiver i . Instead, it is receiver i which suppresses the signal from transmitter j . *Therefore, interference cancellation between a pair of links is done entirely by one of the links at the optimal point.*

5.2.4 Handling Primary Interference

Primary interference occurs when a station is involved in more than one communication task at the same time (sending and receiving, receiving from two different transmitters, etc.). Let $G = (V, E)$ be a subgraph of the communication graph of the system containing all links that have data to transmit. In general, the links of G are *not* free of primary interference.

A matching of G is a set of edges, where each vertex appears in at most one edge of the matching. Thus, by definition the links making up a matching of G are free of primary interference. The following theorem demonstrates that, if we obtain a maximum matching M of G and apply our optimal construction from the previous sections to the links contained in M , then this achieves the maximum total number of streams possible among all of the links in G . This then provides an optimal solution to the one-shot stream scheduling problem under consideration.

5.2.5 Optimal total number of streams in an SCD with Uniform Rate Model

In subsections 5.2.1, 5.2.2, and 5.2.3, we showed that the optimal total number of streams that can be scheduled on l links is $\frac{2kl}{l+1}$, with every link multiplexing an equal number $\frac{2k}{l+1}$ of streams. See equation 23. Since the minimum number of streams an active link can multiplex is one, we must have

$$\frac{2k}{l+1} \geq 1 \tag{25}$$

$$\implies l \leq 2k - 1 \tag{26}$$

This result implies that for a given value of k , there is a limit to the number of links that can be concurrently active in an SCD. In other words, the maximum size of the SCD equals $2k - 1$. This is the case of *spatial reuse only*. Since each link transmits only one stream, no spatial multiplexing is done. This is stated in the lemma below (See Lemmas 1 and 2 in our work [49]).

Lemma 2 *The maximum size, l , of a single collision domain where every node has an array size of k , is equal to $2k - 1$ i.e. $l \leq 2k - 1$*

Proof: See discussion above. ■

Furthermore, equation 23 leads to the result that the minimum value of the optimal total number of streams that can be scheduled in an SCD is k streams. This occurs when l equals one. This is the case of *spatial multiplexing only* where a single link is active and it employs all its DOFs to multiplex k streams.

The maximum value of the optimal total number of streams in an SCD asymptotically equals $2k$. This is when k is arbitrarily large. In practice however, the value of k is usually around 4 to 8 antennas. Therefore the maximum total number of streams in practice equals $2k - 1$, which is achieved when $l = 2k - 1$ links are scheduled with one stream on each link.

Theorem 1 *Let M be a maximum matching among all links having data to transmit. Let the number of links in M be $l \in [1, 2k - 1]$ such that $\frac{2k}{l+1} \in Z$. This implies $l = 2m + 1$ for some $m \in N$ and k is a multiple of $(m + 1)$. Then the optimal total number of streams is $S_{max} = \frac{2kl}{l+1}$.*

Proof: In subsections 5.2.1, 5.2.2, and 5.2.3, it was proven that in the absence of primary interference, the optimal total number of streams supported by a set of $l \leq 2k - 1$ links in a single collision domain is $S_{max} = \frac{2kl}{l+1}$ for odd l . Since this is an increasing function of l , no set of links which form a matching smaller than l

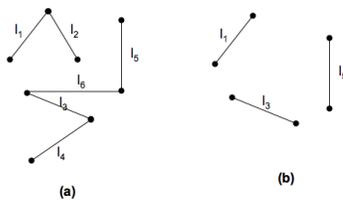


Figure 9: (a) Communication graph in a single collision domain (b) Maximum matching of the communication graph

(size of the maximum matching) can support a higher total number of streams. (See Theorem 3 in our work [49]) ■

Note that Theorem 2 holds only if the number of links in the maximum matching is odd. Exactly characterizing the optimal solution for an even number of links is an open problem. However, the optimal constructions for odd numbers of links can be used to achieve bounds on optimality for even numbers of links. For example, in Table 2, observe that for 6 links, we can achieve 12 streams using the optimal construction for 5 links. Furthermore, the optimal for 6 links can be no better than the optimal for 7 links, which achieves 14 streams. Therefore, the 5-link construction is within two streams of optimal for 6 links. Note also that as the number of links increases, this bound gets tighter. From the same table, we can see that the optimal construction for 7 links is also guaranteed to be optimal for 8, 10, and 12 links. In general, if we use the optimal construction for $l - 1$ links for an even l , we can get within at least $\frac{4k}{l(l+2)}$ of optimal, which decreases in proportion to l^2 .

5.2.6 An Example

Consider a single collision domain with 6 links as shown in Figure 9a. Assume all links have data to transmit. We want to schedule the maximum number of streams possible in one time slot across these links. Obtain a maximum matching of size 3 as shown in Figure 9b. Thus we get the maximum number of primary-interference-free links to be $l = 3$. Choose the size of the antenna array to be $k = 2$. We will apply the optimization procedure derived above to this setting in order to find the maximal

Table 3: Output of the Lagrange multiplier optimization method.

n	0	1	2	3	4	5	6
$w_m(\mathbf{s}_1, \mathbf{n})$	6	4.5	3.60	3	2.5714	2.25	2
$w_m(\mathbf{s}_2, \mathbf{n})$	2	2.25	2.5714	3	3.60	4.5	6
$w(\mathbf{s}_m)$	2	2.25	2.5714	3	2.5714	2.25	2

total number of streams and the structure of the A_1 and A_2 matrices at the optimal point.

The weight of A_1 is swept from $n = 0$ to $n = 6$. Note the symmetry about the mid-point, which is a result of the duality property. For values of n smaller than the mid-point i.e. for $n < 3$, we have $w(A_1) < w(A_2)$. This implies $W_T < W_R$ i.e. the total work done by the transmitters in IC is less than that done by the receivers. The number of degrees of freedom available at the transmitters for multiplexing data streams is therefore larger than that available at the receivers. Hence the receiver side matrix A_2 determines (imposes a stronger constraint on) the achievable total number of streams. This is evident from Table 3. On the other hand, for values of n larger than the mid-point i.e. for $n > 3$, we have $w(A_1) > w(A_2)$. In this case, the achievable total number of streams is constrained more strongly by the transmit side matrix A_1 as seen in the table.

5.2.7 Discussion

In this section, we qualitatively compare our results with the ones (based on numerical evaluation) reported in [8]. We stress that the network setting considered in [8] is quite different from ours: multi-hop flows are to be scheduled on a set of links with arbitrary collision domains. On the other hand, our approach assume single links to be scheduled (one-hop flows), and all links are part of a single collision domain. Despite the different network settings, the main qualitative findings of [8] are fully confirmed by our analytical results. To be specific, Hamdoui, Shin, and Jiang observe that, as

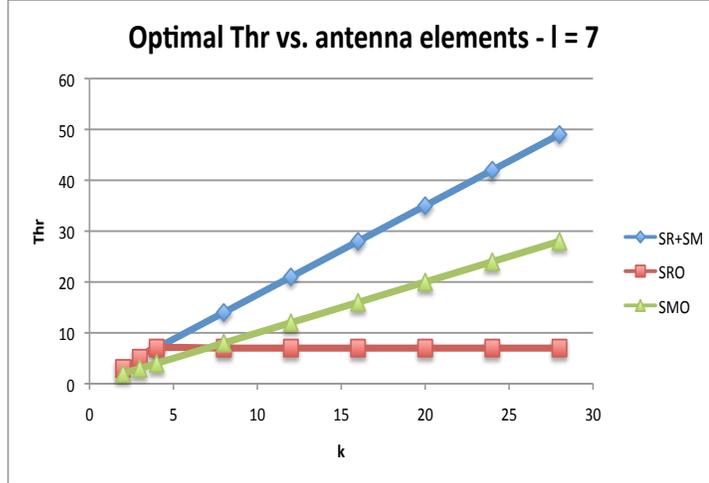


Figure 10: Optimal total number of streams achievable with $l = 7$ links and increasing number of antenna elements, with SR only, SM only, and SM+SR MIMO systems.

the number of antenna element increases, the maximum achievable total number of streams first raises and then flattens out asymptotically under SRO, while it increases “almost” linearly under SMO or SR+SM. As seen from Figure 10, this behavior can be observed also when the results derived in the previous subsections are extended to arbitrarily large values of k : in case of SRO, the total number of streams increases when relatively few DOFs are available; once the available DOFs are sufficient to null/suppress all interference, the optimal total number of streams flattens to the optimal value of l , corresponding to scheduling one stream on each possible link. In case of SMO, only one link can be active at a time. Hence, optimal total number of streams increases linearly with k , which corresponds to the maximum possible number of streams that can be transmitted on the active link. In case of SM+SR, additional total number of streams benefit (near two-fold) can be achieved by combining the two MIMO techniques. It is also worth observing that when relatively few DOFs are available ($k \leq 5$), all DOFs are used to mitigate interference (SRO and SM+SR curves are overlapped); as the number of available DOFs increases, interference mitigation can be combined with spatial multiplexing to achieve considerable total-number-of-streams gains over the SRO and SMO approaches.

5.3 Complexity of MIMO Feasibility

In this section, we study the complexity of checking the feasibility of a stream allocation vector in a MIMO network. As is made clear in the formal problem definition given in Section 4.1, feasibility is a special type of Boolean satisfiability problem. It is well known that many variations of Boolean satisfiability are NP-complete. Due to the specialized constraints in the MIMO feasibility problem, it is unlikely that a proof of NP-completeness will be found, but we conjecture that MIMO feasibility is NP-complete. However, certain special cases of the feasibility problem are solvable in polynomial time and we provide proofs of this for several cases in this section.

The fact that feasibility can no longer be trivially solved with MIMO links could have important implications for scheduling algorithms. As mentioned earlier, many greedy scheduling algorithms attempt to assign links to the first slot in which they are feasible. This common approach assumes that feasibility can be efficiently tested, so that repeated execution of feasibility checks does not negatively impact the execution time of the scheduling algorithm. Since this assumption is not valid for the general MIMO link scheduling case, alternative approaches to building schedules might have to be developed. For example, approaches that build schedules, which are provably feasible by the manner in which they are constructed and thus do not have to employ feasibility tests, could be preferable.

5.3.1 Receiver-Side Suppression

When CSI is available only at the receivers and not at the transmitters, then only receiver side interference suppression can be done. Theorem 2 states that, in this special case, the feasibility problem is polynomial time in complexity.

Theorem 2 *Checking feasibility of a stream allocation vector S and a link set L over an arbitrary MIMO network with receiver-side-suppression only can be done in polynomial time.*

Proof: Let $S = [s_1, \dots, s_m]$ and recall that $s_i > 0$, for all i (otherwise, we can simply remove link i from L and consider feasibility of the smaller vector and link set). Denote the conflict graph of the network by $G_c = (L, E_c)$. Note that Condition 3 of feasibility (see Section 4.1) says that for all distinct links i and j that conflict with each other, either $a_{ij}^t = 1$ (meaning t_i suppresses its interference on r_j) or $a_{ji}^r = 1$ (meaning r_j suppresses interference from t_j). Since only receiver-side suppression is done in this case, $a_{ij}^t = 0$ and in order to satisfy Condition 3, it is required that $a_{ji}^r = 1$. Thus, every receiver must necessarily suppress interference from every transmitter whose link conflicts with the receiver's link.

Checking feasibility amounts to checking whether all of the conditions from the feasibility problem definition are true. In light of the fact that $A^t = 0$ with receiver-side-suppression only, the following procedure suffices to check feasibility.

Step 1: Check that L is primary-interference-free. This can be done by simply scanning through all links and counting the number of occurrences of every node. If any node appears more than once, L is not primary-interference-free and S and L are not feasible. If all nodes appear at most once in L , then continue to Step 2.

Step 2: Check whether there exist A^t and A^r that satisfy Conditions 1-3 of feasibility. From the above discussion A^t is the matrix of all zeros and therefore Condition 1 ($A^t S \leq K^t$) is trivially satisfied. Since A^t is all zeros, A^r is therefore fixed by Condition 3. Therefore, it is only necessary to check that Condition 2 is satisfied for every link. In the case under consideration, for a given receiver r_i , Condition 2 becomes:

$$s_i + \sum_{j: j \neq i \text{ and } (l_i, l_j) \in E_c} s_j \leq k_i^r$$

Since all s_i 's are given by the input stream allocation vector S , checking this condition amounts to simply checking the above inequality for every receiver. This can be easily done in $O(l^2)$ time.

■

The essence of what makes feasibility polynomial time in the special case of receiver-side-suppression only is that the choice of how to suppress interference (either by the transmitter of the interfering link or by the receiver of the interfered with link) is removed. In the general MIMO case, for every pair of interfering links, there is a choice as to how to suppress the interference, and combining these choices over all pairs of interfering links yields an enormous number of possibilities that are all potential ways to make a stream allocation vector feasible. The case of transmitter-side-suppression only also removes the choice of how to suppress interference and, therefore, it has the same effect on problem complexity, i.e. feasibility for the transmitter-side-suppression only case is also of polynomial complexity.

5.3.2 Maximum Antenna Array Size $K = 2$

Another interesting special case is when the DOFs of all nodes are small. In particular, when every node in the network has $k = 2$ DOFs, even when interference suppression can be done at *both* transmitter side and receiver side, then the feasibility problem is polynomial time in complexity. This result is stated in Theorem 2.

Theorem 3 *Checking the feasibility of a stream allocation vector S and a link set L over an arbitrary MIMO network where every node has $k = 2$ degrees of freedom is a polynomial time operation.*

Proof: The proof is constructive, i.e., we describe a polynomial time algorithm that, given inputs S and L , returns **True** if and only if stream allocation vector S is feasible for link set L . The algorithm first checks whether L is primary-interference-free in polynomial time (as in the proof of Theorem 2). If L is not primary-interference-free, the algorithm returns **False**, otherwise it continues with the procedure described next.

Let the conflict graph of the MIMO network be $G_c = (L, E_c)$. Every active link l_i carries a number of streams $s_i = \{1, 2\}$. Inactive links (with zero streams allocated)

are not represented in G_c . Let $L_2 = \{l_i \in L : s_i = 2\}$. Since each link in L_2 utilizes its full multiplexing capacity, no resources for interference suppression are available. The remaining links are contained in the set $L_1 = L \setminus L_2$, composed of links carrying a single stream.

The feasibility algorithm first checks whether all links of L_2 are isolated vertices in G_c . If not, the algorithm returns **False**, otherwise, it considers the subgraph G_1 of G_c induced by node set L_1 . Let G^1, \dots, G^h be the connected components of graph G_1 . The algorithm checks whether for each $G^i = (L^i, E^i)$, inequality $|E^i| \leq |L^i|$ is satisfied; if the inequality is not satisfied for any of the G^i , the algorithm returns **False**, otherwise it returns **True** and terminates.

It is immediate to see that the above algorithm has polynomial time complexity. We now prove that, when the algorithm returns **False** on input S, L , stream allocation vector S is infeasible for L . To prove this, we observe that the algorithm returns **False** if only if one of the following conditions hold:

- 1) set L is not primary-interference-free; in this case, it is clear that any non-zero stream allocation vector S for L is infeasible.
- 2) L_2 contains at least one link, which is not an isolated vertex in G_c ; denote such a link by l_i and suppose it is adjacent to link l_j in the conflict graph. Since l_i carries two streams, it has no DOFs available for suppression. Link l_j carries at least one stream and, therefore, has at most one DOF remaining, which is not enough to suppress the two streams on l_i . Hence, condition (3) for feasibility cannot be satisfied for links l_i, l_j unless conditions (1) and (2) are violated. This implies that stream assignment S is not feasible for link set L .
- 3) there exists a connected component G^j of graph G_1 such that $|E^j| > |L^j|$. A simple counting argument can be used to prove that S is not feasible for L : for each link $l \in L^j$, two DOFs are available at the link endpoints to suppress

interference (one at the transmitter and one at the receiver side). Thus, $2|L^j|$ DOFs in total are available to suppress interference within G^j . On the other hand, suppressing interference between any two adjacent links l_i, l_j in the conflict graph requires using 2 DOFs: one for suppressing interference generated by t_i on r_j , and one for suppressing interference generated by t_j on r_i . Thus, $2|E^j|$ DOFs in total are needed to suppress the interference the $|L^j|$ links in G^j cause to each other receivers. Hence, $|E^j| > |L^j|$ implies that not enough radio resources (DOFs) are available within G^j to completely suppress interference, which proves that stream allocation vector S is infeasible for L .

