# ROBUST BEAMFORMING FOR TWO-WAY RELAY SYSTEMS

A Dissertation

by

AHSAN UL AZIZ

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

| | |
|---|---|
| Chair of Committee, | Costas N. Georghiades |
| Co-Chair of Committee, | Shuguang Cui |
| Committee Members, | Krishna Narayanan |
| | Chanan Singh |
| | Ronald Bryan |
| Head of Department, | Chanan Singh |

August 2013

Major Subject: Electrical Engineering

# ABSTRACT

In wireless communication systems, relays are widely used to extend coverage. Over the past years, relays have evolved from simple repeaters to more sophisticated units that perform signal processing to improve signal to interference plus noise ratio (SINR) or throughput (or both) at the destination receiver. There are various types of relays such as amplify and forward (AF), decode and forward (DF), and compress and forward (CF) (or estimate and forward (EF)) relays. In addition, recently there has been a growing interest in two-way relays (TWR). By utilizing the concept of analog network coding (ANC), TWRs can improve the throughput of a wireless system by reducing the number of time slots needed to complete a bi-directional message exchange between two destination nodes. It's well known that the performance of a TWR system greatly depends on its ability to apply signal processing techniques to effectively mitigate the self interference and noise accumulation, thereby improving the SINR. We study a TWR system that is equipped with multiple antennas at the relay node and a single antenna at the two destination nodes. Different from traditional work on TWR, we focus on the case with imperfect knowledge of channel state information (CSI).

For such a TWR, we formulate a robust optimization problem that takes into account norm-bounded estimation errors in CSI and designs an optimal beamforming matrix. Realizing the fact that this problem is extremely hard to solve globally, we derive two different methods to obtain either optimal or efficient suboptimal beamforming matrix solutions. The first method involves solving the robust optimization problem using the S-procedure and semidefinite programming (SDP) with rank-one relaxation. This method provides an optimal solution when the rank-one relaxation

condition for the SDP is satisfied. In cases where the rank-one condition cannot be satisfied, it's necessary to resort to sub-optimal techniques. The second approach presented here reformulates the robust non-convex quadratically constrained quadratic programming (QCQP) into a robust linear programming (LP) problem by using first-order perturbation of the optimal non-robust beamforming solution (which assumes no channel estimation error). Finally, we view the TWR robust beamforming problem from a practical standpoint and develop a set of iterative algorithms based on Newton's method or the steepest descent method that are practical for hardware implementation.

# DEDICATION

To my grandmother and my son Abrar.

# ACKNOWLEDGEMENTS

I am deeply grateful to my advisors, Prof. Costas N. Georghiades and Prof. Shuguang Cui, for their guidance and support. Dr. Georghiades is not just my advisor, he has always been a mentor to me throughout my career since my MS study under him. I am honored to have him as my Ph.D advisor. I would like to thank Dr. Cui for introducing me to the research topic and his strong guidance in my research. I highly appreciate his attention to details that made this dissertation stronger. He is an excellent advisor and a good friend. I would like to give special thanks to Prof. Christopher Thron, who also provided a lot of advises and co-authored papers with me. Dr. Thron is an outstanding person, a brilliant researcher, and a great friend; we had many long discussions on various topics related to this thesis. He provided some unique perspective to my research. Being an off-campus student and a full-time professional, the big part I missed is the interaction with other students from the group. I am grateful to Dr. Meng Zeng, a former member of Prof. Cui's group, for providing me a very nice start on my research topic. He has just been a great collaborator and a nice person to talk to on various topics. I would like to acknowledge my committee members Prof. Krishna Narayanan, Prof. Chanan Singh and Prof. Ronald Bryan for their support and having faith in me till this final point.

My heartiest gratitude goes to my family. I thank my parents and my grandmother who wanted nothing but the best for me at all time. My son Abrar provided me the inspiration I needed to continue my education. I deeply appreciate my wife, Nusrat, who not only provided me the mental support but killed my slackness, so that I could continue my education. Without her understanding and encouragement, this work would not have come to fruition. I would like to also thank my employer

National Instruments and in particular my manager James Kimery for providing a perfect work environment and a path for personal development. Finally, I believe from the deepest of my heart that it's God's blessing that made everything possible. I will always remain indebted to him.

NOMENCLATURE

| | |
|---|---|
| $a, A$ | Scalar - normal font letters |
| $\mathbf{a}$ | Vectors - bold lower case |
| $\boldsymbol{A}$ | Matrix - bold upper case |
| $\boldsymbol{I}_M, \boldsymbol{O}_M$ | Identity matrix and Zero matrix of dimension $M$ |
| $\mathbf{1}_{M\times 1}$ | All one column vector of dimension $M$ |
| $\text{vec}(\boldsymbol{A})$ | Vectorization of matrix $\boldsymbol{A}$ |
| $\text{ivec}(\boldsymbol{a})$ | Inverse operation of $\text{vec}(\boldsymbol{A})$ |
| $\otimes$ | Kronecker product |
| $\odot$ | Hadamard (elementwise) multiplication |
| $\mathbf{A} \succeq 0$ | $\mathbf{A}$ is positive semi-definite |
| $(\cdot)^T$ | Matrix transpose |
| $\text{diag}(\mathbf{a})$ | Matrix with diagonal entry $\text{diag}(\mathbf{a})$ |
| $\text{tr}(\cdot)$ | Trace of a matrix |
| $\text{rank}(\cdot)$ | Rank of a matrix |
| $(\cdot)^T$ | transpose |
| $(\cdot)^H$ | Hermitian transpose |
| $(\cdot)^*$ | conjugate operations |
| $\Re(\cdot)$ | real part of a complex variable |
| $\Im(\cdot)$ | imaginary part of a complex variable |
| $\mathcal{R}, \mathcal{C}$ | Real and complex number fields respectively |
| $\mathcal{R}^{N\times 1}, \mathcal{C}^{N\times 1}$ | $N \times 1$ Real and complex vector fields respectively |
| $\mathcal{R}^{N\times N}, \mathcal{C}^{N\times N}$ | $N \times N$ Real and complex matrix fields respectively |

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

# 1. INTRODUCTION

A wireless system always faces the inherent challenges of communicating over the freespace media (wireless channel). In many cases, the wireless channel can be very harsh thereby posing serious challenges for the system to operate effectively. Below are some typical conditions under which a wireless system may need to operate. First, the electromagnetic waves travelling from the transmitter to the receiver always undergo path loss, whereby the electromagnetic signal strength decays according to an inverse power-law over the distance traveled [1]. In addition to path loss, signals suffer from large-scale shadowing which is dependent on the terrain in which the system is operating, where large object such as hills or large buildings obscure the main signal path between the transmitter and the receiver. Then there are the effects of small-scale multipath fading, which is due to the multipath signal reflection, propagation, and non coherent combining. Depending on the relative speed between the source and destination nodes, the fading fluctuations can be very drastic. A deep channel fade can cause the received signal strength to drop drastically, hindering the ability of the destination node to decode the source message. At last, transmissions from other nodes can cause interferences, which is inevitable in wireless communications.