The next step is to prove that whenever none of conditions 1), 2), 3) hold on given input S, L , then stream allocation vector S is feasible for L , which implies correctness of our feasibility checking algorithm (which returns **True** in this situation). We prove this last step by showing a construction (DOF assignment) that makes S feasible for L subject to the fact that none of the conditions 1), 2), 3) are satisfied.

If condition 3) is not satisfied, we have $|L^j| \leq |E^j|$ for each connected component G^j of G_1 . We first observe that DOF assignments for the G^j s can be built independently, since links in different G_1 connected components do not interfere with each other. We hence show the construction for a single G^j , making the overall construction the result of the composition of DOF assignments for the single connected components. It is not difficult to see that the topology of G^j can take only one of the four following forms: a) single vertex; b) tree; c) simple cycle; d) connected graph containing a single simple cycle. If G^j is of type a), no DOF has to be allocated for interference suppression. If G^j is a tree (type b)), perform the following procedure:

1. Designate some vertex in L^j to be the root.
2. For every edge $(l_i, l_k) \in E^j$, use the two available DOFs of the link deeper in the tree (say, l_k) to suppress mutual interference between l_i and l_k .

It is easy to see that, since every vertex in a tree (except the root) has a single parent, each link in the above construction uses at most 2 DOFs to suppress interference, thus not exceeding the available DOFs. On the other hand, mutual interference between all links in G^j is taken care of at the end of the above procedure, implying that the resulting DOF assignment makes S feasible (when restricted to G^j).

Let us now consider case c). In this case, it is sufficient to give either clock-wise or counterclock-wise orientation to the edges in E^j , and to choose an arbitrary vertex l_i in L^j . Consider any two adjacent vertices l_s, l_t in G^j , and assume w.l.o.g. that l_s precedes l_t in the chosen orientation, starting from l_i . Then, the two DOFs available at l_s are used to suppress mutual interference between l_s and l_t . It is easy to see that, similarly to what happens in case b), this construction results in a feasible DOF assignment for S (when restricted to G^j).

Finally, consider case d). In this case, we start by designating every vertex in L^j that is contained in the simple cycle and is of degree equal to 3 as the root of the tree component it belongs to. DOFs are then assigned by combining the construction for case b) within the trees, with construction for case c) along the single simple cycle contained in G^j . Note that the resulting construction is feasible since root vertices in construction b) do not use their available DOFs to suppress interference with other links in the tree; hence, these available DOFs can be used to suppress interference with the successive vertex (link) in the simple cycle as described in the construction for case c). Thus, the resulting DOF assignment makes S feasible (when restricted to G^j), and the theorem is proved. ■

5.3.3 Maximum Antenna Array Size $K=3$

Consider a MIMO network where every node has $K = 3$ antennas. For simplicity, interference between links is assumed to be symmetric. The resulting conflict graph is an undirected graph, call it $G = (V, E)$. We include in G only links that 1) carry a

non-zero stream allocation and that 2) carry less than three streams. For feasibility, any links with three streams must form an independent set. Since this can be easily checked in polynomial time, we do this as a pre-processing step and then remove these links from the graph. Links that are not allocated any streams can also be safely ignored. Thus, links in G can carry either one or two streams.

We first consider the special case of checking feasibility when all links carry exactly one stream in a MIMO network. This problem can be re-formulated as below:

Problem 1 - Can the edges of graph G be directed such that every vertex has at most two outgoing edges?

Theorem 2 states that the $K = 3$ special case of feasibility, where every link carries exactly one stream is equivalent to checking whether the conflict graph G has any subgraphs of average degree greater than 4.

Theorem 1 *Let D_1 be the property of a graph $G = (V, E)$, whereby every vertex induced subgraph of G has an average degree at most equal to four. D_1 is necessary and sufficient for the edges of G to be directed such that every vertex has at most two outgoing edges.*

Proof:

Necessary condition: Assume the edges of G can be directed such that every vertex has at most two outgoing edges. We will prove that Property D_1 holds, i.e. that all subgraphs of G have average degree no greater than 4. Consider an arbitrary subgraph G_1 with n_1 vertices. Since, in some complete labeling of G each vertex has at most two outgoing edges, the total number of edges in the subgraph can be at most $2n_1$. Since each edge is incident on two vertices, the average degree is at most $\frac{2 \cdot 2n_1}{n_1} = 4$.

Sufficient condition: Suppose the given graph G satisfies D_1 . We prove the sufficient condition through construction, by determining a direction for all of G 's edges such that every vertex has at most two outgoing edges. The construction is described

in Procedure Proc I, which is given after the following definitions. The quantities defined by these definitions are dynamic, i.e. they are recalculated dynamically as the construction proceeds.

1. Let the quantity n_v denote the number of remaining edges that can be marked as outgoing from vertex v in V . At the start, $n_v = 2$ for all vertices v . At any intermediate point during the construction, this value equals two minus the number of edges that have already been marked as outgoing from v . The construction does not allow more than 2 edges to be marked as outgoing for any vertex and, therefore, $0 \leq n_v \leq 2$ always.¹
2. Define for any subgraph $G_{sub} = (V_{sub}, E_{sub})$ of G the quantity ED (ED stands for ‘extra DOF’s’). Let $E_{um} \subseteq E_{sub}$ be the set of edges of G_{sub} that are not yet marked with a direction.

$$ED(G_{sub}) = \sum_{v \in G_{sub}} n_v - |E_{um}|$$

Property D_1 implies that ED is greater than equal or to zero for every subgraph, at the beginning of Procedure Proc 1.

3. Define for any edge (v, v') in E , the boolean quantity DO (DO stands for ‘directable outwards’):

$$DO(v, v') = \bigwedge_{\forall G_{sub}} (ED(G_{sub}) > 0)$$

where G_{sub} is a vertex-induced sub-graph of G/v' such that it contains vertex v and \bigwedge refers to the Boolean AND operation.

The DO definition is illustrated in Figure 11. If all subgraphs containing v but not v' have “extra DOFs”, then it is safe to direct the edge (v, v') outwards.

¹The “number of remaining edges n_v that can be marked as outgoing from v ”, refers to the number of DOF’s that are available for interference suppression at the transmitter and at the receiver of link v in the MIMO feasibility problem.

$$DO(v,v')=(ED1>0)\wedge(ED2>0)\wedge(ED3>0)\wedge(ED4>0)\dots$$

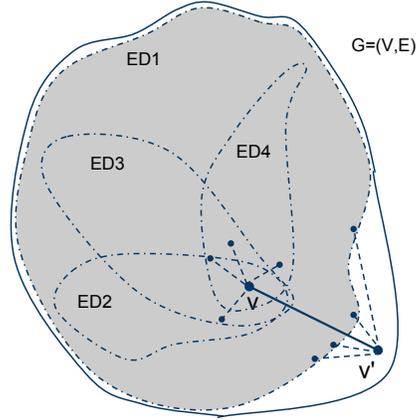


Figure 11: Illustration of definition: $DO(v, v')$

Keeping the above definitions in mind, apply Procedure Proc 1 to a graph $G = (V, E)$ which satisfies D_1 . By definition, *every* subgraph of G has an ED value greater than equal to zero at the beginning of the procedure.

Begin Procedure Proc I

Input: $G = (V, E)$ satisfying Property D_1

Output: $f : E \rightarrow \{0, 1, \dots, n\}$, where $f((u_i, u_j)) = i$ indicates that the edge is directed from u_i to u_j and $f((u_i, u_j)) = j$ indicates that the edge is directed from u_j to u_i

1. **Repeat:** If any vertex v_i in V has all but p edges marked as incoming, where $p \in \{1, 2\}$, mark these p edges as outgoing, i.e. $f((v_i, v_j)) = i$ for these edges, and set $n_{v_i} = n_{v_i} - p$

Until: No such vertex exists

2. $V_n =$ the set of all vertices with at least one unmarked edge
3. **while** there exists a vertex v_i in V_n with $n_{v_i} = 2$ (i.e. with *no* outgoing edges assigned)
 - 3a. Let $v_j, j = 1, 2, 3\dots$ be the neighbors of v_i connected by unmarked edges

3b. $j = 1$

3c. **while** ($n_{v_i} = 2$)

if ($DO(v, v_i) = \text{TRUE}$)

 Mark (v_i, v_j) as outgoing, i.e. $f(v_i, v_j) = i$

$n_{v_i} = n_{v_i} - 1$

else

 Mark (v_i, v_j) as incoming, i.e. $f(v_i, v_j) = j$

$n_{v_j} = n_{v_j} - 1$

end if

if (v_i has only two unmarked edges and $n_{v_i} = 2$)

 Mark the remaining two edges incident to v_i as outgoing, i.e. $f((v_i, v_j)) = i$

for these two edges

$n_{v_i} = n_{v_i} - 2$

end if

$j = j + 1$

end while

3d. **Repeat:** If any vertex v_i in V has all but $p \leq n_{v_i}$ edges marked as incoming, mark these p edges as outgoing, i.e. $f((v_i, v_j)) = i$ for these edges, and set $n_{v_i} = n_{v_i} - p$

Until: No such vertex exists

3e. **end while**

4. **while** there exists a vertex v_i in V_n with $n_{v_i} = 0$

 mark the remaining unmarked edges incident to v as incoming, i.e. $f(v_i, v_j) =$

j for these edges

end while

5. Let $G' = (V', E')$ be the graph that results from removing all marked edges from G and removing all vertices that have every edge marked
6. Direct the edges of G' according to the procedure used for the $k = 2$ case presented in our earlier work [54]

End Procedure

Analysis of procedure:

Step 1: At the start of the procedure, every subgraph of G has $ED \geq 0$. During this step, all edges of degree-one and degree-two vertices in V are marked.

Step 2: V_n contains only vertices in V with degree > 2 .

Step 3: If during the procedure, we determine $DO(v, v')$ to be **TRUE**, that means that *every* subgraph containing v but not v' has $ED > 0$. The procedure is allowed to direct vertex v towards v' *if and only if* $DO(v, v') = \text{TRUE}$. Consequently, after the procedure directs v outwards to v' , *every* subgraph containing v but not v' will have $ED \geq 0$. Moreover, all subgraphs containing v' as well as all subgraphs with neither v nor v' will have an unchanged ED value.

On the other hand, if $DO(v, v') = \text{FALSE}$ and $DO(v', v) = \text{TRUE}$ then the procedure directs v' outwards to v . The procedure is allowed to direct vertex v' towards v *if and only if* $DO(v', v) = \text{TRUE}$. By the same argument as above, after doing this, *every* subgraph containing v' but not v will have $ED \geq 0$ and other subgraphs will not be affected.

Therefore, if the $DO(v, v_i) = \text{TRUE}$ condition in Step 3c is entered or the else condition is entered with $DO(v_i, v) = \text{TRUE}$, then all subgraphs will maintain the property that $ED \geq 0$.

Next, we show that at no point during the procedure, can both $DO(v, v')$ and $DO(v', v)$ simultaneously evaluate to **FALSE**. This is stated in Lemma L1.

Lemma L1: $DO(v, v') | DO(v', v)$ is always equal to **TRUE** at every iteration of Step 3c.

Proof of lemma:

If we were to have $DO(v, v') = FALSE$ and $DO(v', v) = FALSE$, that would imply that v is contained in some subgraph $G_1 = (V_1, E_1)$ such that it does not contain v' and has $ED = 0$. Also, v' is contained in some subgraph $G'_1 = (V'_1, E'_1)$ such that it does not contain v and has $ED = 0$. And since an edge exists between v and v' , the subgraph induced in G by $V_1 + V'_1$ has $ED = -1$. Since *every* subgraph of the original graph G had $ED \geq 0$, this means that at some previous point in the procedure, the subgraph induced in G by $V_1 + V'_1$ had $ED \geq 0$. Consider the point in the procedure when the value of this ED was exactly equal to zero. Suppose without loss of generality that G_1 had an ED value equal to 1 and G'_1 had an ED value equal to 0 at this point (this would make the ED of the subgraph induced in G by $V_1 + V'_1$ equal to zero). Now, in order for the ED value of the subgraph induced in G by $V_1 + V'_1$ to become -1 from 0, there must have been some vertex v_a from $G_1 = (V_1, E_1)$ that was directed outwards by the procedure to a vertex v_b where v_b is *not* contained in V_1 ² and also *not* contained in V'_1 ³. If v_a was directed outwards to such a v_b , that would imply that the procedure was applied incorrectly since clearly there is a subgraph containing v_a , namely the subgraph induced in G by $V_1 + V'_1$ which has an ED value exactly equal to zero, meaning that we are not allowed to direct v_a to v_b .

End of Lemma L1 proof.

Note that the final if statement in Step 3c does not cause any ED to become less than zero, because if v_i satisfies the condition and its last two edges are marked, this can only increase ED for subgraphs containing v_i .

²if v_b was in V_1 , then the ED value of G_1 would remain one and the ED value of the subgraph induced in G by $V_1 + V'_1$ would still remain zero.

³if v_b was in V'_1 , that would violate the assumption that at this point, the ED value of the subgraph induced in G by $V_1 + V'_1$ is zero (because, since we have said that ED of G_1 is one and ED of G'_1 is zero, the fact that an edge between v and v' exists and the fact that an edge between v_a and v_b exists would make the ED value of this subgraph equal to -1).

Since $DO(v, v')$ is true or $DO(v', v)$ is true at all times and $ED \geq 0$ for every subgraph of G in either of these cases and the final if condition does not cause the ED condition to be violated, we have the following invariant:

Invariant 1 - $ED \geq 0$ for every subgraph of G at every iteration of Step 3, including after the final iteration.

Next, we prove another invariant of the procedure. If the original graph G satisfies D_1 , the procedure will never decrement the value of n_v below zero for all $v \in V$. Therefore, $n_v \geq 0$ is also an invariant. This follows from Lemma L1 and Lemma L2, stated next.

Lemma L2: *At every stage of the procedure, a vertex $v \in V_n$ can have at most n_v neighbors that are contained in subgraphs that exclude v and with ED value equal to zero. Moreover, if v has exactly n_v such neighbors, then it is part of an average-degree-four-subgraph in the original graph G .*

Proof of lemma:

Suppose first that $n_v = 0$. Consider a subgraph $G_1 = (V_1, E_1)$ that contains a neighbor of v , but not v itself. Let the ED value of G_1 be ED_1 . Therefore, the subgraph induced by $V_1 + v$ in G will have $ED = ED_1 - 1$. Since by Lemma L1, the ED value of *every* subgraph of G is always greater than equal to zero, we must have $ED_1 \geq 1$. Therefore, v has $n_v (= 0)$ neighbors that are contained in subgraphs that do not contain v and with $ED=0$.

Similarly, for $n_v > 0$, suppose that v has p such neighbors, contained in subgraphs $G_1 = (V_1, E_1), \dots, G_p = (V_p, E_p)$. The subgraph induced by $V_1 + \dots + V_p + v$ in G will have $ED = ED_1 + \dots + ED_p - p + n_v$. Since by Lemma L1, ED of *every* subgraph of G is greater than equal to zero, we must have $p \leq n_v$.

End of Lemma L2 proof.

Note 1 - Lemmas L1 and L2 preclude the following from occurring at any stage of the procedure: $DO(v, v_i) = FALSE$ and $n_{v_i} = 0$. This is because we must have: 1)

$DO(v, v_i) = TRUE$ and/or $DO(v, v_i) = TRUE$ by Lemma L1, and 2) v_i has $n_{v_i} = 0$ neighbors in subgraphs with $ED = 0$ by Lemma L2. This means that if $n_{v_i} = 0$, then $DO(v, v_i) = TRUE$ will always hold and v can be marked outwards to v_i .

This establishes the following invariant.

Invariant 2 - $n_v \geq 0$ for every $v \in V$ at every stage of the procedure.

Step 4: When Step 3 terminates, every vertex $v \in V_n$ has n_v equal to either one or zero. This follows because, at Step 3c, vertex v with $n_v = 2$ will have at least one of its incident edges marked as outgoing, and hence its n_v value will be decremented by at least one.

During Step 4, every vertex $v \in V_n$ with $n_v = 0$ is fully marked. This is done by marking every incident edge towards v . After this, all marked vertices and edges are removed. Therefore, the remaining vertices v all have $n_v = 1$.