To combat the above issues, modern digital wireless communication systems employ a variety of techniques, some of which assume additional hardware resources such as multiple transmitter and (or) receiver antennas and relay nodes. In some cases, as in a distributed network, the antennas can even be geographically distributed. Most of these communication systems use sophisticated signal processing and channel coding techniques to take the full advantage of the additional hardware

resources to combat harsh wireless channel conditions. Beamforming is a technique that is widely used to obtain significant boost in performance in a wireless system. In addition, wireless relays are deployed to extend coverage and to improve the quality of service (QoS). In the following subsections of this chapter, we will discuss various types of wireless relay systems and in particular, two-way relay (TWR) systems. We will then discuss the concept of beamforming in wireless communications system and its application to relay networks. Next, we will introduce the concept of robust beamforming followed by a discussion on prior works in beamforming and robust beamforming, and finally we will discuss the main contribution of this dissertation.

## 1.1 Relays for Wireless Systems

In a wireless communication system, relays may be used to improve coverage and signal quality. A relay node typically resides between the source and the destination to assist in communications. Wireless relays can be roughly categorized into three major types : amplify and forward (AF), decode and forward (DF), and compress and forward (CF) (or estimate and forward (EF)). AF relays, traditionally known as repeaters, simply amplify the received signal at the relay node and forward that to the destination. There is no signal processing done (in single-antenna relays) to improve the signal to interference-plus-noise ratios (SINR) of the forwarded signal, such that both signal and noise get amplified. In a DF relay, the signals received at the relay node are decoded and re-encoded before being forwarded to the destination, thereby improving the SINR. In the case of CF (or EF), the received signals at the relay nodes are decoded and re-encoded to improve signal quality. The DF and CF schemes can provide significant performance gain at the expense of some nominal encoding/decoding latency and added complexity.

In addition to the various relaying methods mentioned above, there are also

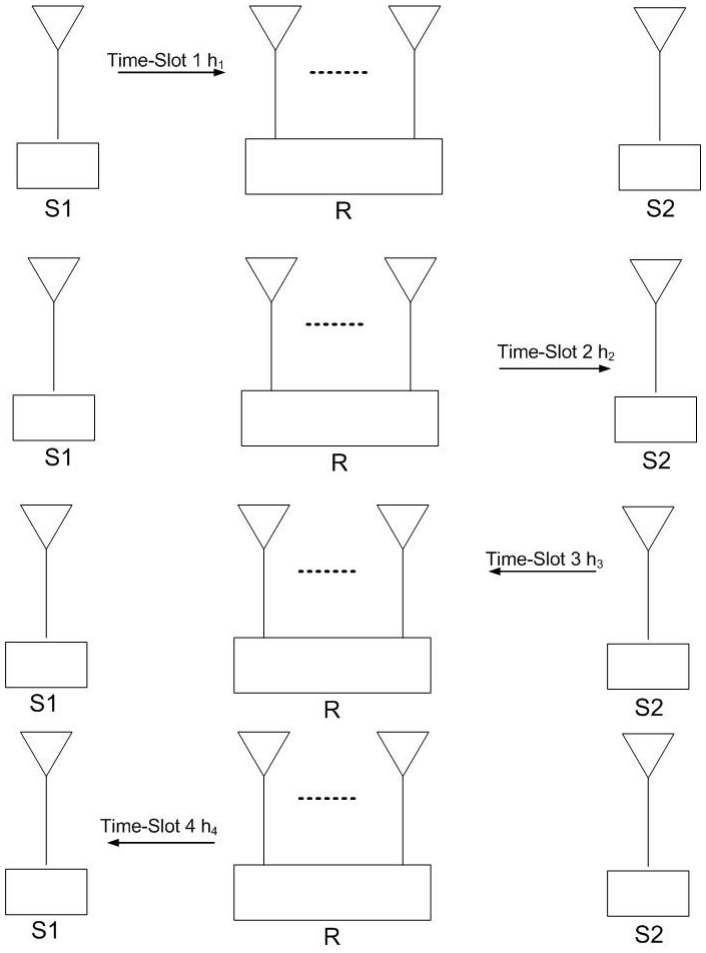


Figure 1.1: One-Way Relay System.

different signal multiplexing techniques for relay systems. In a typical one-way relay (OWR) system, as shown in the Fig. 1.1, it takes four time slots for the information exchange between nodes $S1$ and $S2$. This can be reduced to two time slots using the concept of analog network coding (ANC) [2] and TWR system [3–5]. Network coding [2] allows intermediate network nodes to mix the data or signals received from multiple links. As opposed to completely avoiding interference by separating interference and signals of interest, ANC tries to take advantage of the presence of interference and improve the overall spectral efficiency of the system.

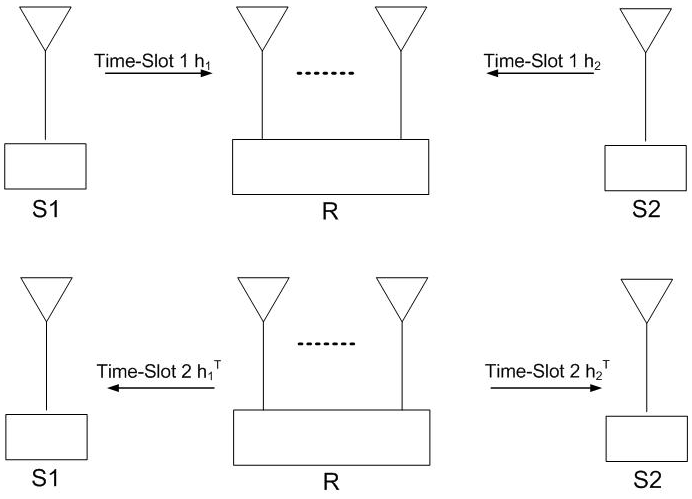


Figure 1.2: Two-way Relay System.

Consider the TWR network of Fig. 1.2. In the first time slot, both $S1$ and $S2$ transmit their information to the relay node $R$. The relay node receives the two "interfering" signals and performs some signal processing and broadcasts that signal to $S1$ and $S2$ simultaneously. Since $S1$ and $S2$ know their own transmitted information, they can remove the "self-interference" from the received signal that was broadcasted by the relay in the second time slot, thereby achieving a better decoding of the intended information. The applications of channel coding techniques for TWR systems by the authors of [6], where they introduced the use of lattice codes to directly decode a function of the two transmitted signals, which is closely related to the sum of the two functions. They proposed a scheme that can be thought of as a joint physical-layer network-layer code which showed significant gain over general analog network coding schemes. In addition to channel coding, beamforming is a technique that can be used both at the transmitter side or at the receiver side of a communication system to improve system performance with multiple antennas. In the following section we will introduce the idea of beamforming in general and how

it applies to wireless relay systems in particular.

## 1.2 Beamforming in Wireless Systems

Single-antenna systems have fixed radiation pattern, that cannot be flexibly changed to concentrate the energy in a particular direction without physically re-orienting the antenna element. When the transmitter/receiver or both nodes contain multiple antenna elements, beamforming can be applied on both side to further enhance the signal quality (through diversity). Beamforming is a technique that has historically evolved from radar and sonar applications, where it is used to detect and locate objects by steering the angular response of an antenna array in any target direction [7–10]. In this sense, beamforming is a signal processing technique that enables a system to direct the signals in the direction of interest by controlling the phase and amplitude of the transmitted or received signals from each antenna element. This is typically achieved by applying complex gains at the antenna array to create the beam patterns.

Beamforming can be divided into two major categories: static beamforming or beamsteering and adaptive beamforming. Beamsteering involves applying a set of fixed predetermined weights (including phase shifts) to an antenna array where the array elements have some fixed spacing (typically $\geq \lambda/2$ apart, where $\lambda$ is the wavelength of the carrier). Adaptive beamforming can be performed at the transmitter side or at the receiver side. In the case of transmit beamforming, the transmitter can use the channel state information (CSI) between the transmitter and the intended receiver as an input to a beamforming algorithm to determine the antenna weights that are required to achieve a given performance target. For transmit beamforming, typically CSI is estimated at the receiver side and fed back to the transmitter. The transmitter uses this CSI to adapt/design the beamformer. There are several

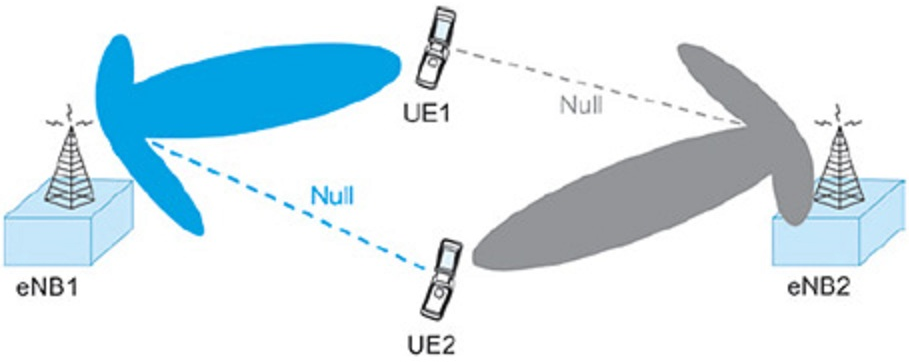


Figure 1.3: Adaptive Beamforming in a Wireless System.

advantages to this type of adaptive beamforming schemes, for instance beams can be designed very effectively to avoid interference to other unintended users. The direction of the beam can also be dynamically changed to accommodate variations in the channel condition and mobility. In addition, beamforming with geographically distributed antenna elements can be possible [11].

Fig.1.3 shows a typical application of beamforming in modern day cellular communication system. In Fig. 1.3, the base stations are identified as eNB1 and eNB2 and the mobile devises are identified as UE1 and UE2 respectively. The base station eNB1 is trying to maximize its transmitted signal energy towards its intended user UE1 while at the same time trying to minimize its interference power to mobile user UE2. Similarly the base station eNB2 is trying to maximize its transmitted energy to user UE2 while minimizing its interference to UE1. Note that eNB1 needs to know both the channel to its intended user UE1 and the unintended user UE2 in order to design this type of beamforming solution. A similar situation holds for eNB2.

In a practical wireless system, most parameter such as CSI, frequency offsets, timing offsets, SINR etc. that are critical to the operation of the system are usually

estimated. CSI are typically estimated at the receiver side and the estimates are fedback to the transmitter. Errors in CSI are inevitable and can come from at least three sources. First, there is error in estimating the CSI [12, 13]; second, typically there is restriction in the amount of data that is allowed to be fedback, which in turn effects the granularity/accuracy of the CSI [14] and third the channel may evolve (change) from the time it was observed to the time when the information is received at the transmitter [15]. If the errors in CSI are not taken into account in the design of the beamformer, the performance of the beamforming scheme can degrade very significantly and can even completely fail to provide any reasonable performance [16]. The importance of making an optimization problem (related to beamforming) immune to design parameter variation was emphasized in [16], where it was shown that a slight variation in the design parameter rendered the solution nearly useless. So, robust beamforming design is very important from a practical standpoint.