Step 4 is justified by the observation that we made in Note 1 above based on Lemmas L1 and L2, that the following can never occur at any stage of the procedure: $DO(v, v_i) = FALSE$ and $n_{v_i} = 0$. In other words, whenever $n_v = 0$, $DO(v', v)$ is always TRUE where v' is a neighbor of v . Therefore, we are allowed to mark all edges incident to v as incoming.

Step 5: Lemma L3: *If the input graph G satisfies D_1 then every connected component of the resulting graph $G' = (V', E')$ at the end of Step 4 will be a tree or a simple cycle.*

To see this, note that the invariant $ED \geq 0$ holds true for every subgraph of G' . Moreover, every vertex v in V' has $n_v = 1$. Let $G'_i = (V'_i, E'_i)$ be the i^{th} connected component of G . We have $ED(G'_i) = \sum_{v \in V'_i} n_v - |E'_i| = |V'_i| - |E'_i|$. Since by Corollary C1, $ED(G'_i) \geq 0$, we have $|V'_i| \geq |E'_i|$, making every connected component of G' a tree or a simple cycle.

Step 6: When we reach Step 6, every vertex $v \in V'$ has $n_v = 1$ and every connected component is a tree or a simple cycle. In [54], we proved that such a

graph can be directed such that every vertex has at most one outgoing edge and we provided a procedure to do so. By applying this procedure to the remaining graph G' , the result is that every edge of the original graph G is directed such that every vertex has at most two outgoing edges.

■

5.4 Conclusion

5.4.1 Conclusions on SCD

We derived an algebraic expression for the maximum number of streams that can be scheduled when the number of links is odd. This optimal is achieved by a combination of spatial reuse and spatial multiplexing. We showed that optimum total number of streams is achieved when the work of interference cancellation is shared equally between every transmitter and every receiver, and therefore all links multiplex the same number of streams. Moreover, we showed that at the optimum point, the interference between every pair of links is canceled entirely by one of the links. That is, if the transmitter of a given link nulls itself at the receiver of an interfering link, then the receiver of the given link will suppress the signal from the transmitter of the interfering link. Finally, we showed that the optimal total number of streams obtained with spatial multiplexing and spatial reuse combined ($S_{max} = \frac{2kl}{l+1} \approx 2k$ streams) is approximately twice of what it is with spatial multiplexing only (which is k streams, all scheduled on a single link).

From the above result for optimal total number of streams, we obtain a bound on the size l of the SCD when every node has an array size of k . This bound is given by $l \leq 2k - 1$. The maximum number of links that can be simultaneously scheduled is therefore equal to $2k - 1$. This is the case of *spatial reuse only*. The optimal total number of streams in this case is thus equal to $\frac{2kl}{l+1} = \frac{2k(2k-1)}{2k} = 2k - 1$.

5.4.2 Conclusions on Multi-hop Networks

For arbitrary multi-hop networks where CSI is available only at the receivers i.e. when the network is enabled with receiver-side-suppression only, the check for feasibility is a polynomial time operation. In the case of arbitrary multi-hop networks where the maximum array size is limited to two, verifying feasibility was seen to be polynomial in complexity. If a stream allocation vector is feasible on such a communication network (with $k = 2$) then the corresponding conflict graph was seen to have an average vertex degree equal to at most two. This means that the conflict graph is such that each connected component is a tree and/or a simple cycle. Finally, in the case that the maximum array size is equal to three, we showed that allocating a single stream to all links of the communication graph requires the corresponding conflict graph G to be such that every vertex-induced subgraph has average vertex degree at most equal to four. The complexity of verifying this however remains an open problem. We conjecture that is is related to the NP-complete clique problem.

CHAPTER VI

FEASIBILITY: EXPERIMENTAL RESULTS

6.1 Feasibility Heuristics

6.1.1 Simple Greedy and Extended Greedy

Given that the general MIMO feasibility problem is quite possibly NP-complete, heuristics for checking feasibility are necessary. Perhaps the most obvious heuristic is to see whether all interference can be suppressed by greedily allocating DOFs for interference suppression. The algorithm works as follows. Sort the links in order of non-increasing number of allocated streams. Begin with the first link and use its DOFs to suppress interference on the links with which it interferes one by one until all its DOFs are used. Then, move onto the next link and continue until all interference is suppressed or all DOFs are used up, whichever comes first. If all interference can be removed with the available DOFs in the network, the allocation vector is declared to be feasible. We refer to this approach as Algorithm Simple Greedy.

In experimenting with Algorithm Simple Greedy, we found that it tends to concentrate DOFs among small groups of nodes, rather than more evenly distributing those resources across links in the network, and this causes it to frequently label feasible vectors as infeasible. To remedy this problem, we developed the algorithm in Figure 12, which we refer to as Algorithm Extended Greedy. This algorithm, when considering multiple candidate links, all carrying equal number of streams, on which to suppress interference, chooses a target link uniformly at random from the candidates. This tends to produce a better distribution of resources and outperforms Algorithm Simple Greedy. In Figure 12, note that the standard notation $\langle V, W \rangle$ is used to represent the inner product of vectors V and W and that I is the identity

Input: Stream allocation vector S , link set L , K^t , K^r , conflict graph $G_c = (V_c, E_c)$
Output: feasible $\in \{\text{true}, \text{false}\}$, A^t , A^r

- 1: Order S in non-increasing fashion. Permute G_c accordingly.
- 2: $A^t = A^r = I_{|L| \times |L|}$
- 3: for $i = 1 \rightarrow |L|$
- 4: if $\langle A^t(i, 1 : i), S^{1:i} \rangle \leq K_i^t$, distribute 1's in $A^t(i, G_c(i, i+1 : l))$ greedily, giving equal priority to columns of equal weight such that $\langle A_i^t, S \rangle \leq K_i^t$
- 5: if $\langle A^r(i, 1 : i), S^{1:i} \rangle \leq K_i^r$, distribute 1's in $A^r(i, G_c(i, i+1 : l))$ greedily, giving equal priority to columns of equal weight such that $\langle A_i^r, S \rangle \leq K_i^r$
- 6: $A^r(m, i) = 1 - A^t(i, m)$ and $A^t(m, i) = 1 - A^r(i, m) \forall m \geq i+1 : (i, m) \in E_c$
- 7: end for
- 8: feasible = true if $A^t S \leq K^t$ and $A^r S \leq K^r$, else feasible = false

Figure 12: Algorithm Extended Greedy

matrix.

Both Algorithm Simple Greedy and Algorithm Extended Greedy are safe, in the sense that they always label infeasible vectors as infeasible. However, they are both non-optimal in that they each label some feasible vectors as infeasible. The accuracy of the two heuristics is evaluated in Section 6.1.2, in terms of the percentage of feasible vectors that are labeled infeasible.

6.1.2 Accuracy of Greedy Heuristics for Uniform Antenna Arrays

The scalability of the heuristics for verifying feasibility of a stream allocation vector in a MIMO network is studied experimentally by calculating the entire feasible space for values of $K^t = K^r = K = 8, 12, 16$ and network sizes up to 15 links, for a single collision domain. The results are shown in the graph of Figure 13. Note that the Extended Greedy heuristic is significantly more accurate than Simple Greedy. Extended Greedy is inaccurate at most 5% of the time with $k = 8$ and $k = 12$ and at most 10% of the time with $k = 16$, for the network sizes studied here.

Accuracy of feasibility tests for arbitrary multi-hop networks is not studied in this section. This is done in Section 7.5.2 and Section 7.5.3, where we simulate the performance of stream allocation heuristics developed in Chapter 7. These heuristics

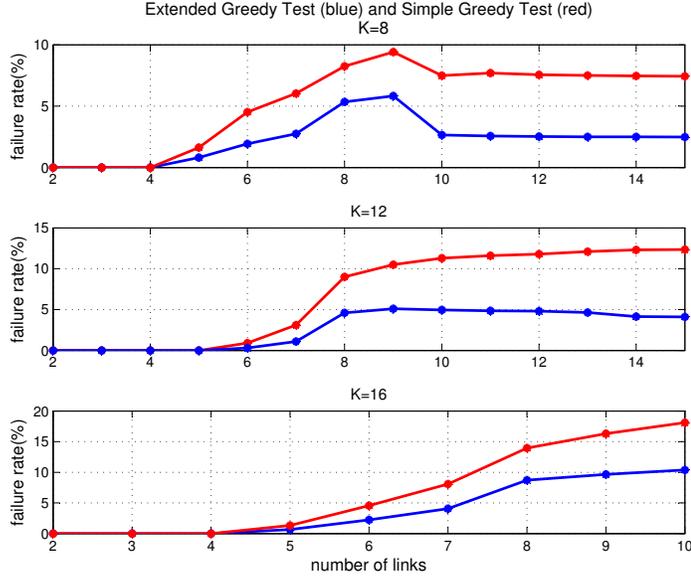


Figure 13: Failure Rates of Simple Greedy and Extended Greedy Heuristics: SCD, uniform array size

that employ the Extended Greedy feasibility test as a subroutine. This allows us to quantify the performance of the Extended Greedy test and its impact on the performance of the overall stream allocation heuristic in arbitrary multi-hop network scenarios.

6.1.3 Accuracy of Greedy Heuristics for Non-uniform Antenna Arrays

The performance of Greedy and Extended Greedy feasibility tests in the case when nodes have arbitrary antenna array sizes is experimentally studied here. Again, a single collision domain is considered. Nodes can have array sizes from anywhere between two and eight i.e. $2 \leq K^t \leq 8$ and $2 \leq K^r \leq 8$. The array sizes are distributed uniformly. SCDs up to 16 links are considered. The results are averaged over 10 array size distribution samples and plotted in Figure 14. Observe that at 9 links, Extended Greedy has a failure rate of approximately 13% which is about twice of what it is in the uniform case of subsection 6.1.2, where all antenna arrays were of size 8. The reason for this is that when array sizes are not uniform, then “giving

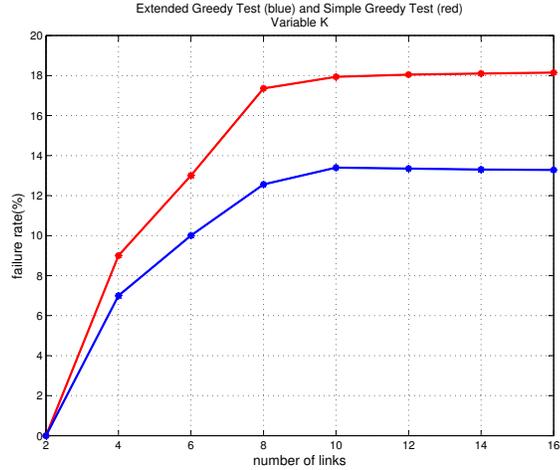


Figure 14: Failure Rates of Simple Greedy and Extended Greedy Heuristics: SCD, non-uniform array size

equal priority to columns of equal weight” in the Extended Greedy routine does not yield as much benefit as it does in the uniform case. This potentially provides an interesting opening for further research in developing higher accuracy feasibility tests for non-uniform antenna array distributions.

Yet another experiment was performed when antenna arrays can have a size of either 4 or 8 i.e. $K^t \in \{4, 8\}$ and $K^r \in \{4, 8\}$. Results for a single collision domain are shown in Figure 15. Both extended Greedy and Greedy perform very well, with the failure rate of the former being just under 2% at 12 links. Since arrays can be only either of size 4 or size 8, there is still a lot of symmetry in the system and both tests perform very well.

6.1.4 isFeasible3 Heuristic for $K \leq 3$

As a result of the problem analysis in subsection 5.3.3, we give the following method for testing for feasibility when $K = 3$, which is referred to as Algorithm isFeasible. The method is approximate and polynomial in complexity.

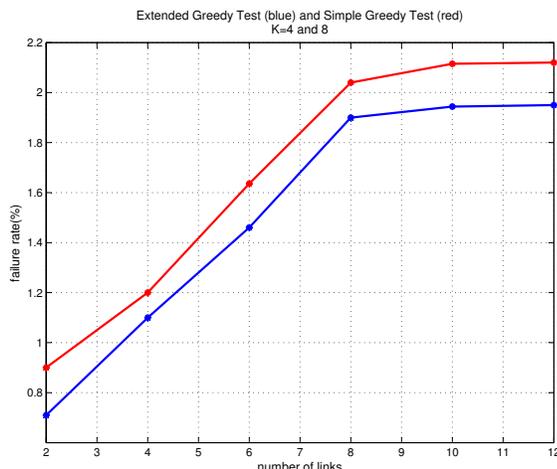


Figure 15: Failure Rates of Simple Greedy and Extended Greedy Heuristics: SCD, array size= 4 or 8

6.1.5 Accuracy of isFeasible

The scalability of the heuristics for verifying feasibility of a stream allocation vector in a MIMO network is studied experimentally by calculating the entire feasible space for values of $K_t = K_r = K = 3$ and network sizes up to 20 links. The results are shown in the graphs of Figures 17 and 18, which have logarithmic scales on the y axes. Note that the isFeasible method is significantly more accurate than the Extended Greedy heuristic. Here, we have also made a slight optimization to Extended Greedy that is specific to the $K = 3$ case. isFeasible is inaccurate at most 0.005% of the time whereas the optimized Extended Greedy is inaccurate at most 7.0% of the time for an average conflict graph degree of 5.5 at 20 links. These numbers are respectively 0.2% and 13% for average conflict graph degree of 7.5.

6.2 Conclusion

We now ask the question, whether the improvement in performance delivered by isFeasible over Extended Greedy can be exploited to develop better stream allocation

Input: Stream allocation vector $S = \{s_1, s_2 \dots s_l\}$, link set L , conflict graph $G_c = (V_c, E_c)$
Output: feasible $\in \{\text{true}, \text{false}\}$, A^t , A^r

- 1: **Repeat:** If any vertex v in V has all but p edges marked as incoming, where $p \leq 3 - s_v$, mark these p edges as outgoing.
Until: No such vertex v exists.
- 2: Remove all marked edges and marked vertices from G . Call the set of all unmarked vertices V_n .
- 3a: while there is at least one vertex v in V_n with $s_v = 2$.
- 3b: for every neighbor v' of v
- 3c: if $n_{v'} = 2$, mark v' outgoing toward v .
- 3d: if any neighbor v'' of v' has $n_{v''} = 0$ then feasible=FALSE. Exit procedure.
- 3e: else mark all neighbors v'' outwards to v' . $n_{v''} = n_{v''} - 1 \forall v''$. GOTO 3h.
- 3e: end if
- 3f: end if
- 3g: end for
- 3h: **Repeat:** If any vertex v in V_n has all but $p \leq n_v$ edges marked as incoming, mark these p edges as outgoing. $n_v = n_v - p$
Until: No such vertex v exists.
- 3i: Remove all marked edges and marked vertices from G . Call the set of all unmarked vertices V_n .
- 3j: end while
- 4: Look for a vertex v in V_n with $n_v = 2$.
- 5: Mark as outgoing a randomly chosen edge incident on v . $n_v = n_v - 1$.
- 6: **Repeat:** If any vertex v in V has all but $p \leq n_v$ edges marked as incoming, where $p = 3 - s_v$, mark these p edges as outgoing.
Until: No such vertex v exists.
- 7: Remove all marked vertices and edges. GOTO 4 and repeat through 6 until every vertex v in V_n has $n_v = 1$.
- 0: If the resulting graph $G' = (V', E')$ is a tree and/or simple cycle then feasible = TRUE and G' is directed as per the procedure used for the $K = 2$ case presented in subsection 5.3.2.
else feasible = FALSE.
end if

Figure 16: Algorithm isFeasible

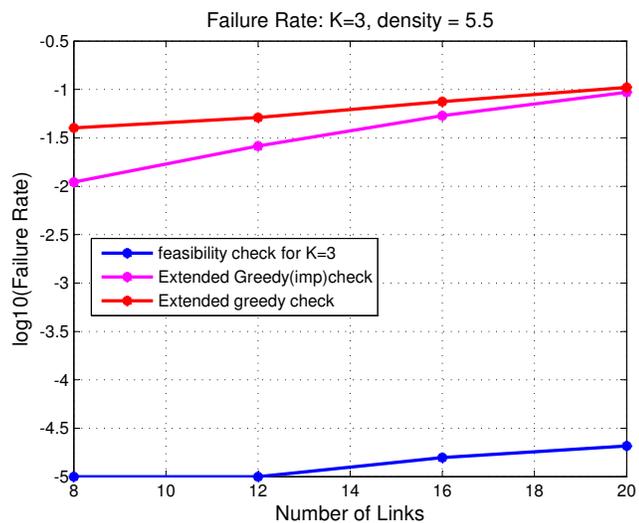


Figure 17: Failure rates of feasibility heuristics: $\text{cgd} = 5.5$

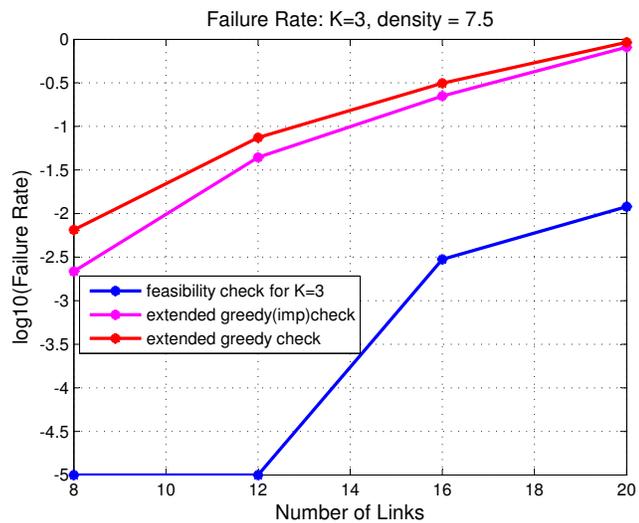


Figure 18: Failure rates of feasibility heuristics: $\text{cgd} = 7.5$

heuristics when $K = 3$. This question is answered with the affirmative in Chapter 7 (Figure 30) where we show by experimental results that stream allocation heuristics that employ *isFeasible* are superior to those that employ the Extended Greedy method, thereby delivering a higher throughput.

CHAPTER VII

STREAM ALLOCATION AND ONE-SHOT LINK SCHEDULING

The problem that we consider is that of maximizing throughput in a MIMO network while accounting for variable rate streams on MIMO links. The stream rates on a link depend on the channel conditions of the link, and the manner in which the diversity-multiplexing trade-off is handled. In this work, we use the dependence of stream rates on the channel to develop methods of link selection and stream allocation that approximately maximize the aggregate throughput. Maximizing throughput is closely tied to the problem of allocating streams based on the stream rates of the selected links.

Consider the general one-shot link scheduling problem of maximizing the aggregate throughput over an arbitrary set of links (that are not necessarily primary-interference-free). We approach the problem by splitting it into two subproblems. In the first problem (stream allocation), an algorithm determines a stream allocation vector that approximately maximizes the throughput, given a set of primary-interference-free links. The second problem considers how to select a “good” set of primary-interference-free links to provide as input to the stream allocation algorithm. When solving the overall one-shot link scheduling problem, we first run the primary-interference-free link selection algorithm, then run the stream allocation algorithm using the output of the link selection algorithm.

Optimal stream allocation/link scheduling is very complex even for networks with

15 links or less. Section 7.1 and Section 7.4 are devoted to developing efficient heuristics that approximately maximize throughput in a multi-hop MIMO network. Section 7.2 develops a one-shot link scheduling algorithm. Section 7.3 and Section 7.5 present experimental results on stream allocation in both single collision domains and in arbitrary multi-hop networks.

For all simulations, the channel is modeled as an idealized rich scattering static environment, which corresponds to a quasi-static flat Rayleigh fading channel model. Therefore, the channel has i.i.d. complex, zero mean, unit variance elements as described by [58]. The gain of each channel matrix is calculated using Friis transmission equation and the log-distance path-loss model with a path-loss exponent of 3 ([59, 70, 71, 2]). We assume channel state information is available to the transmitters and therefore include optimal power allocation in our rate calculations. The data rate is calculated from Shannon’s capacity formula using the optimal power allocation [60, 5].

In order to approximate the data rate due to the use of some DOFs for interference suppression, we first perform antenna selection and then find the optimal data rate as described above. Suppose the transmitter uses t antenna elements for transmission of s streams and that the receiver uses r antenna elements for reception. We perform best eigen-value selection by picking the t transmit elements and the r receive elements that maximize the data rate of the link. We then calculate the rates for the case of $1 \leq s \leq \min(t, r)$ streams by allocating power through the best s eigenmodes of the $t \times r$ channel.

7.1 Stream Allocation Algorithm: StreamMaxRate

A simple heuristic for the stream allocation problem, which we will use as a baseline for comparison, is to adopt a greedy approach. Streams are scheduled successively from highest to lowest rate and the allocation vector is tested for feasibility at each

step. We refer to this as the “upward construction” approach, because it simply keeps adding streams in a greedy fashion until no more streams can be added without making the allocation vector infeasible.

We now present an algorithm that provides a better approximation to the stream allocation problem (compared to the simple upward construction approach). For a given set of primary-interference-free links, the value of the stream allocation vector S is initialized so as to maximize the aggregate throughput while satisfying the following two constraints: (1) interference between every pair of links is suppressed and (2) weight of the stream allocation vector is bounded by w_0 such that

$$w_0 = \begin{cases} \lfloor \frac{2kl}{l+1} \rfloor & \text{for a single collision domain} \\ kl & \text{for a multi-hop network} \end{cases}$$

where l is the number of non-zero entries in the vector and k is the median value of the vector resulting from taking the minimum of the elements of K^t and K^r .¹ All pairwise interference constraints can be checked in polynomial time. However, since the initialization only checks pairwise interference between links, the initial vector might not be feasible. In fact, for networks that are not extremely sparse, it is highly likely that it will be infeasible. Therefore, the initial vector that is produced will probably have a high throughput that provides a minimum level of interference suppression. Pseudo-code is shown for the initialization procedure in Figure 19.

Once an initial stream allocation vector is determined, it is tested for feasibility using any feasibility checking algorithm (in the description presented herein, we assume the Extended Greedy heuristic of Section 6.1.1). If the initial vector is feasible, then it becomes the final output of the algorithm. In most cases, the initial stream allocation vector will not be feasible and the algorithm will then adjust it to try

¹If $K_i^t = K_i^r = k, \forall i$, then $\lfloor \frac{2kl}{l+1} \rfloor$ is the maximum number of streams [49].

- 1: Let $S^0 = [0, \dots, 0]_{|L|}$
- 2: repeat
- 3: add the highest rate stream not already in S^0 that maintains the following conditions $\forall i, j \in L$:
 - (1) If $G_c(i, j) = 1$, either $s_i + s_j \leq K_i^t$ and/or $s_1 + s_2 \leq K_j^r$
 - (2) If $G_c(j, i) = 1$, either $s_i + s_j \leq K_j^t$ and/or $s_1 + s_2 \leq K_i^r$
 - (3) $s_i \leq \min(K_t^i, K_r^i)$
 - (4) $s_j \leq \min(K_t^j, K_r^j)$
- 4: until $\sum_{i=1}^{|L|} s_i^0 = w_0$, such that

$$w_0 = \begin{cases} \left\lfloor \frac{2kl}{l+1} \right\rfloor & \text{for a single collision domain} \\ kl & \text{for a multi-hop network} \end{cases}$$

where $l = \text{no. of non-zero entries in } S^0$ and $k = \text{median of } \min(K^t, K^r)$

Figure 19: Initialization Procedure for Algorithm StreamMaxRate

Input: Primary-interference-free link set L , K^t , K^r , $R(t_i, r_i, ADOF_{t_i}, ADOF_{r_i})$, conflict graph $G_c = (V_c, E_c)$

Output: Feasible stream allocation vector S for L , A^t and A^r that make S feasible

- 1: Initialization: Choose S^0 to satisfy pairwise interference constraints and approximately maximize aggregate rate as detailed in Figure 19
- 2: $S = S^0$
- 3: (feasible, A^t , A^r) = ExtGr(S, L, K^t, K^r, G_c)
- 4: while not feasible
- 5: (feasible, S, A^t, A^r) = UpdateRule($S, L, K^t, K^r, A^t, A^r, G_c$)
- 6: end while

Figure 20: Algorithm StreamMaxRate

to make it feasible. This is done by removing streams from the initial vector until it becomes “more feasible” and then trying to add more streams in where possible without reducing feasibility. The stream allocation vector is adjusted by an update rule procedure, which is guaranteed to increase the number of rows that are feasible of the LHS in each of Conditions 1 and 2 by at least one. Thus, repeated calls to the update rule will eventually produce a stream allocation vector that is completely feasible. Pseudo-code is shown for the overall StreamMaxRate algorithm in Figure 34 and for the allocation vector updating procedure in Figure 21.

Input: Stream allocation vector S , link set L , K^t , K^r , A^t , A^r

Output: feasible $\in \{\text{true}, \text{false}\}$, updated stream allocation vector S , A^t , A^r

- 1: $\text{nfr}^0 = \min(\text{nfr}^t, \text{nfr}^r)$, where nfr^t and nfr^r are the number of feasible rows in A^t and the number of feasible rows in A^r , with respect to S
- 2: repeat
- 3: remove the lowest rate stream from S
- 4: until $\min(\text{nfr}^t, \text{nfr}^r) > \text{nfr}^0$
- 5: $\text{nfr}^1 = \min(\text{nfr}^t, \text{nfr}^r)$
- 6: for each stream s_i not included in S from highest rate stream to lowest rate stream
- 7: add s_i to S
- 8: (feasible, A^t , A^r) = ExtGr(S , L , K^t , K^r , G_c)
- 9: calculate nfr^t and nfr^r from A^t and A^r
- 10: if $\min(\text{nfr}^t, \text{nfr}^r) < \text{nfr}^1$ then remove s_i from S
- 11: end for

Figure 21: Update Rule for Algorithm StreamMaxRate

7.2 One-Shot Link Scheduling Algorithms

As mentioned earlier, our approach to one-shot link scheduling is to first pick a “good” set of primary-interference-free links and then apply Algorithm StreamMaxRate to optimize stream allocation among those links. Any set of links making up a matching of the communication graph is primary-interference-free and is therefore an eligible candidate for the input to Algorithm StreamMaxRate. Clearly, the optimal solution will use a set of links corresponding to some maximal matching.

We consider two different primary-interference-free link selection heuristics, based on weighted matching algorithms:

1. The weight of each link is set equal to the inverse of the physical distance between the transmitter and receiver of that link, i.e. $w_i = \frac{1}{d_i}$. Here, we find a *maximum* weighted matching using the algorithm of [73].
2. The weight of each link is set equal to the physical distance between the transmitter and receiver of that link, i.e. $w_i = d_i$. We consider maximal matchings with at least a given number of links. From among these candidates, we choose the matching with minimum total weight.

7.3 Stream Allocation and One-Shot Scheduling Results for Single Collision Domain

We have proposed the StreamMaxRate Heuristic for approximately maximizing the throughput over a given set of links in Section 7.1. In this section, we will define an experimental set up and use simulation results to compare the performance of the StreamMaxRate heuristic against the optimal throughput. Because of the relatively small number of links that can be active concurrently within a single collision domain, we are able to calculate the optimal solution for a good portion of the input parameter space considered. Additionally, we have simulated the greedy upward construction approach of finding a stream allocation and use this as a second comparison point. Finally, we show some results on the overall one-shot link scheduling problem by including the two weighted matching methods of selecting a set of primary-interference-free links. Since brute-force searching the space of all possible maximal matchings is infeasible, we only compare our approach to the greedy construction in this case.

7.3.1 Simulation Set-up

For all simulations, every node is equipped with an antenna array of size $k = 8$. This allows for a maximum of $l = 15$ links to be active concurrently, given the single collision domain assumption. The experimental set up for the stream allocation results is as follows. We distribute 50 nodes (with a uniform distribution) over a field of fixed dimensions. All nodes are within interference range of each other. We select 50 randomly generated matchings (sets of primary-interference-free links) with sizes ranging from two to fifteen. This is averaged by repeating the procedure over a sample space of node distributions. For different matching sizes, we compute (a) the optimal throughput (b) the throughput resulting from application of the StreamMaxRate heuristic and (c) the throughput obtained by applying the greedy upward construction procedure.

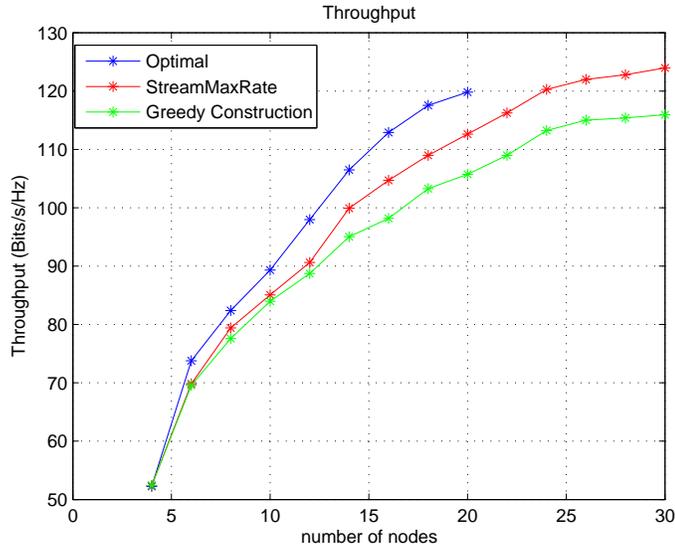


Figure 22: Throughput vs number of nodes for randomly selected matchings

The set-up for the one-shot link scheduling results is as follows. We distribute (uniformly) an even number of nodes, ranging from $N = 2$ to $N = 30$ over a field of fixed dimensions. For each value of N , we select a maximal matching (of size $N/2$, since all nodes are within transmission range of each other). We do this by the two weighted matching procedures described in Section 7.2. We combine these two matching selection procedures with the two stream allocation heuristics (StreamMaxRate and greedy) to produce four different curves. For each data point on a curve, an average was obtained by repeating the process over a large sample space of node distributions.

7.3.2 Simulation Results

Figure 22 shows the results for the stream allocation problem alone. Due to the large computation time of determining the optimal value of the throughput for larger numbers of nodes and links, the optimal result is shown only up to 20 nodes, which corresponds to 10 links. Note that at $n = 20$, the throughput from StreamMaxRate is within 7% of the optimal. The greedy upward construction approach is within 15%

of the optimal at this point. Thus, StreamMaxRate cuts the difference between the heuristic and the optimal in half at this point. Note also that the difference between the greedy heuristic and StreamMaxRate increases with network size. Extrapolating the optimal curve in a natural way would indicate that the halving of the difference from optimal produced by StreamMaxRate should continue over the range of network sizes simulated.

We now present results for one-shot link scheduling. In this case, we have the two weighted matching methods for selecting the set of primary-interference-free links and we evaluate those using both Algorithm StreamMaxRate and the greedy construction method for stream allocation. The results are shown in Figure 23. For both methods of finding matchings, StreamMaxRate retains its performance advantage compared to the greedy stream allocation heuristic (about 10-15% higher throughput for the largest network size simulated). We also find that the matching selection approach that finds the maximum weighted matching with links weighted by the inverse of their distances outperforms the one that finds the minimum weighted matching with weights equal to the distances. The difference between the two matching selection approaches is only moderate, however, being about 5% for the largest network size.

Note that, for one-shot link scheduling, we cannot compare against the overall optimal solution, since checking all maximal matchings is not feasible for the network sizes considered. However, given the result of the maximum matching heuristic, we can find the optimal allocation (as we did for the stream allocation problem results). We did this comparison and found that StreamMaxRate was within 6% of optimal for these specific matchings (essentially the same as its performance on random matchings described earlier).