There are different ways to view the modeling of parameter uncertainties in an optimization framework. Some of the techniques that are of importance are : scenario analysis, stochastic programming and robust optimization. The choice of the modeling typically depends on the problem at hand, available information about the uncertainty parameter and the desired optimization goal. Scenario analysis is conceptually the simplest. If there are relatively few scenarios for the uncertainty parameter then this approach maybe the best one. In this case each of the possible scenario are included separately and the optimization problem is solved for each scenario. This method becomes prohibitive in complexity if there are many cases to consider. The next approach is stochastic programming, wherein the assumption is that some knowledge about the uncertainty set, such as its distribution, is available. The field of robust optimization was started by the seminal work of Ben-Tal and Ne-

mirovski [17]. robust optimization uses an uncertainty modeling approach suitable for a situation where the range of the uncertainty is known and not necessarily the distribution. Typically in robust optimization it is assumed that some of the inputs take an uncertain value anywhere between a fixed minimum and a fixed maximum. Solutions are designed to be feasible for all the constraints when the inputs drift within the uncertainty ranges. An extension of robust optimization is the idea of introducing a chance constraint, i.e one can provide a probability for a solution to satisfy specific constraints. After the introduction of the robust optimization framework by the authors of [17], it spurred interest in researchers to investigate its application in variety of different fields of study. The robust optimization framework assumes that the optimization problem is convex. If the problem is not convex, then typically the problem is relaxed into a convex problem. In this dissertation we mainly adopt the robust optimization framework [17] for our uncertainty modeling.

## 1.3 Prior Results on Optimal Beamformer Design

The design of optimal beamforming solution in the context of wireless systems have been studied extensively by various authors. Even though the concept of optimal beamformer design is the same, the actual solution and the problem definition vary widely based on the target application. Significant progress has been made in the field after the introduction of the convex optimization framework [18] and its efficient solution by using interior point methods [19]. The authors in [20] were some of the first to formulate an optimal transmit strategy for downlink beamforming and show the solution can be calculated efficiently using interior point methods for semidefinite programming (SDP). Their algorithm minimized the total transmitted power under certain constraints to guarantee a specific QoS. A more recent survey on application of convex optimization for beamforming is provided in [21]. Design

of downlink beamforming solution that is robust to channel uncertainty was considered by the authors in [22]. Their design criterion was to provide robustness against uncertainty in the downlink channel covariance matrices. They assumed a Gaussian distribution of the estimation error to obtain the covariance matrix for uncertainty. Then the problem was formulated to minimize the total downlink transmit power under the constraint that the outage probability does not exceed a certain threshold value. The non-convex optimization was recast into a convex rank-relaxed SDP and solved with interior point method. In [23], the authors also consider the design of beam-formers for broadcast channels with some QoS constraints for each user, under uncertain CSI at the transmitter. They considered a norm-bonded channel uncertainty for each user, and set out to design a robust beamformer that minimizes the total transmission power while satisfying each user's SINR constraint. Again, this problem is computationally intractable. They then relaxed the problem into a convex problem by applying some restriction on the uncertainties such as independence in the uncertainty between the left and righthand side of the constraint thereby making the problem convex. In other cases the uncertainty set was made more restrictive to yield convex solutions. These approaches allowed the robust problem to be reformulated into a SDP and solved via standard SDP solvers. The results were then validated empirically. In [24], the problem of transmit beamforming to multiple co-channel multicast groups were considered for the special case when the channel vectors are Vandermonde. This arises when a uniform linear antenna (ULA) array is used at the transmitter under far-field line-of-sight propagation conditions. Two design approaches were considered, the first one minimizes the total transmitted power subject to SINR constraints for each intended receiver; and second approach was maximizing the minimum received SINR under a total transmit power restriction. It was shown that for Vandermonde channel vectors, it is possible to recast the

optimization in terms of the autocorrelation sequences of the beamforming vectors, yielding an equivalent convex reformulation and the optimal beamforming vectors can be recovered using spectral factorization. In addition to that, the robust extensions of this problem is a special case that admit convex reformulation due to the channel structure. The authors in [25] address the robust downlink beamforming problem by using an iterative method to solve an alternating sequence of optimization and worst-case analysis, where in each step, a convex optimization problem is solved with interior-point method.

Next, we look at some of the prior results for robust beamformig with AF-relay. In [26], the authors considered the problem of designing a distributed beamfroming solution for non-regenerative relaying with imperfect CSI at the relays. The beamformer design was based on the minimization of the total relay transmit power under a SINR constraint. They adopted a worst-case design approach to recast the problem into a convex optimization problem that is tractable. In [27] the optimal beamforming solution for distributed AF relay under channel uncertainty was provided (with additional condition on the quality of the estimated CSI). The design problem was also defined as minimization of the relay transmit power subject to SINR constraints at the destination nodes. They formulated the robust optimization problem with S-Procedure and rank relaxation techniques and obtained an optimal solution under some restrictions. In addition, a sub-optimal solution with a lower complexity was proposed.

The design of a optimal non-robust beamforming matrix associated with a TWR with multiple antennas at the relay node and a single antenna at each source node was addressed by the authors of [5], where they characterized the capacity region for a TWR system. They formulated the problem to minimize the relay transmit power subject to SINR constraints at the source nodes. The problem was then transformed

into a SDP and solved with rank-one relaxation. They also showed that it is always possible to find a solution which is rank-one and hence optimal. Their work provided a upper bound on for optimal performance in a TWR system equipped with multiple antennas at the relay node and a single antenna at each source node.

In [28] the authors presented a joint optimization of the precoders, equalizers and relay beamformer of a multiple input multiple output (MIMO) TWR channel operating under norm-bounded CSI error. They minimized the sum mean square error (SMSE) subject to power constraints at both source nodes and the relay node. This problem is nonconvex and semi-infinite in its constraints. They used a generalized version of Peterson's lemma to handle the semi-infiniteness and reduced the original problem to a single linear matrix inequality (LMI). Then they used an iterative algorithm based on the alternating convex search methodology to solve the problem. The authors of [29] setup their TWR robust beamforming problem to maximize the minimum SINRs of two source nodes subject to a total relay power budget. They decomposed the non-convex problem into a series of relay power minimization problems under minimum SINR constraints by using bisection search. Then the relay power minimization problem was recast to a rank-relaxed SDP. Based on this they obtained a suboptimal solution.

## 1.4 Motivation and Contribution

To the best of our knowledge, there are limited work ( [28] and [29]) done on robust beamforming design for TWR systems. In this dissertation we extend the design [5] and make it robust to channel estimation errors. We represent the channel estimation errors as norm-bounded i.e. the uncertainty region as a sphere. We then define the robust optimization problem to minimize the maximum transmit power at the relay node subject to the constraints that the worst-case SINR at the two source

nodes are above a predefined threshold.

The key difference of our approach to [28] and [29] is that we minimize the maximum relay power subject to SINR constraints at the terminal nodes. Unlike [28], we do not make any additional signal processing requirements at the terminal nodes. Although we can not guarantee global optimality for the robust solution (as this problem is not convex in general, which renders the globally optimal solution extremely hard to obtain), but we can claim optimality for certain beamforming matrix solution that satisfy the convex relaxation rank-one criteria. This is an advantage over [28] and [29], as it provides a very useful upper bound on power which can be used to imperially compare other computationally efficient methods. Furthermore, we propose a very low complexity method to create a robust beamforming solution from the optimal non-robust TWR beamforming solution [5]. Lastly, we develop a more practical view for TWR robust beamforming problem based on iterative search. In order to evaluate the performance of the methods, we define two performance metric, power and outage.

In the following we summarize the main contribution of this dissertation :

- In our first approach the optimization problem is cast into an SDP problem by using the S-procedure and rank one relaxation and solved using standard SDP solver. The robust beamforming matrix is then extracted using a principle eigenvector based rank-one reconstruction. A Rank-one solution, when obtained, guarantees an optimal solution and therefore provides a lower bound for the power required by the TWR system to operate under CSI error.

  This rank-one relaxed SDP solution suffers from two weaknesses - a) optimum solution may not exist due to rank-one condition violation b) the problem can become infeasible after convex relaxation. In order to address these problem

we propose a "robust-hybrid" approach that combines the non-robust solution of [5] when the robust method fails to provide an "acceptable" beamforming solution.

- Realizing the fact that the original non-robust TWR beamforming design problem always provides a guaranteed optimal solution [5] and the fact that the channel estimation errors are typically small; we propose a method to obtain a linearized robust counterpart of the non-robust problem [5] by using perturbation and worst-case analysis. The reformulated problem becomes a linear programming (LP) problem which can be solved analytically to obtain a solution that is "acceptable" both in terms of power and outage performance. This approach has very low complexity and is a good candidate for practical implementation.

- Finally, we develop a set of iterative algorithms for TWR robust beamforming that is suitable from a practical standpoint. This technique uses the Newton's method and the steepest descent method to solve the TWR robust beamforming problem by a process of alternating between a worst-case CSI error vector search and solving an optimization problem.

The rest of the dissertation is organized as follows. In Chapter 2, we provide the system model and formulate the TWR robust beamforming optimization problem and cast it into convex SDP problem with rank-relaxation. In Chapter 3, we extend the results from [5] and introduce a linearization framework for obtaining a robust beamforming solution starting from the optimum non-robust solution [5]. In Chapter 4, we develop an iterative technique for robust TWR beamforming that is more suitable for deployment on practical system. At last we conclude our work in Chapter 5.

# 2. ROBUST BEAMFORMING FOR THE TWO-WAY RELAY SYSTEM WITH CONVEX RELAXATION

In this chapter we presents the design of a robust beamforming scheme for a TWR network, composed of one multi-antenna relay node and two single-antenna terminals, with the consideration of channel estimation errors. Given the assumption that the channel estimation errors are within a certain range, we aim to minimize the transmit power at the multi-antenna relay node and guarantee that the SINRs at the two terminals are larger than a predefined value. Such a robust beamforming matrix design problem is formulated as a non-convex optimization problem, which is then converted into an SDP problem by the S-procedure and rank-one relaxation. The robust beamforming matrix is then derived from a principle eigenvector based rank-one reconstruction algorithm. We further propose a hybrid approach based on the best-effort transmission to improve the outage probability performance, which is defined as the probability that one of two resulting terminal SINRs is less than the predefined value. Simulation results are presented to show that the robust design leads to better outage performance than the traditional non-robust approaches.

## 2.1 Introduction

In the next-generation wireless communication systems, smart relaying is one of the key technology that is being investigated to extend the cell coverage and to increase system capacity. For example, relaying will be standardized for the long term evolution-advanced (LTE-A) wireless systems, In particular TWR has drawn a great amount of attentions where multiple terminal nodes utilize common relays to

perform two-way information exchanges [2, 5, 30–32].

Specifically, the authors in [5], considered the beamforming design at the relay and characterized the capacity region with the assumption that the relay is equipped with multiple antennas and the terminal nodes are each equipped with a single-antenna. Afterwards, different variations and various system setups were considered. For example, the authors of [32] studied the achievable rate region for the two-way collaborative relay beamforming problem, where multiple collaborative single relays were considered. The authors in [30] studied the AF-based TWR system with collaborative beamforming, where the focus is to minimize the total transmit power across the terminal nodes and the relay cluster under a given pair of SINR constraints. In [31], the authors studied the case where all the nodes in the network are equipped with multiple antennas and proposed some suboptimal solutions.

However, all the above results on beamforming design for the TWR system are based on the assumption that the CSI at all links are perfectly known. Unfortunately, this assumption may not be true in practice, since CSI can only be obtained by channel estimation, which is usually not perfect. As a result, in this chapter we consider the robust TWR beamfroming under channel uncertainty where the SINR at each terminal is constrained to be larger than a predefined value. Similar problems for OWR relay robust beamforming were studied in [27] and [33] for single-antenna relays was further extended to multiple antenna in [34].

The rest of chapter is organized as follows. In Section 3.2, we present the system model and formulate the problem. Afterwards, we transform the original non-convex problem into a SDP problem in Section 2.3 with the help of the S-procedure and rank-one relaxation. In Section 2.4, we propose the principle eigenvector based rank-one reconstruction approach to obtain the beamforming matrix. In Section 2.5, simulation results are presented to show the performance improvement. In Section

3.5, we conclude the chapter.