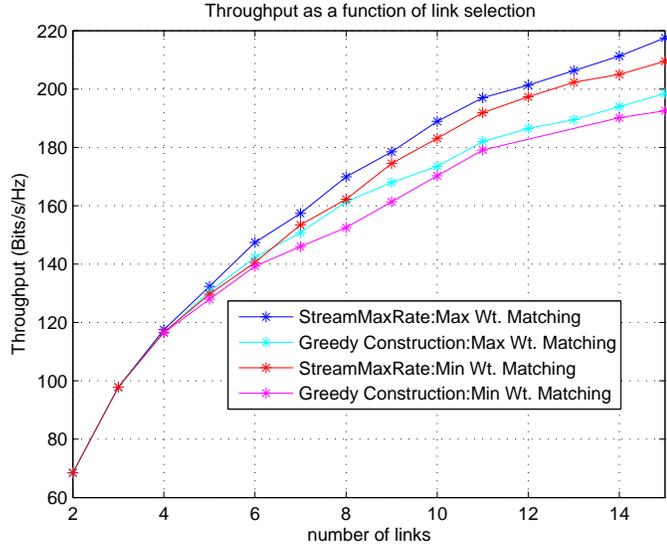


Figure 23: Throughput vs number of active links for two different methods of selecting matchings

7.4 Stream Allocation Algorithm: *MultihopMaxThrpt*

The StreamMaxRate algorithm was presented in Section 7.1 and is seen to be an overall “downwards” procedure in that a net number of streams are removed from the initial infeasible vector to produce a feasible output stream allocation vector. Since the intermediate values of the vector are infeasible, and therefore have no actual measurable throughput, the update rule can meet the throughput maximization criterion only by using an approximate estimate of vector throughput. In our implementation, throughput estimation at intermediate stages of the algorithm was done by calculating the rate on each active link under the assumption that it has at its disposal, all transmitter and receiver side resources available for multiplexing i.e. we calculated the rate of each link as if the entire transmitter and receiver antenna arrays were available.

Here, we present the MultihopMaxThrpt algorithm for stream allocation, which in contrast to StreamMaxRate is a “build-up” procedure. It initializes the value of

the stream allocation vector S to zero, adding streams to S such that at each iteration, S is incremented by that stream that if added to S , keeps S feasible and maximizes the incremental throughput. MultihopMaxThrpt therefore involves checking for feasibility (of the intermediate stream allocation vectors) and calculation of throughput (resulting from intermediate stream allocation vectors) at each iteration. Since intermediate vectors are always in the feasible space, we can calculate throughput at each iteration accurately. Note that in contrast, Algorithm StreamMaxRate requires an approximate throughput estimation since intermediate vectors might fall outside the feasible space. Finally, we contrast MultihopMaxThrpt with the simple greedy heuristic that we used as a baseline comparison against StreamMaxRate in Section 7.1. In the simple upward greedy construction, streams were scheduled successively from highest to lowest rate and the allocation vector was tested for feasibility at each step. This differs from MultihopMaxThrpt in the following respects: In the simple greedy algorithm, 1) stream rates of links were calculated assuming that all MIMO resources of each link were available for multiplexing. In other words, stream rates were only estimated and not exactly computed based on the actual number of DOFs available for multiplexing; and 2) at each iteration of MultihopMaxThrpt, the number of streams multiplexed is incremented by one for every link in turn. The update to the stream allocation vector is finally made corresponding to the highest incremental throughput achieved. The simple greedy method on the other hand simply estimates the highest rate stream at every iteration and updates the stream allocation vector accordingly.

The MultihopMaxThrpt algorithm is presented in Figure 34. At each iteration of the procedure, the feasibility check is performed by a call to the function *isFeasible* and the throughput calculation is done by calling the function *VariableRateThrpt*. Four variants of MultihopMaxThrpt potentially result depending on whether *isFeasible* checks for feasibility optimally or sub-optimally and whether *VariableRateThrpt*

calculates throughput resulting from a stream allocation vector optimally or sub-optimally. These variants are distinguished by correspondingly appending to Multi-hopMaxThrpt one of the four following subscripts: a) OptF-OptT b) OptF-SubOptT c)SubOptF-OptT d) SubOptF-SubOptT where (Sub-)OptF indicates that *isFeasible* is implemented (sub-)optimally and (Sub-)OptT indicates that *VariableRateThrpt* is implemented (sub-)optimally.

The optimal implementation of *isFeasible* ensures that whenever a vector is feasible it is successfully identified as being feasible. Moreover, vectors that are not feasible are always marked as being infeasible. For any given vector, the procedure constructs DOF assignments until one is found that supports the vector (all interference is suppressed) or until no further DOF assignments can be constructed. For vector lengths greater than 28 checking for feasibility optimally becomes computationally very expensive.

The sub-optimal version of *isFeasible* on the other hand can be implemented by running any heuristic method for checking feasibility. In our experiments, we employ the Extended Greedy method presented in Section 6.1.1. The pseudo-code for this algorithm was presented in Figure 12. This heuristic is safe, in the sense that it always labels infeasible vectors as infeasible. However, it is non-optimal in that it labels some feasible vectors as infeasible. In subsection 6.1.4, we developed a better feasibility test for the special case when all nodes have an antenna array size $K = 3$.

The optimal implementation of *VariableRateThrpt* finds the absolute maximum throughput deliverable by any given stream allocation. This is done by constructing all DOF assignments which support the given stream allocation and for each of these, computing the corresponding throughput as the sum of $R(t_i, r_i, ADOF_{t_i}, ADOF_{r_i})$ over all links i .

The sub-optimal version of *VariableRateThrpt* is implemented by calculating the throughput of a stream allocation vector resulting from a single DOF assignment that

Input: Primary-interference-free link set L , K^t , K^r , $R(t_i, r_i, ADOF_{t_i}, ADOF_{r_i})$ conflict graph $G_c = (V_c, E_c)$

Output: Feasible stream allocation vector S for L , throughput MaxThroughput

```

1: feasible = 1; MaxThroughput = 0
2: WHILE feasible
3:    $i_o = -1$ , feasible = 0
4:   FOR each link  $l_i$  in  $L$ 
5:     increment  $s_i$  by one.
6:     (retval,  $A^t$ ,  $A^r$ ) = isFeasible( $S, L, K^t, K^r$ )
7:     IF retval = 1 then
7:       CurrentThrpt = VariableRateThrpt( $S, L, K^t, K^r$ )
7:       IF CurrentThrpt > MaxThroughput then
7:         MaxThroughput = CurrentThrpt;  $i_o = i$ 
8:   feasible=feasible | retval
9:   decrement  $s_i$  by one.
10: end FOR
11: IF  $i_o \neq -1$  then Increment the number of streams on the chosen
    link  $l_{i_o}$  by one i.e.  $s_{i_o} = s_{i_o} + 1$ 
12: ELSE break out of WHILE loop.
13: end WHILE

```

Figure 24: Algorithm MultihopMaxThrpt

supports the vector. In the case of the variant SubOptF-SubOptT this is the DOF assignment resulting from *Extended Greedy*.

Of the four variants of MultihopMaxThrpt, the OptF-OptT is the most computationally expensive as both feasibility checking and throughput calculation are done optimally at each iteration of the algorithm. The SubOptF-SubOptT version is computationally the least expensive.

7.5 Stream Allocation Results For Multi-hop Networks

7.5.1 Simulation Set-up

For all simulations, every node has equal transmission range and an interference range equal to 1.5 times the transmission range. The experimental set up for the stream allocation results is as follows. We distribute a fixed number of nodes (with a uniform distribution) over a field of fixed dimensions. From the link set induced by the given transmission range we generate 50 matchings each, of sizes ranging from 8

to 36. The matchings are selected by using a maximum-weighted-matching algorithm with the links weighted by the inverse of the link lengths. Moreover, the average density of the corresponding conflict graphs is maintained constant across the range of matching sizes. This is done by controlling the area of the field over which links are selected i.e. as the size of the matching grows, the area of the field within which links can be selected also grows. The transmission range and interference range are left as parameters that are tuned to generate conflict graphs of different densities. Increasing the transmission range (while keeping the field area and node distribution constant) results in an increase in the average vertex-degree of the resulting conflict graph which we denote by “cgd”. We study graphs with cgd equal to 2.5, 5.5 and 7.5. Furthermore all nodes are equipped with an antenna array of size K . Simulations were done with $K = 4$ and $K = 6$. Since the results for these two cases were quite similar, only the $K = 4$ results are presented herein.

7.5.2 Simulation Results: Uniform Array Size

The results plotted in Figure 25 compare three different variants of MultihopMaxThrpt with one another and with the optimal throughput for conflict graphs with a cgd of 2.5 and for nodes that all have an array size of $K = 4$. Also plotted is the performance of Algorithm StreamMaxRate (see Section 7.1), which we have adapted here to work in multi-hop networks. The SubOptF-SubOptT variant and StreamMaxRate both use the Extended Greedy feasibility check. Due to high computational complexity, the optimal and MultihopMaxThrpt-OptF-OptT results are shown only up to 24 and 28 links respectively. The OptF-SubOptT involves a lower computational time and is plotted up to 32 links. Finally, the SubOptF-SubOptT and StreamMaxRate methods are simulated up to 36 links. We see that OptF-OptT is 3.0% below optimal at 24 links. Furthermore, the OptF-OptT, OptF-SubOptT and SubOptF-SubOptT variants all scale similarly with network size. OptF-SubOptT

and SubOptF-SubOptT are respectively only 1.1% and 1.7% below OptF-OptT at a network size of 28. StreamMaxRate performance is slightly worse than SubOptF-SuboptT.

The experiment is repeated for conflict graphs with cgd equal to 5.5 (generated by increasing the transmission and interference range) and the results are plotted in Figure 26. MultihopMaxThrpt-OptF-OptT is only about 2.2% below optimal at 24 links. Also, OptF-SubOptT is only about 2.3% lower than the OptF-OptT variant for a network size of 28. We see that the OptF-OptT and OptF-SubOptT variants scale similarly with network size. The performance of SubOptF-SubOptT, on the other hand, is good for small networks but starts to drop off around a network size of 28, where it is 10.5% below OptF-SubOptT and then going down to 19% below OptF-SubOptT at 32 links. The SubOptF-SubOptT therefore seems to deteriorate in performance with network size for conflict graphs with higher cgd . The StreamMaxRate curve shows the same trend as well, but again is slightly worse in performance compared to SubOptF-SubOptT. The reason for this, as explained in Section 7.1 is because an estimated value of throughput is used at every iteration of StreamMaxRate, whereas exact values are calculated at each step of SubOptF-SubOptT.

Figure 27 shows results for cgd equal to 7.5 and $K = 4$. The same patterns as in Figure 26 are seen. This indicates that at higher graph densities, the loss in performance of MultihopMaxThrpt due to a sub-optimal check for feasibility by *isFeasible* is much greater than the loss of performance due to sub-optimal throughput calculation by *VariableRateThrpt*. Note that the Extended Greedy method is used by both SubOptF-SubOptT and StreamMaxRate as the feasibility test. We see from Figures 25–27 that the performances of these two heuristics are much closer to optimal for conflict graphs with lower cgd than those with higher cgd . This can be explained by looking at the performance of Extended Greedy as a function of network density.

Figure 17 and Figure 18 show the failure rate of Extended Greedy at average

conflict graph degrees of 5.5 and 7.5, respectively. For any given network size, we see that the failure rate is higher when the cgd is 7.5 compared to that when it is 5.5. This drop in performance with increasing network density, of the feasibility testing heuristic Extended Greedy, is the likely explanation for the drop in performance with increasing network density, of heuristics like SubOptF-SubOptT and StreamMaxRate, which employ the Extended Greedy method.

7.5.3 Simulation Results: Non-uniform Array Size

Here, the performance of the stream allocation heuristics are simulated for the case of non-uniform array sizes. Nodes in the network are equipped with antenna arrays of varying sizes anywhere in the range from two to six i.e. $2 \leq K \leq 6$. The array sizes are distributed uniformly. The communication graph is generated as above by selecting maximum-weighted-matching algorithm with the links weighted by the inverse of the link lengths. Note that the maximum number of streams that a link with transmitter size equal to K_t and receiver size equal to K_r is $\min(k_t, K_r)$. The optimal throughput, MultihopMaxThrpt-OptF-OptT, OptF-SubOptT, SubOptF-SubOptT and SMR results are plotted in Figure 28 and Figure 29 for cgd values equal to 2.5 and 5.5 respectively. In Figure 28 we see that SubOptF-SubOptT is 12.5% below OptF-SubOptT for 32 links (and that SMR is a little poorer than SubOptF-SubOptT). For the same value of cgd= 2.5 in the uniform array case, the difference between OptF-SubOptT and SubOptF-SubOptT was only 1.5% for 32 links (see Figure 25). The reason for the poorer performance of SubOptF-SubOptT in the non-uniform case is because of the degradation of performance of Extended greedy for non-uniform network scenarios that we saw in subsection 6.1.3. In Figure 29 the same patterns as in Figures 26, 27 are seen, only they are even more pronounced. That is, the performance of SubOptF-SubOptT and SMR falls off even more sharply - due to the combined effect of a higher cgd and non-uniform array sizes.

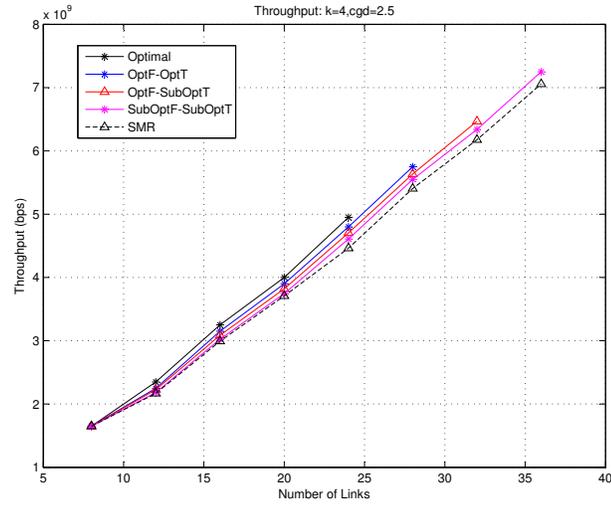


Figure 25: Throughput vs no. of nodes for randomly selected matchings: $cgd = 2.5$

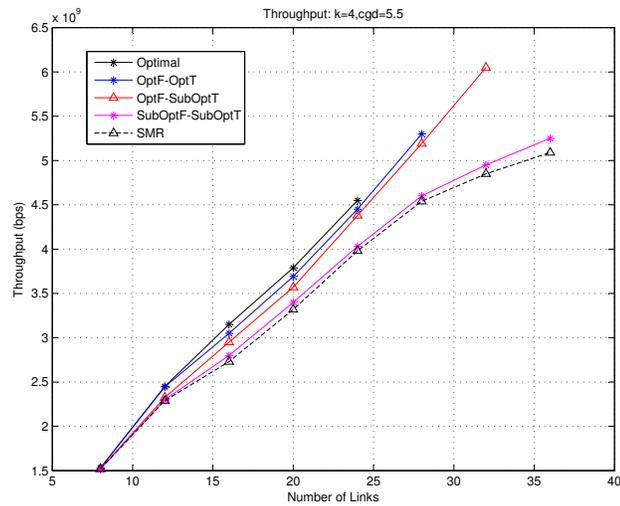


Figure 26: Throughput vs no. of nodes for randomly selected matchings: $cgd = 5.5$

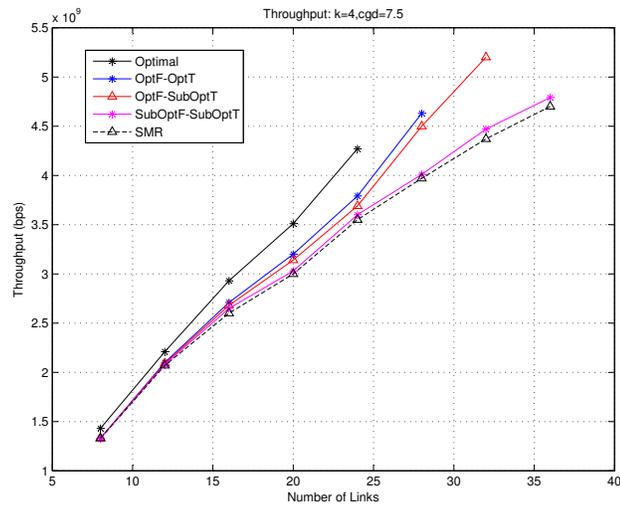


Figure 27: Throughput vs no. of nodes for randomly selected matchings: $cgd = 7.5$

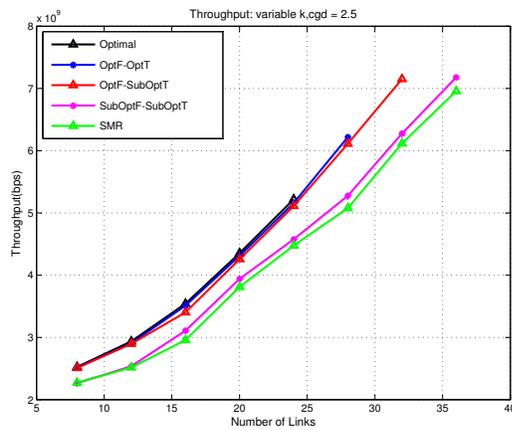


Figure 28: Throughput vs number of nodes for randomly selected matchings: $cgd = 2.5$; Variable K case: K values can be anywhere between 2 and 6

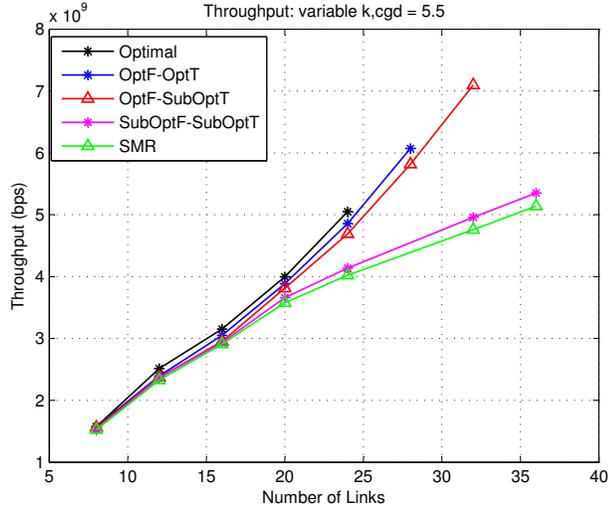


Figure 29: Throughput vs number of nodes for randomly selected matchings: $\text{cgd} = 5.5$; Variable K case: K values can be anywhere between 2 and 6

This is a direct motivation for developing more accurate feasibility testing methods. Clearly, Extended Greedy does not scale very well at higher conflict graph degrees. Note that we developed a highly accurate feasibility checking heuristic, that is far more accurate than Extended Greedy, for the special case when every node has $K = 3$ antennas in Section 6.1.4. Developing feasibility heuristics with good accuracy that scale well with network density remains an important open problem for $K > 3$.