## 2.2 System Model

We consider a two-way relay system similar to the one introduced in Chapter 1, Fig. 1.2, which consists of the relay node $R$ and two terminal nodes $S1$ and $S2$. The relay is equipped with $M$ antennas and the terminal nodes are each equipped with a single antenna. Based on the principle of ANC [2], the two terminal nodes exchange information in two consecutive time slots via the help of $R$. In the first time slot, terminal nodes $S1$ and $S2$ send messages $s_1$ and $s_2$ with power level $p_1$ and $p_2$ respectively to $R$, and the received signal at $R$ is given as

$$\boldsymbol{y}_R = \boldsymbol{h}_1\sqrt{p_1}s_1 + \boldsymbol{h}_2\sqrt{p_2}s_2 + \boldsymbol{z}_R, \tag{2.1}$$

where $\boldsymbol{h}_1, \boldsymbol{h}_2 \in \mathcal{C}^{M\times 1}$ are complex channel gains from the terminal nodes $S1$ and $S2$ to the relay respectively, $\boldsymbol{z}_R$ is the circularly symmetric complex Gaussian (CSCG) noise with covariance $\sigma_R^2\boldsymbol{I}$, and $E[s_i] = 1$, $i = 1, 2$. In the second time slot, the relay $R$ multiplies a beamforming matrix $\boldsymbol{A}$ to the received signal $\boldsymbol{y}_R$ and transmits the resulting vector signal $\boldsymbol{A}\boldsymbol{y}_R$ to the two terminal nodes. Based on the assumption of channel reciprocity [32], the received signals at $S1$ and $S2$ are given as

$$y_1 = \boldsymbol{h}_1^T\boldsymbol{A}\boldsymbol{h}_1\sqrt{p_1}s_1 + \boldsymbol{h}_1^T\boldsymbol{A}\boldsymbol{h}_2\sqrt{p_2}s_2 + \boldsymbol{h}_1^T\boldsymbol{A}\boldsymbol{z}_R + z_1, \tag{2.2}$$

$$y_2 = \boldsymbol{h}_2^T\boldsymbol{A}\boldsymbol{h}_2\sqrt{p_2}s_2 + \boldsymbol{h}_2^T\boldsymbol{A}\boldsymbol{h}_1\sqrt{p_1}s_1 + \boldsymbol{h}_2^T\boldsymbol{A}\boldsymbol{z}_R + z_2, \tag{2.3}$$

where $z_1$ and $z_2$ are the CSCG noise at $S1$ and $S2$ with variances $\sigma_1^2$ and $\sigma_2^2$, respectively. In the ideal CSI case as in [5], $S1$ and $S2$ can cancel out the self-interference terms $\boldsymbol{h}_1^T\boldsymbol{A}\boldsymbol{h}_1\sqrt{p_1}s_1$ and $\boldsymbol{h}_2^T\boldsymbol{A}\boldsymbol{h}_2\sqrt{p_2}s_2$ from $y_1$ and $y_2$, respectively. However, in practical scenarios, the terminal nodes only have access to the estimated versions of

$\boldsymbol{h}_1$ and $\boldsymbol{h}_2$, which could be modeled as $\hat{\boldsymbol{h}_1} = \boldsymbol{h}_1 - \Delta\boldsymbol{h}_1$ and $\hat{\boldsymbol{h}_2} = \boldsymbol{h}_2 - \Delta\boldsymbol{h}_2$, where $\boldsymbol{\Delta}h_1$ and $\boldsymbol{\Delta}h_2$ are the channel estimation errors. Here we assume that $\|\Delta\boldsymbol{h}_1\| \leq \epsilon_1$ and $\|\Delta\boldsymbol{h}_2\| \leq \epsilon_2$ where $\epsilon_1$ and $\epsilon_2$ are some small positive constant. Note that we could represent the true channel gains as $\boldsymbol{h}_1 = \hat{\boldsymbol{h}}_1 + \Delta\boldsymbol{h}_1$ and $\boldsymbol{h}_2 = \hat{\boldsymbol{h}}_2 + \Delta\boldsymbol{h}_2$. By substituting $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ into (2.2) and (2.3) and using the fact that the amplitude of channel estimation error is usually much smaller than the estimated channel gain, the second-order terms of the channel estimation errors can be neglected. After subtracting out the self interference terms $\hat{\boldsymbol{h}}_1^T\boldsymbol{A}\hat{\boldsymbol{h}}_1\sqrt{p_1}s_1$ and $\hat{\boldsymbol{h}}_2^T\boldsymbol{A}\hat{\boldsymbol{h}}_2\sqrt{p_2}s_2$ from $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ respectively, we obtain the received signals $\tilde{y}_1$ and $\tilde{y}_2$ as in (2.4).

$$\begin{aligned}
\tilde{y}_1 \approx \underbrace{(\hat{\boldsymbol{h}}_1^T\boldsymbol{A}\Delta\boldsymbol{h}_1 + \Delta\boldsymbol{h}_1^T\boldsymbol{A}\hat{\boldsymbol{h}}_1)\sqrt{p_1}s_1}_{\text{remaining self-interference}} &+ \underbrace{(\hat{\boldsymbol{h}}_1^T\boldsymbol{A}\hat{\boldsymbol{h}}_2 + \hat{\boldsymbol{h}}_1^T\boldsymbol{A}\Delta\boldsymbol{h}_2 + \Delta\boldsymbol{h}_1^T\boldsymbol{A}\hat{\boldsymbol{h}}_2)\sqrt{p_2}s_2}_{\text{desired signal}} \\
&+ \underbrace{(\hat{\boldsymbol{h}}_1^T\boldsymbol{A} + \Delta\boldsymbol{h}_1^T\boldsymbol{A})\boldsymbol{z}_R + z_1}_{\text{noise}}, \\
\tilde{y}_2 \approx \underbrace{(\hat{\boldsymbol{h}}_2^T\boldsymbol{A}\Delta\boldsymbol{h}_2 + \Delta\boldsymbol{h}_2^T\boldsymbol{A}\hat{\boldsymbol{h}}_2)\sqrt{p_2}s_2}_{\text{remaining self-interference}} &+ \underbrace{(\hat{\boldsymbol{h}}_2^T\boldsymbol{A}\hat{\boldsymbol{h}}_1 + \hat{\boldsymbol{h}}_2^T\boldsymbol{A}\Delta\boldsymbol{h}_1 + \Delta\boldsymbol{h}_2^T\boldsymbol{A}\hat{\boldsymbol{h}}_1)\sqrt{p_1}s_1}_{\text{desired signal}} \\
&+ \underbrace{(\hat{\boldsymbol{h}}_2^T\boldsymbol{A} + \Delta\boldsymbol{h}_2^T\boldsymbol{A})\boldsymbol{z}_R + z_2}_{\text{noise}}.
\end{aligned} \tag{2.4}$$

The corresponding transmit power at the relay $R$ is given by (2.5) and the SINRs at node $Si$ are given by the $SINR_i$ $(i = 1, 2, \quad j = 3 - i)$ in (2.6).

$$\begin{aligned}
p_R(\boldsymbol{A}) &= \|\boldsymbol{A}\boldsymbol{h}_1\|^2 p_1 + \|\boldsymbol{A}\boldsymbol{h}_2\|^2 p_2 + \mathrm{tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \\
&= p_1\hat{\boldsymbol{h}}_1^H\boldsymbol{A}^H\boldsymbol{A}\hat{\boldsymbol{h}}_1 + p_2\hat{\boldsymbol{h}}_2^H\boldsymbol{A}^H\boldsymbol{A}\hat{\boldsymbol{h}}_2 + \mathrm{tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \\
&+ 2\Re(p_1\hat{\boldsymbol{h}}_1^H\boldsymbol{A}^H\boldsymbol{A}\Delta\boldsymbol{h}_1) + 2\Re(p_2\hat{\boldsymbol{h}}_2^H\boldsymbol{A}^H\boldsymbol{A}\Delta\boldsymbol{h}_2) \\
&+ p_1\Delta\boldsymbol{h}_1^H\boldsymbol{A}^H\boldsymbol{A}\Delta\boldsymbol{h}_1 + p_2\Delta\boldsymbol{h}_2^H\boldsymbol{A}^H\boldsymbol{A}\Delta\boldsymbol{h}_2.
\end{aligned} \tag{2.5}$$

$$\mathrm{SINR}_i = \frac{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j + \hat{\boldsymbol{h}}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_j + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j|^2 p_j}{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_i + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_i|^2 p_i + \|(\hat{\boldsymbol{h}}_i^T + \Delta\boldsymbol{h}_i^T)\boldsymbol{A}\|^2\sigma_R^2 + \sigma_i^2}. \tag{2.6}$$

In order to guarantee the SINR requirements at both $S1$ and $S2$ in the presence of channel uncertainty, the SINRs are desired to be greater than or equal to $\gamma_1$ and $\gamma_2$, respectively, for all possible estimation errors. Since we assume the estimation errors are bounded, i.e., $\|\Delta\boldsymbol{h}_1\| \leq \epsilon_1, \|\Delta\boldsymbol{h}_2\| \leq \epsilon_2$, the robust beamforming design problem can be formulated as follows,

$$\begin{aligned} \min_{\boldsymbol{A}} \quad & \max_{\|\Delta\boldsymbol{h}_1\|\leq\epsilon_1, \|\Delta\boldsymbol{h}_2\|\leq\epsilon_2} p_R \\ \text{s.t.} \quad & \min_{\|\Delta\boldsymbol{h}_1\|\leq\epsilon_1, \|\Delta\boldsymbol{h}_2\|\leq\epsilon_2} \mathrm{SINR}_1 \geq \gamma_1 \\ & \min_{\|\Delta\boldsymbol{h}_1\|\leq\epsilon_1, \|\Delta\boldsymbol{h}_2\|\leq\epsilon_2} \mathrm{SINR}_2 \geq \gamma_2. \end{aligned} \tag{2.7}$$

## 2.3 SDP Formulation with Rank-one Relaxation

We note that the problem in (2.7) is not convex in general, which renders the globally optimal solution extremely hard to obtain. Therefore, we resort to sub-optimal solutions, that achieves an "acceptable" performance in terms of power and bears a low outage probability.

The problem in (2.7) could be equivalently recast as

$$\mathcal{Q}_1 : \quad \min_{\boldsymbol{A}, t} \ t \tag{2.8}$$

$$\text{s.t.} \quad \min_{\|\Delta\boldsymbol{h}_1\|\leq\epsilon_1, \|\Delta\boldsymbol{h}_2\|\leq\epsilon_2} \mathrm{SINR}_1 \geq \gamma_1 \tag{2.9}$$

$$\min_{\|\Delta\boldsymbol{h}_1\|\leq\epsilon_1, \|\Delta\boldsymbol{h}_2\|\leq\epsilon_2} \mathrm{SINR}_2 \geq \gamma_2 \tag{2.10}$$

$$\max_{\|\Delta\boldsymbol{h}_1\|\leq\epsilon_1, \|\Delta\boldsymbol{h}_2\|\leq\epsilon_2} p_R \leq t. \tag{2.11}$$

**Definition 1** *Given the SINR thresholds $\gamma_1$ and $\gamma_2$, define the system outage as the event that the resulting SINR$_1 < \gamma_1$ or/and SINR$_2 < \gamma_2$.*

In the optimization problem $\mathcal{Q}_1$, the number of constraints is essentially infinite, since $\Delta\boldsymbol{h}_1$ and $\Delta\boldsymbol{h}_2$ are of continuous complex values. As a result, problem $\mathcal{Q}_1$ cannot be solved directly. Therefore, we need to transform the original problem into an effectively solvable problem of finite dimensions by applying the following theorem [35].

**Theorem 1 (S-procedure)** *Given Hermitian matrices $\boldsymbol{A}_j \in \mathcal{C}^{n\times n}$, vectors $\boldsymbol{b}_j \in \mathcal{C}^n$, and numbers $c_j \in \mathcal{R}$, define the functions $f_j(\boldsymbol{x}) = \boldsymbol{x}^H\boldsymbol{A}_j\boldsymbol{x} + 2\Re(\boldsymbol{b}_j^H\boldsymbol{x}) + c_j$ with $\boldsymbol{x} \in \mathcal{C}^n$, $j = 0, 1, 2$. The following two conditions are equivalent.*

1. *$f_0(\boldsymbol{x}) \geq 0$ for every $\boldsymbol{x} \in \mathcal{C}^n$ such that $f_1(\boldsymbol{x}) \geq 0$ and $f_2(\boldsymbol{x}) \geq 0$;*
2. *There exist $\lambda_1, \lambda_2 \geq 0$ such that* $\begin{pmatrix} \boldsymbol{A}_0 & \boldsymbol{b}_0 \\ \boldsymbol{b}_0^H & c_0 \end{pmatrix} \succeq \lambda_1 \begin{pmatrix} \boldsymbol{A}_1 & \boldsymbol{b}_1 \\ \boldsymbol{b}_1^H & c_1 \end{pmatrix} + \lambda_2 \begin{pmatrix} \boldsymbol{A}_2 & \boldsymbol{b}_2 \\ \boldsymbol{b}_2^H & c_2 \end{pmatrix}$.