The running times of optimal stream allocation calculation, Algorithm StreamMaxRate, and the three MultihopMaxThrpt variants are shown in Table 7.5.3. Running times are shown per sample, for the case when the cgd is equal to 5.5. As expected, OptF-OptT has the highest running time of all the heuristics compared as it performs both feasibility checking and throughput calculation operations optimally at every iteration. OptF-SubOptT is next in computational load as it performs feasibility checking optimally but computes the throughput sub-optimally thus making it faster. SubOptF-SubOptT and StreamMaxRate are both very efficient in terms of running time.

Table 4: Average running times (per sample) for 24 links

optimal calculation	2059 min
OptF-OptT	819 min
OptF-SubOptT	598 min
SubOptF-SubOptT	3 min
StreamMaxRate	0.55 min

Finally, we answer the question, whether the improvement in performance delivered by *isFeasible* over Extended Greedy can be exploited to develop better stream allocation heuristics when $K = 3$. To answer this, we use the same simulation set-up as in Subsection 7.5.1, set $K = 3$, and focus on an average conflict graph degree of 5.5. For this scenario, we evaluated MultihopMaxThrpt-*SubOptF-SubOptT* using both the *isFeasible* feasibility test and Extended Greedy. For comparison, we also evaluated the OptF-SuboptT variant, which uses a perfect feasibility test, and the exact optimal algorithm. The results are plotted versus network size in Figure 30. We can see that the new feasibility heuristic does improve the stream allocation al-

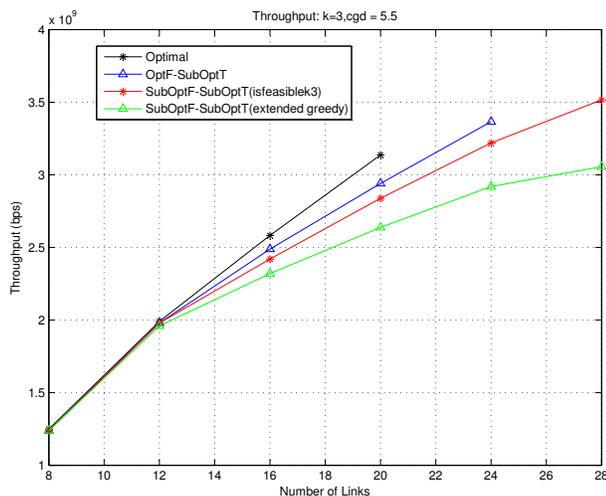


Figure 30: Throughput vs no. of nodes for randomly selected matchings: $K=3$, $cgd=5.5$

gorithm substantially compared to Extended Greedy. For 24 links, the throughput

with the new feasibility heuristic is about 13% higher than the throughput with Extended Greedy, and it is only about 3% from the throughput achieved with a perfect feasibility evaluation. For 28 links, the throughput with the new heuristic is about 15% higher than with Extended Greedy. These results show that the better accuracy of the specialized $K = 3$ feasibility heuristic *does* translate into substantially better performance for the stream allocation algorithm and, for the range of network sizes evaluated, it is close to what can be achieved with a perfect feasibility evaluator.

CHAPTER VIII

PARTIAL RESULTS AND FUTURE DIRECTIONS

8.1 Stream Allocation: Independent Connected Component Partitioning

We saw in Figure 26 and Figure 27 of Section 7.5, that the performance of SMR falls 16% below ‘optimal’ for network sizes exceeding 28 links. An attempt is made here to develop a stream allocation method which improves upon the performance of the StreamMaxRate algorithm for larger network sizes. We were successful in achieving a performance improvement for grid-shaped networks. However, algorithmic parameters needed to be carefully chosen even for these grids. Further experimentation and analysis is necessary before the method can be made reliable for arbitrary network topologies. As far as the running time of the method is concerned, it is clearly lower than its SMR counterpart. Our results show some merits of the method which indicate that it might be possible to refine the method so as to increase its utility.

The proposed stream allocation heuristic is called ‘PartitionedStreamMaxRate’. The conflict graph of the multihop MIMO network is first partitioned into a carefully chosen number of independent connected components. The connected components are independent in the sense that no edges are present between pairs of components. Once the partitioning is done, the StreamMaxRate procedure is applied over each connected component independently. We have studied cases where the number of partitions N varies from one to four. The running time of the heuristic is seen to decrease with increase in N . For grids of size equal to 80 links and more, well chosen input parameters delivers a performance improvement of as much as 10% – 15% over SMR. Note that when $N = 1$, PartitionedSMR simply reduces to SMR.

8.1.1 PartitionedStreamMaxRate Algorithm

For a given set of primary-interference free links, the corresponding conflict graph $G_c = (V_c, E_c)$ is computed. This graph will be partitioned into N independent components. Suitable values for N are studied through simulation in sub-section 8.1.3. The value of N is chosen such that the size of the components are small enough to ensure a low running time but such that too many vertices are not removed by the partitioning. The N components are initialized to contain a single vertex each, belonging to an appropriately chosen maximal independent set (IS) of V_c of size N . This starting independent set is chosen so as to approximately maximize the pairwise hop distance between its vertices. The reason for doing so is to separate the components from one other as much as possible. The components are grown iteratively such that they continue to remain independent of one another. The idea is to keep the size of the components as balanced as possible. Maintaining independence requires removing neighbors that are common to two or more components from G_c at every iteration. Maintaining connectivity requires augmenting each component with one or more of its neighbor vertices. Denote by ‘*Neighbors*’, the set of vertices at any iteration of the algorithm that are not contained in any of the components and that do not have neighbors in more than one component. This set constitutes the possible candidates which can be used to grow the components. At each iteration an independent set of vertices will be select from *Neighbors* to be used to augment the components. Let $G_n = (V_n, E_n)$ be the subgraph induced by *Neighbors* in G_c . Remove from E_n edges between vertices which are neighbors of the same component. Enumerate all maximal independent sets of G_n . Augment the connected components with each of these in turn. Apply StreamMaxRate to each augmented component and compute its throughput. The maximal independent set that maximizes the sum total throughput over the N components is chosen as the augmenting set. The iterative procedure ends when all vertices in V_c are checked i.e either (1) removed on account of

- Input:* conflict graph $G_c = (V_c, E_c)$, number of partitions N , link set L, K^t, K^r , Rates $R(t_i, r_i, ADOF_{t_i}, ADOF_{r_i})$ conflict graph $G_c = (V_c, E_c)$
- Output:* independent partitions $G_i = (V_i, E_i) \forall i = 1 \dots N$, Root, stream allocation vector S
- 1: find an independent set, $Root \in V_c$, of G_c of size N that approximately maximizes the sum total pairwise hop distance between the vertices in Root. $Root = \text{findRoot}(G_c, N)$.
 - 2: Initialize the N connected components as: $V_i = \text{Root}(i) \forall i = 1 \dots N$
 - 3: The vector \vec{Trpt} of the throughput of each components is initialized as $\vec{Trpt} = \text{Rates}(\text{Root})$
 - 4: Number of unchecked vertices $N_v = |V_c|$
 - 5: repeat
 - 6: $Neighbors$ = set of all vertices adjacent to a connected component.
 - 7: remove vertices that are common neighbors to two or more components from $Neighbors$.
 - 8: $G_n = (V_n, E_n)$ is the subgraph induced by $Neighbors$ in G_c .
 - 9: Remove from E_n , the edges between neighbors of the same connected component and from N_v .
 - 10: Consider all maximal independent sets IS_n^j of G_n , $j = 1 \dots \text{MaxNum}$. (MaxNum = no. of such sets)
 - 11: for $j = 1 \rightarrow \text{MaxNum}$
 - 12: Add each vertex in IS_n^j to the corresponding connected component $G_i = (V_i, E_i)$ to produce $G'_i = (V'_i, E'_i)$.
 - 13: Compute the approximate maximum throughput of each component. $Trpt_i = \text{StreamMaxRate}(G'_i = (V'_i, E'_i), K^t, K^r, R(t_i, r_i, ADOF_{t_i}, ADOF_{r_i})) \forall i = 1 \dots N$.
 - 14: $TotalTrpt_j = \sum \{Trpt\}$ is the sum total throughput of the N components.
 - 15: end for
 - 16: $j_0 = \arg \max TotalTrpt_j$
 - 17: Augment the connected components $G_i = (V_i, E_i)$ with the vertices in $IS_n^{j_0}$
 - 18: Remove V_n from N_v .
 - 19: until $N_v = 0$

Figure 31: PartitionedStreamMaxRate Algorithm

being a common neighbor (2) added to a component. Finally vertices that have been removed by the above procedure are scheduled by applying SMR over the subgraph of G induced by these, while keeping stream allocation and DOF assignment over all other vertices unchanged. This is illustrated in Figure 32 and Figure 33.

8.1.2 Simulation Set-Up

In this section, we will define an experimental set up and use simulation results to compare the performance of the PartitionedStreamMaxRate heuristic against SMR.

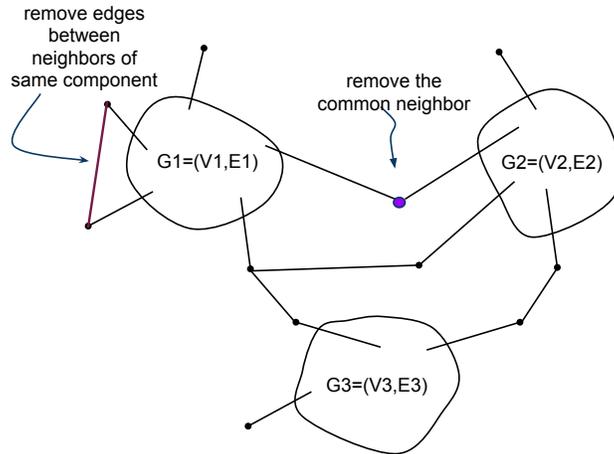


Figure 32: PartitionedSMR: neighbors of connected components identified and common neighbors removed.

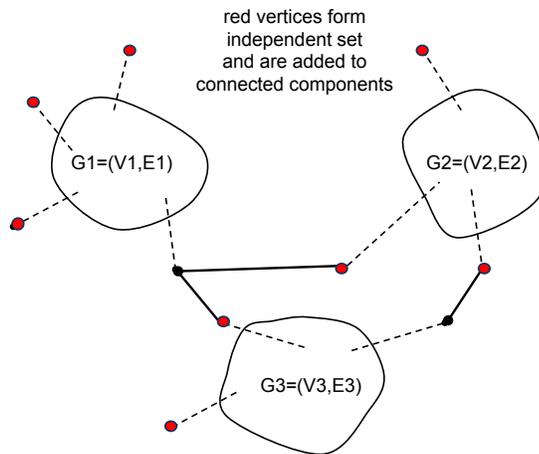


Figure 33: PartitionedSMR: maximal IS of neighbors selected to augment partitions

- 1: *Input:* conflict graph $G_c = (V_c, E_c)$, number of partitions N .
- 2: *Output:* independent set $Root$ of size N
- 3: Consider $N_{is} \leq \frac{|V_c|}{2}$ candidate independent sets each of size greater than N .
- 4: $Root = 0$
- 5: for each candidate independent set IS do the following.
- 6: $R = 0$. Add an arbitrary vertex in IS to R .
- 7: Greedily add vertices from IS such that the minimum hop distance of the vertex from $Root$ is maximized.
- 8: Continue until size of $Root$ is equal to N .
- 9: If R has a cumulative minimum distance greater than $Root$, then set $Root = R$.
- 10: end for

Figure 34: findRoot procedure

For all simulations, every node is equipped with an antenna array of size $K = 8$. We consider a rectangular grid of size 20×10 . Primary-interference-free links are selected by picking different matchings of varying sizes. The transmission range and interference range are defined in terms of hop intervals. We sweep the transmission range (TR) over three hop intervals and the interference range (IR) over nine hop intervals. If two nodes are within transmission range of one another, then they can potentially form a link in the matching. If two nodes belonging to different links are within interference range of one another, then the conflict graph will have an edge (directed) between the two links. Tuning the transmission range controls the number of links of different qualities that can be potentially included in a matching. Tuning the interference range controls the density of the conflict graph. We pre-picked three different matchings by varying the TR. For each matching, the interference range is tuned in steps of three over a nine hop interval. Every choice of matching and interference range results in a communication graph with links of fixed qualities and a conflict graph of a certain density. PartitionedStreamMaxRate is applied over each of these, with N as a parameter. Matching sizes from 50 to 100 are considered.

8.1.3 Simulation Results

Experiments are done on grid-based communication graphs. The type 4 grid is shown in Figure 35.

Figure 36 shows results for a type 4 grid network with 70 links, $IR = 4$ hop intervals and $TR = 2$ hop intervals. The partitionedSMR is applied as described in Figure 31 but instead of calling the *findRoot* method to initialize the starting independent set, we selected this set manually for each value of N . Through a careful selection of the starting independent set, partitioned-SMR is seen to increase the achievable throughput when the number of partitions increases from $N = 1$ (SMR) to $N = 3$ while simultaneously reducing the computational load. (The point $N = 0$

Type 4 Grid

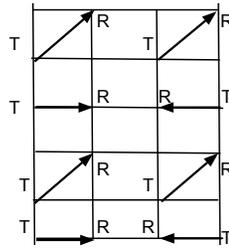


Figure 35: Type 4 grid: links of length equal to one hop length and two hop lengths corresponds to scheduling the links belonging to a maximal independent set.) The experiment is repeated for a higher interference range of $IR=5$ in Figure 37 and the same trends are seen here.

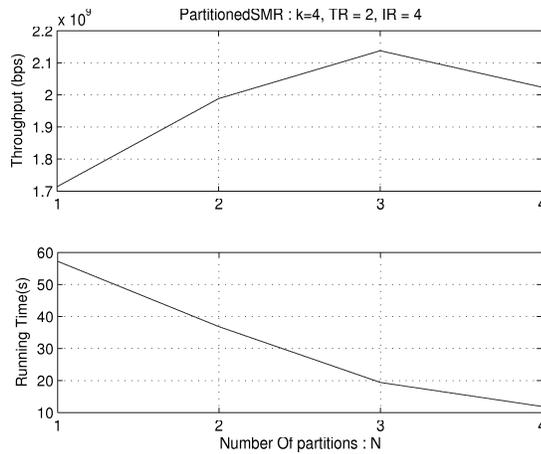


Figure 36: Throughput and running time from PartitionedSMR for type 4 grid as a function of number of partitions

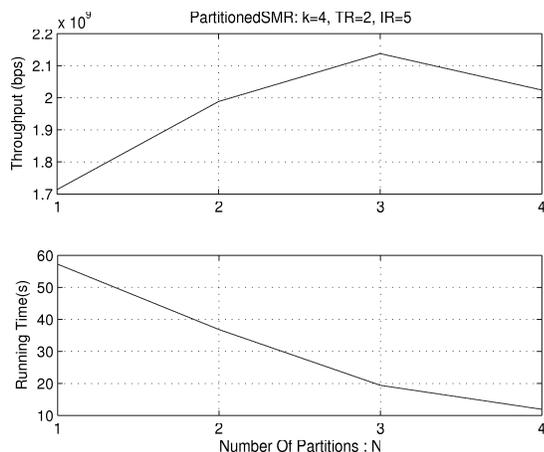


Figure 37: Throughput and running time from PartitionedSMR for type 4 grid as a function of the number of partitions

8.1.4 Drawbacks Of PartitionedStreamMaxRate

A crucial factor governing the performance of PartitionedSMR is the selection of the starting independent set from which the connected components are grown. We observed earlier that it is desirable to keep the sizes of the components as balanced as possible. Our results showed that partitioning delivered an improvement in throughput (over SMR) only when the components had more or less equal sizes. One obvious requirement to achieve this is that the vertices of the starting set should be far apart from one another as possible. The method *findRoot* achieves this condition. However, it appears that this condition is not sufficient to ensure balanced component sizes. As an example Figure 38 shows the throughput and running time as a function of the number of partitions for a type 4 grid with $IR = 4$. It is the same set up as that used in Figure 36 except that here, the method *findRoot* is used to determine the starting sets (rather than choosing them manually as done before). The independent components created are not well balanced and no performance improvement in throughput is seen when partitioning is applied. The throughput falls slightly (approx 5%) when N increases from one to four. Therefore a method of determining the

starting set needs to be developed such that well balanced components result. When this is done, PartitionedSMR can be tested on random network topologies. This is a subject for future work and has not been studied further in this thesis.

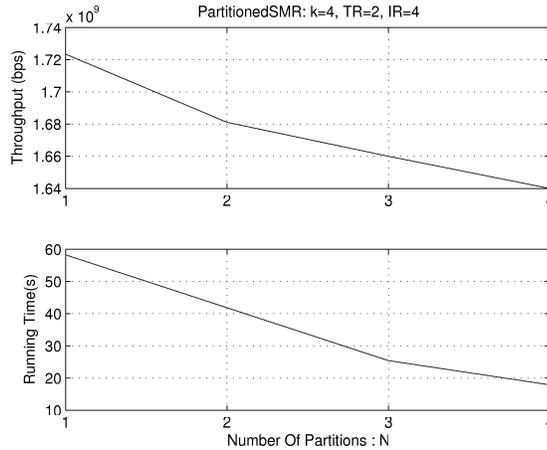


Figure 38: Throughput and running time from PartitionedSMR for type 4 grid as a function of the number of partitions

8.2 Complexity of Verifying Feasibility for Arbitrary MIMO Networks

In Chapter 5, the complexity of verifying feasibility was studied for several specific MIMO scenarios. It was shown that checking feasibility of a stream allocation vector in an arbitrary MIMO network with only receiver-side interference suppression is polynomial in complexity. It was also shown that in the case of a multi-hop MIMO network where every node has at most $K = 2$ antennas, the feasibility of an allocation vector can be checked in polynomial time. Next, the case when every node in the network has $K = 3$ antennas was studied. The condition $D1$, stated in the theorem below was established as necessary and sufficient for a vector to be feasible over such a network. However, the complexity of verifying whether this condition holds still remains an open problem.

Theorem 2 *Let $D1$ be the property of a graph $G = (V, E)$, whereby every vertex induced subgraph of G has an average degree at most equal to four. $D1$ is necessary and sufficient for the edges of G to be directed such that every vertex has at most two outgoing edges.*

8.2.1 Open Problem

The following well defined problem remains open.

Open Problem P1: What is the complexity of determining whether the edges of graph G can be directed such that every vertex has at most two outgoing edges?

In other words, the question to be answered is - What is the complexity of verifying condition $D1$?

From our analyses of this problem, we conjecture that the complexity might be linked with the well known NP complete clique-problem. Problem P1 is a well defined graph problem and lends itself easily to mathematical analysis. Therefore, trying to answer this question as opposed to the very general question - “what is the complexity of verifying feasibility over an arbitrary MIMO network” - gives us a better starting point to tackle this problem. We believe that mapping the general question of the complexity of MIMO feasibility to this specific graph problem is an important step in terms of enabling future study of this subject.

8.2.2 Efficient Feasibility Check for $K = 4$

A very efficient and high performance feasibility check when $K = 3$ was developed in Chapter 6. This check proceeds by construction and performs so well because it is specifically tailored to the $K = 3$ case. This makes it seem very promising that a similar constructive procedure tailored to the specific case of $K = 4$ can be developed and which would probably be superior to generic feasibility tests like the Extended Greedy check.

8.3 Integrating Physical Layer Operations with the Networking Layer

Our laboratory is currently active in the design of a distributed, network-wide interference handling technique which involves computing MMSE-based weight vectors to be used at transmit and receive MIMO arrays to handle interference. The procedure is iterative, with each transmitter (receiver) weight vector being dependent on the weight vector of all the other transmitters and receivers in the network.

It was seen that if a stream allocation vector was determined to be feasible (according to the full degrees-of-freedom model) then a set of weight vectors for all nodes of the network does exist such that all interference can be successfully removed. The throughput estimated by this approach however deviates from the throughput that we calculated based on the variable-rate-DOF model, the former being much more realistic/accurate than the latter.

Clearly, given the fundamental nature of feasibility, it is clear that integrating interference handling techniques with procedures for feasibility checking and stream allocation is a very important step in obtaining a complete solution to the network throughput optimization problem. In simple words, firstly, a feasible stream allocation vector could be determined by any of the methods we have developed in this thesis - MultihopMaxThrpt, SMR etc (based on the full DOF model). Secondly, the given stream allocation vector is realized in practice (in the PHY layer) by applying the interference management technique discussed above.

8.4 Practical Applications of this Work

The work presented in this dissertation would find practical application in outdoor mesh networks if our procedures could be implemented in a distributed manner. One approach would be to follow the distributed method that the authors of [44] developed

to implement their stream controlled medium access (SCMA) protocol¹. However, our stream allocation heuristics require the distributed implementation to perform several fundamental functions that are not a part of the approach in [44]. Firstly, since in our framework we assume a CL-MIMO system, channel state information has to be available to transmitters. The approach identifies cliques in the conflict graph and performs stream allocation within each clique. Feedback of CSI from *every* receiver to *every* transmitter within every clique in the conflict graph, must take place regularly at appropriate time intervals in accordance with the channel dynamics. This would require the distributed implementation to operate within a TDMA framework (unlike in [44] where a contention based framework suffices). Secondly, since interference cancellation is being done both by transmitters and by receivers, a DOF assignment must be realized which allocates the interfering receivers at which a transmitter nulls itself. Once all transmitters have performed nulling operations, each receiver will decode its intended signal by suppressing the remaining interference with zero-forcing or MMSE. The components of the distributed approach are listed here.

1. Coloring - At system start up, every transmitter transmits one stream. Each receiver observes a certain number of streams, either less than or equal to K or greater than K . The roles of transmitters and receivers are then reversed for one time slot and all receiver nodes transmit one stream. Each transmitter observes a certain number of streams, either less than or equal to K or greater than K . Depending on the numbers of streams observed by the transmitter and receiver of a link, the link colors itself as white if an average of less than or equal to K streams is observed, else it colors itself as red (bottle neck link).
2. Channel Access - White and red link sets are scheduled in alternate time slots as per the TDMA scheme.

¹This is a novel MAC protocol for ad hoc MIMO networks that enables the unique characteristics of MIMO to be coupled with optimization considerations.

3. Co-ordinated Scheduling - White links within interference range, by definition are able to transmit simultaneously by multiplexing an appropriately reduced ($< K$) number of streams and performing interference cancellation with remaining DOFs. Red links on the other hand transmit at the full rate (K streams) when the white links are inactive (these links do not interfere with one another and therefore do not need to perform IC).
4. Stream Allocation - As mentioned earlier, CSI must be fed back from every receiver to every transmitter within every clique at regular intervals, made possible by the TDMA framework within which we implement our scheme. At designated time slots, no data transmissions are made. Only CSI is exchanged between nodes. Along with this exchange, we propose to exchange node IDs (IP addresses for example) as well. Therefore, 1) every transmitter (receiver) knows the channel between itself and every other receiver (transmitter) and 2) every transmitter knows every other receiver's ID.

With this information available to the nodes, we can implement the DOF assignment for cliques that was developed in our work [54]. For a clique of size equal to l , the maximum number of streams that can be scheduled is $\lfloor \frac{2Kl}{l+1} \rfloor$. At the transmitters, all interfering receiver node IDs are ordered in a pre-determined fashion. Suppose (for the purpose of illustration), that we label these receiver IDs as $\{1, 2 \dots l\}$. The transmitter whose receiver has an ID equal to i then nulls itself at the $\lfloor \frac{l-1}{2} \rfloor$ receivers with IDs ranging from $mod(i+1, l)$ to $mod(i + \lfloor \frac{l-1}{2} \rfloor, l)$. Once this is done, the receivers will have sufficient MIMO resources to decode the desired signal and suppress remaining interference with a linear technique such as ZF or MMSE.

CHAPTER IX

CONCLUSION

Contributions made by this thesis to the subject of MIMO networking are

1. We proposed the *variable rate* model for a MIMO link, which accounts for diversity gain and the law of diminishing returns of spatial multiplexing. This overcomes a major limitation of the degrees-of-freedom model which does not account these factors and which models stream rates of links uniformly. This is called the *uniform rate* model and is widely used by researchers within the DOF framework. For the test case of a single collision domain of size 10 links, we achieved approximately a 20% higher network throughput by applying the variable rate model as compared to what was achieved using the uniform-rate model. (Chapter 1)
2. We cast the feasibility problem as a matrix formulation. This mathematical framework was the basis for (a) our subsequent theoretical results on feasibility; (b) the design of heuristics for feasibility tests and (c) the design of stream allocation algorithms for optimizing MIMO network throughput. We see that feasibility is a variation of a Boolean satisfiability. (Chapter 5)
3. We showed that in a single collision domain (SCD) with l links and where every node has an array size equal to k , the optimal throughput is equal to $\lfloor \frac{2kl}{l+1} \rfloor$. This was done using the Lagrange multiplier method of optimization. We note that the maximum size of such an SCD is $2k - 1$ links. (Chapter 5)
4. We showed that checking feasibility when only receiver-side interference suppression is done is of polynomial-time complexity.

5. We showed that for arbitrary multi-hop networks, if the size of every antenna array is no greater than two, then the complexity of checking feasibility is polynomial.
6. We derived a necessary and sufficient condition D_1 (see subsection 5.3.3) for verifying feasibility in an arbitrary symmetric multi-hop network where the size of every antenna array is no greater than three and every link carries one stream. Although the complexity of verifying D_1 remains an open problem, our investigations lead us to conjecture that this problem is related to the known NP-complete clique-problem. This is an important open problem as the problem of determining the complexity of verifying D_1 lends itself to mathematical analysis easier than the general problem of determining the complexity of verifying feasibility. (Chapter 5)
7. We developed two efficient heuristic feasibility tests: Extended Greedy and isFeasible (Chapter 6)
8. We designed several efficient stream allocation heuristics: SteamMaxRate, Multihop-Max-Thrpt-OptF-SubOptT, Multihop-Max-Thrpt-SubOptF-OptT and Multihop-Max-Thrpt-SubOptF-SubOptT. (Chapter 7)
9. We developed yet another stream allocation heuristic: Partitioned Stream-MaxRate which scales very well with network size. Certain aspects of the algorithm need further development before it becomes fully reliable. The method in general is elegant and therefore this is an interesting subject for future work. (Chapter 8)

REFERENCES

- [1] J. Barry, E. Lee, and D. Messerschmitt, *Digital Communication, 3rd Edition*, Springer, 2003.
- [2] T. Rappaport, *Wireless Communications, Principles and Practice, 2nd Edition*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1996.
- [3] S. M. Alamouti, “A simple transmit diversity techniques for wireless communications,” *IEEE Journal on Selected Areas in Communication*, vol. 16, no. 8, pp.1451-1458, October 1998.
- [4] C. E. Shannon, “A mathematical theory of communication,” *Bell Systems Technical Journal*, vol. 27, pp. 379-423, July-October, 1948.
- [5] David Tse and Pramod Viswanath, “Fundamentals of Wireless Communication”, *Cambridge University Press*, 2005
- [6] P. Gupta and P. R. Kumar, “The Capacity of Wireless Networks”, *IEEE Transactions on Informtion Theory*, vol. 46, no. 2, pp. 388-404, March 2000.
- [7] S. Chu, X. Wang, “Opportunistic and Cooperative Spatial Multiplexing in MIMO Ad Hoc Networks”, *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, pp. 63-72, October 2010.
- [8] B. Hamdaoui, K. Shin, D. Jiang, “Characterization and Analysis of Multi-Hop Wireless MIMO Network Throughput”, *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pp. 120 - 129, 2007.
- [9] R. Bhatia, L. Li, “Throughput Optimization of Wireless Mesh Networks with MIMO Links”, *Proceedings of IEEE International Conference on Computer Communications*, pp. 2326 - 2330, 2007.

- [10] M. Alicherry, R. Bathia, L. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-Radio Wireless Mesh Networks", *Proceedings of ACM International Conference on Mobile Computing and Networking*, pp. 58-72, 2005.
- [11] L. Badia, A. Erta, L. Lenzini, F. Rossetto, M. Zorzi, "A Physical Model Scheduler for Multi-Hop Wireless Networks Based on Local Information", *Proceedings of IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pp. 213- 222, 2008.
- [12] G. Brar, D. Blough, and P. Santi, "Computationally Efficient Scheduling with the Physical Interference Model for Throughput Improvement in Wireless Mesh Networks", *Proceedings ACM International Conference on Mobile Computing and Networking*, pp. 2 - 13, 2006.
- [13] S. Ramanathan, E. Lloyd, "Scheduling Algorithms for Multihop Radio Networks", *IEEE Transactions on Networking*, vol. 1, pp. 166 - 177, 1993.
- [14] J. Wang, et al., "Joint bandwidth allocation, element assignment and scheduling for wireless mesh networks with MIMO links", *Computer Communications*, Elsevier, vol. 31, pp. 1372-1384, 2008.
- [15] E. Baccarelli, M. Biagi, C. Pelizzoni, N. Cordeschi, and F. Garzia, "Interference Suppression in MIMO Systems for Throughput Enhancement and Error Reduction", *Proceedings of the International Conference on Wireless Communications and Mobile Computing*, pp. 611 - 616, 2006
- [16] S. Vishwanath and S. Jafar, "On the Capacity of Vector Gaussian Interference Channels", *IEEE Information Theory Workshop*, pp. 365-369, 2004.
- [17] X. Shang, B. Chen, M.J. Gans, "On the achievable sum rate for MIMO Interference Channels", *IEEE Transactions on Information Theory*, vol. 52, no.9, pp. 4313 - 4320, September 2006.

- [18] X. Shang, B. Chen, G. Kramer, and H. V. Poor, “On the Capacity of MIMO Interference Channels”, *Proceedings of 46th Annual Allerton Conference on Communication, Control and Computing*, pp. 365 - 369, 2008.
- [19] R. Chen, R. Heath, and J. Andrews, “Transmit Selection Diversity for Unitary Precoded Multiuser Spatial Multiplexing Systems With Linear Receivers”, *IEEE Transactions on Signal Processing*, vol. 55, pp. 1159 - 1171, Issue 3, March 2007.
- [20] H. Viswanathan, S. Venkatesan, and H. Huang, “Downlink capacity evaluation of cellular networks with known-interference cancellation”, *IEEE Journal on Selected Areas in Communication*, vol. 21, no. 5, pp. 802 – 811, June 2003.
- [21] W. Yu and J. M. Cioffi, “Trellis precoding for the broadcast channel”, *IEEE Proceedings of the Global Telecommunications Conference*, pp. 1344-1348, November 2001.
- [22] W. Yu and J. M. Cioffi, “Sum capacity of Gaussian vector broadcast channels”, *IEEE Transactions on Information Theory*, vol. 50, no. 9, pp. 875 - 892, September 2004.
- [23] S. Jafar and M. Fakhreddin, “Degrees of freedom for the MIMO interference channel”, *IEEE Transactions on Information Theory*, vol. 53, pp. 2637 -2642, July 2007.
- [24] T.Gou, S. A. Jafar, “Degrees of Freedom of the K User MIMO Interference Channel”, *Proceedings of 42nd Asilomar Conference on Signals, Systems and Computers(Invited Paper)*, pp. 126 - 130, October 2008.
- [25] S.JAFAR, M., “Degrees of freedom for the mimo interference channel”, *Proceedings of ACM International Conference on Mobile Computing and Networking*, pp. 2 -13, May 2006.
- [26] V. R. Cadambe and S. A. Jafar, “Interference Alignment and Degrees of Freedom of the K-User Interference Channel”, *IEEE Transactions on Information Theory*, vol. 54, no.8, pp. 3425 - 3441, August 2008.

- [27] M. Maddah-Ali, A. Motahari, and A. Khandani, “Signaling over MIMO multibase systems: Combination of multiaccess and broadcast schemes”, *Proceedings International Symposium on Information Theory*, pp. 2104 - 2108, 2006.
- [28] M. Maddah-Ali, A. Motahari, and A. Khandani, “Communication over MIMO channels: Interference alignment, decomposition, and performance analysis”, *IEEE Transactions on Information Theory*, vol. 54, issue 8, pp. 3457 -3470, August 2008.
- [29] C.Huang, V. R. Cadambe, S .A. Jafar, “Interference Alignment and the Generalized Degrees of Freedom of the X Channel”, *e-print, arxiv:0810.4741*
- [30] S. A. Jafar and S. Shamai, “Degrees of freedom region for the MIMO X channel”, *ArXiv pre-print cs.IT/0607099v2 [Online]. Available: <http://arxiv.org/abs/cs/0607099v2>*
- [31] A. Ghasemi, A.S. Motahari and A.K. Khandani, “Interference Alignment for the K User MIMO Interference Channel”, *IEEE Proceedings of the International Symposium on Information Theory*, [arXiv:0909.4604v1], Austin, Texas, U.S.A., June 13 - 18, 2010
- [32] Viveck R. Cadambe, Syed A. Jafar, “Interference Alignment and Spatial Degrees of Freedom for the K User Interference Channel”, *IEEE International Conference on Communications*, pp. 971 -975, 19-23 May 2008,
- [33] C. M. Yetis, T. Gou, S. A. Jafar and Ahmet H. Kayran, “On Feasibility of Interference Alignment in MIMO Interference Networks”, *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4771 -4782, September, 2010.
- [34] S. Gollakota, S. D. Perli, and D. Katabi, “Interference alignment and cancellation”, *Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM)*, pp. 159 - 170, August, 2009.
- [35] Sanjay Karmakar, Mahesh K Varanasi , “The Generalized Degrees of Freedom of the MIMO Interference Channel”, *eprint: arXiv ID: 1103.1672, submitted*, March 2011.

- [36] E. A. Jorswieck, E. G. Larsson and D. Danev, "Complete Characterization of the Pareto Boundary for the MISO Interference Channel", *Proceedings of IEEE Transactions on Signal Processing*, vol. 56 , issue 10, pp. 5292 -5296, August 2008.
- [37] M. Costa and A. El Gamal, "The capacity region of the discrete memoryless interference channel with strong interference", *IEEE Transactions on Information Theory*, vol. 33, no. 5, pp. 710 - 711, September 1987.
- [38] S.JAFAR, "Generalized degrees of freedom of the symmetric gaussian k user interference channel", *rXiv:0804.4489*, April 2008.
- [39] A. Carleial, "A case where interference does not reduce capacity (corresp.)", *IEEE Transactions on Information Theory*, vol. 21, no. 5, pp. 569 - 570, September 1975.
- [40] B. Hamdaoui, K. Shin, and M. Maiya, "Constraint Design and Throughput Evaluation in Multi-Band Wireless Networks using MIMO Links", *The Institution of Electronics and Telecommunication Engineers Technical Review*, pp. 101 - 107, 2009.
- [41] J. Liu, Y. Shi, and Y.T. Hou, "A Tractable and Accurate Cross- Layer Model for Multi-hop MIMO Networks", *Proceedings of the IEEE International Conference on Computer Communications*, pp. 1 - 9, 2010.
- [42] A. Host-Madsen, "Capacity bounds for cooperative diversity", *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1522 -1544, April 2006.
- [43] S. Gaur and M.A. Ingram, "Simple MMSE interference suppression for real and rate-1/2 complex orthogonal space-time block codes", *IEEE Transactions on Wireless Communication*, vol. 7, No. 8, pp. 2901 - 2904, August 2008.
- [44] K. Sundaresan, R. Sivakumar, M.A. Ingram, T-Y. Chang, "Medium Access Control in Ad Hoc Networks with MIMO links: Optimization Considerations and Algorithms", *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 350-365, 2004.
- [45] M.F.Demirkol, M.A. Ingram, "Stream Control in Networks with Interfering MIMO Links", *Wireless Communications and Networking Conference*, pp. 343 - 348, 2003.

- [46] L. E. Li, R. Alimi, S. Dawei, H. Viswanathan, Y. R. Yang, “A General Algorithm for Interference Alignment and Cancellation in Wireless Networks”, *Proceedings of the IEEE International Conference on Computer Communications*, pp. 1 - 9, March 2010.
- [47] Y.Wang, D-M. Chiu, J.C.S. Lui, “Characterizing the Capacity Gain of Stream Control Scheduling in MIMO Wireless Mesh Networks”, *IEEE Wireless Communications and Mobile Computing*, vol. 9, no. 6, pp. 819 – 829, 2009.
- [48] S. Chu, X. Wang, “Adaptive and Distributed Scheduling in Heterogeneous MIMO-based Ad Hoc Networks”, *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pp. 217 - 226, 2009.
- [49] R. Srinivasan, D. Blough, L.M. Cortes-Pea, P. Santi, “Maximizing Throughput in MIMO Networks with Variable Stream Rates”, *IEEE Proceedings European Wireless*, pp. 551 - 559, 12-15 April 2010.
- [50] D. Blough, G. Resta, P. Santi, R. Srinivasan, L.M. Cortes-Pea, “Optimal One-Shot Scheduling for Multihop MIMO Networks”, *Proceedings of the IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 1 - 9, 2011
- [51] Shan Chu and Xin Wang, “MIMO-Aware Routing in Wireless Mesh Networks”, *Proceedings of the IEEE International Conference on Computer Communications*, pp. 1 - 9, 2010.
- [52] K. Sundaresan, W. Wang, and S. Eidenbenz, “Algorithmic Aspects of Communication in Ad-hoc Networks with Smart Antennas”, *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 298 - 309, 2007.
- [53] J. Liu, Y.T. Hou, and H. Sherali, “Routing and Power Allocation for MIMO-based Ad Hoc Networks with Dirty Paper Coding”, *Proceedings of IEEE International Conference on Communications*, pp. 2859 - 2864, 2008.

- [54] R. Srinivasan, D. Blough, P. Santi, “Optimal One-Shot Stream Scheduling for MIMO Links in a Single Collision Domain”, *Proceedings of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 700 - 708, 2009.
- [55] T. Han and K. Kobayashi, “A new achievable rate region for the interference channel”, *IEEE Transactions on Information Theory*, vol. 27, pages 49 - 60, January 1981.
- [56] R. Etkin, D. Tse, and H. Wang, “Gaussian interference channel capacity to within one bit”, *IEEE Transactions on Information Theory*, vol. 54, pp. 5534 - 5562, February 2007.
- [57] R. Srinivasan, D.M.Blough “Stream Allocation Algorithms For Maximizing Throughput in Multi-hop MIMO Networks”, *to be submitted in 2011*.
- [58] G.J. Foschini and M.J. Gans, “On Limits of Wireless Communications in a Fading Environment when Using Multiple Antennas”, *Wireless Personal Communications, Springer*, pp. 311 - 335, 1998.
- [59] A.F. Molisch, “Wireless Communications”, *John Wiley and Sons*, Chichester, UK, 2005.
- [60] D. Gesbert, M. Shafi, D. Shiu, P. Smith, and A. Naguib, “From theory to practice: an overview of MIMO space-time coded wireless systems”, *IEEE Journal on Selected Areas in Communications*, vol. 21, pp. 281 – 302, 2003.
- [61] Karthikeyan Sundaresan, Raghupathy Sivakumar, Mary Ann Ingram and Tae-Young Chang, “A Fair Medium Access Control Protocol for Ad-hoc Networks with MIMO Links”, *IEEE International Conference on Computer Communications*, Hong Kong, China, pp. 2559-2570, March 2004.
- [62] K. Sundaresan and R. Sivakumar, “Routing in Ad-Hoc Networks with MIMO Links”, *IEEE International Conference on Network Protocols*, Boston, MA, USA, November 2005 (Best Paper Award).

- [63] M. Debbah and R. R. Muller, “MIMO channel modeling and the principle of maximum entropy”, *IEEE Transactions on Information Theory*, vol. 51, no. 5, pages 1667-1690, 2005.
- [64] B. Mumey, J. Tang, and T. Hahn, “Algorithmic Aspects of Communications in Multihop Wireless Networks with MIMO Links”, *Proceedings of the IEEE International Conference on Communications*, pp. 1 - 6, 2010.
- [65] E. Biglieri, A. Goldsmith, B. Muquet and H. Sari, “Diversity, Interference Cancellation and Spatial Multiplexing in MIMO Mobile WiMAX Systems”, *IEEE Mobile WiMax Symposium*, pp. 74 - 79, January 2007.
- [66] Karthikeyan Sundaresan, Raghupathy Sivakumar, “A Unified MAC Layer Framework for Ad-Hoc Networks With Smart Antennas”, *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 244 - 255, June 2007.
- [67] M.F. Demirkol and M.A. Ingram, “Control Using Capacity Constraints for Interfering MIMO Links”, *Proceedings of International Symposium on Personal, Indoor, and Mobile Radio Communication*, 2010, vol. 3, pp. 1032 - 1036, Sept. 2002.
- [68] A. Goldsmith, European Wireless 2010, Keynote presentation.
- [69] F. Negro, S. P. Shenoy, I. Ghauri, D. T. M. Slock, “Interference alignment feasibility in constant coefficient MIMO interference channels”, *IEEE Eleventh International Workshop on Signal Processing Advances in Wireless Communications*, pp. 1 - 5, 2010.
- [70] D. Gesbert, H. Bolcskei, D. A. Gore, and A. J. Paulraj, “Outdoor MIMO wireless channels: models and performance prediction”, *IEEE Transactions on Communications*, vol. 50, no. 12, pp. 1926 - 1934, 2002.
- [71] P. Almers, E. Bonek et al, “Survey of Channel and Radio Propagation Models for Wireless MIMO Systems”, *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, Issue 1, Article ID 19070, pp. 56 – 56, 2007.

- [72] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, “Impact of interference on multi-hop wireless network performance”, *Proceedings of ACM MOBICOM*, vol. 11, issue 4, pp. 471 - 487, 2003.
- [73] R. Preis, “Linear Time 1/2-Approximation Algorithm for Maximum Weighted Matching in General Graphs”, *Lecture Notes in Computer Science*, vol. 1563, pp. 259 – 269, 1999.
- [74] N. Devroye and M. Sharif, “The multiplexing gain of MIMO X-channels with partial transmit side-information”, *Proceedings of IEEE International Symposium on Information Theory*, Nice, France, Jun. 2007, pp. 111 – 115.
- [75] G. Caire and S. Shamai, “On the achievable throughput of a multiantenna Gaussian broadcast channel”, *IEEE Transactions on Information Theory*, vol. 49, no. 7, pp. 1691 - 1706, July 2003.
- [76] S. Vishwanath, N. Jindal, and A. Goldsmith, “Duality, achievable rates and sum-rate capacity of Gaussian MIMO broadcast channels”, *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2658-2668, Oct. 2003.
- [77] P. Viswanath and D. N. C. Tse, “Sum capacity of the vector Gaussian broadcast channel and uplink-downlink duality”, *IEEE Transactions on Information Theory*, vol. 49, no. 8, pp. 1912 – 1921, Aug. 2003.
- [78] D. Blough, G. Resta, and P. Santi, “Approximation Algorithms for Wireless Link Scheduling with SINR-based Interference”, *IEEE/ACM Transactions on Networking*, vol. 18, pp. 1701-1712, Dec. 2010.
- [79] P. Santi, D. Blough, S. Das, R. Maheshwari, and G. Resta, “Wireless Link Scheduling under a Graded SINR Interference Model”, *Proceedings of the ACM International Workshop on Foundations of Ad Hoc and Sensor Networking and Computing*, pp. 3 - 12, 2009.

- [80] A. F. Naguib, N. Seshadri, and A. R. Calderbank, "Applications of spacetime block codes and interference suppression for high capacity and high data rate wireless systems", *Proceedings of the IEEE Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1803-1810, November 1998.
- [81] A. Stamoulis, N. AL-Dhahir, and A. R. Calderbank, "Further results on interference cancellation and space-time block codes", *Proceedings 35th Asilomar Conference on Signals Systems and Computers*, vol. 1, pp. 257 - 261, November 2001.
- [82] T. ElBatt, "Towards Scheduling MIMO links in Interferencelimited Wireless Ad Hoc Networks", *IEEE Proceedings of International Military Communications Conference*, pp. 1-7, 2007.
- [83] B. Zhang, G. Zhu, Y. Liu, Y. Deng and Y. He, "Downlink Precoding for Multiuser Spatial Multiplexing MIMO System Using Linear Receiver", *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, 2005.
- [84] H. Weingarten, Y. Steinberg, and S. S. (Shitz), "The capacity region of the Gaussian MIMO broadcast channel", *Proceedings of the Conference on Information Sciences and Systems*, pp. 174-174, March, 2004.
- [85] T. Yoo and A. Goldsmith, "Capacity and optimal power allocation for fading MIMO channels with channel estimation error", *IEEE Transactions on Information Theory*, vol.2, pp. 808 - 813 August 2002.
- [86] G. Brar, D. Blough, and P. Santi "The SCREAM Approach for Efficient Distributed Scheduling with Physical Interference in Wireless Mesh Networks", *Proceedings of the International Conference on Distributed Computing Systems*, pp. 214 - 224, 2008.
- [87] IEEE 80211n-2009-Amendment 5: Enhancements for Higher Throughput, *IEEE Standards Association* (<http://dx.doi.org/10.1109/IEEESTD.2009.5307322>), Oct. 2009.

- [88] S. Gaur and M.A. Ingram, “Stream Control for Interfering MIMO Links with Linear MMSE Receivers”, *Asilomar Conference on Signals, Systems, and Computers*, pp. 1664 - 1668, November 2007
- [89] S. Lakshmanan and R. Sivakumar, “Diversity Routing for Multi-hop Wireless Networks with Cooperative Transmissions”, *Proceedings of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 1 - 9, 2009
- [90] A. Keshavarz-Haddad, R. Riedi, “On the Broadcast Capacity of Multihop Wireless Networks: Interplay of Power, Density and Interference”, *Proceedings of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 314 - 323, 2007.
- [91] A. Goldsmith, S. A. Jafar, N. Jindal, and S. Vishwanath, “Capacity Limits of MIMO Channels”, *IEEE Journal on Selected Areas in Communications, Special Issue on MIMO systems*, vol. 21, pp. 684 - 702 , November 2003.
- [92] T. Nandagopal, T-E. Kim, X. Gao, and V. Bhargavan, “Achieving MAC Layer Fairness in Wireless Packet Networks”, *in Proceedings of the ACM International Conference on Mobile Computing and Networking*, August, 2000.
- [93] T. Ozugur, M. Naghshineh, P. Kermani, C. Olsen, B. Rezvani, and J. Copeland, “Balanced Media Access Methods for Wireless Networks”, *in Proceedings of the ACM International Conference on Mobile Computing and Networking*, October 1998.
- [94] L. M. Cortés-Peña, D. M. Blough, J. R. Barry, “Quantifying the performance loss of assigned interference cancellation techniques in MIMO networks”, *to be submitted to ICC*.