Some details about S-procedure can be found in [36] and its applications can be found in [27].

Based on the S-procedure, we first need to transform the constraints of problem $\mathcal{Q}_1$ into the quadratic forms in terms of $\Delta\boldsymbol{h}_1$ and $\Delta\boldsymbol{h}_2$. We then transform the constraints into linear matrix inequalities (LMIs). Let us define following notations: $\tilde{\boldsymbol{h}}_i \triangleq \hat{\boldsymbol{h}}_i \otimes \mathbf{1}_{M\times 1}$, $\breve{\boldsymbol{h}}_i \triangleq \mathbf{1}_{M\times 1} \otimes \hat{\boldsymbol{h}}_i$, $\Delta\tilde{\boldsymbol{h}}_i \triangleq \Delta\boldsymbol{h}_i \otimes \mathbf{1}_{M\times 1}$, and $\Delta\breve{\boldsymbol{h}}_i = \mathbf{1}_{M\times 1} \otimes \Delta\boldsymbol{h}_i$, for $i = 1, 2$, with $\mathbf{1}_{M\times 1}$ an all-one M-dimensional column vector. In addition, $\boldsymbol{a} \triangleq \mathrm{vec}(\boldsymbol{A})$, $\bar{\boldsymbol{A}} \triangleq \boldsymbol{a}\boldsymbol{a}^H$, $\boldsymbol{h} = \mathrm{vec}(\boldsymbol{h}_1, \boldsymbol{h}_2)$, $\Delta\boldsymbol{h} = \mathrm{vec}(\Delta\boldsymbol{h}_1, \Delta\boldsymbol{h}_2)$, $\boldsymbol{G}_1 = [\boldsymbol{I}_M, \boldsymbol{O}_M]$, $\boldsymbol{G}_2 = [\boldsymbol{O}_M, \boldsymbol{I}_M]$, $\boldsymbol{D}_R = \boldsymbol{I}_M \otimes \mathbf{1}_{M\times 1}$, and $\boldsymbol{D}_L = \mathbf{1}_{M\times 1} \otimes \boldsymbol{I}_M$, with $\boldsymbol{O}_M$ and $\boldsymbol{I}_M$ as the $\boldsymbol{M} \times \boldsymbol{M}$ all-zero and identity matrices respectively. It is easy to see that $\Delta\tilde{\boldsymbol{h}}_i = \boldsymbol{D}_R\boldsymbol{G}_i\Delta\boldsymbol{h}$ and $\Delta\breve{\boldsymbol{h}}_i = \boldsymbol{D}_L\boldsymbol{G}_i\Delta\boldsymbol{h}$, for $i = 1, 2$.

Let $\alpha_1$ be the numerator of $\text{SINR}_1$ (as given in (2.6)). Note that $\boldsymbol{g}^T\boldsymbol{A}\boldsymbol{h} = [(\boldsymbol{h}\otimes\boldsymbol{1}_{M\times 1})\odot(\boldsymbol{1}_{M\times 1}\otimes\boldsymbol{g})]^T\,\text{vec}(\boldsymbol{A})$, we could write $\alpha_1$ as

$$\alpha_1 = |\big(\tilde{\boldsymbol{h}}_2\odot\breve{\boldsymbol{h}}_1 + \Delta\tilde{\boldsymbol{h}}_2\odot\breve{\boldsymbol{h}}_1 + \tilde{\boldsymbol{h}}_2\odot\Delta\breve{\boldsymbol{h}}_1\big)^T\boldsymbol{a}|^2 p_2. \tag{2.12}$$

After some mathematical manipulations $\alpha_1$ can be rewritten as

$$\alpha_1 \;=\; \Delta\boldsymbol{h}^T\boldsymbol{Q}_{\boldsymbol{1}}\Delta\boldsymbol{h}^* + 2\Re(\boldsymbol{q}_1^H\Delta\boldsymbol{h}^*) + c_1,$$

where

$$\begin{aligned}
\boldsymbol{Q}_1 &= p_2\boldsymbol{G}_2^T\boldsymbol{D}_R^T\text{diag}(\breve{\boldsymbol{h}}_1)\bar{\boldsymbol{A}}\text{diag}(\tilde{\boldsymbol{h}}_2^*)\boldsymbol{D}_L\boldsymbol{G}_1 \\
&\quad + p_2\boldsymbol{G}_1^T\boldsymbol{D}_L^T\text{diag}(\tilde{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\text{diag}(\breve{\boldsymbol{h}}_1^*)\boldsymbol{D}_R\boldsymbol{G}_2 \\
&\quad + p_2\boldsymbol{G}_2^T\boldsymbol{D}_R^T\text{diag}(\breve{\boldsymbol{h}}_1)\bar{\boldsymbol{A}}\text{diag}(\breve{\boldsymbol{h}}_1^*)\boldsymbol{D}_R\boldsymbol{G}_2 \\
&\quad + p_2\boldsymbol{G}_1^T\boldsymbol{D}_L^T\text{diag}(\tilde{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\text{diag}(\tilde{\boldsymbol{h}}_2^*)\boldsymbol{D}_L\boldsymbol{G}_1, \\
\boldsymbol{q}_1^H &= p_2(\tilde{\boldsymbol{h}}_2\odot\breve{\boldsymbol{h}}_1)^T\bar{\boldsymbol{A}}\text{diag}(\breve{\boldsymbol{h}}_1^*)\boldsymbol{D}_R\boldsymbol{G}_2 \\
&\quad + p_2\big(\tilde{\boldsymbol{h}}_2\odot\breve{\boldsymbol{h}}_1\big)^T\bar{\boldsymbol{A}}\text{diag}(\tilde{\boldsymbol{h}}_2^*)\boldsymbol{D}_L\boldsymbol{G}_1, \\
c_1 &= p_2\big(\tilde{\boldsymbol{h}}_2\odot\breve{\boldsymbol{h}}_1\big)^T\bar{\boldsymbol{A}}\big(\tilde{\boldsymbol{h}}_2\odot\breve{\boldsymbol{h}}_1\big)^*,
\end{aligned}$$

Similarly, the denominator of $\text{SINR}_1$, denoted by $\beta_1$, can be written as

$$\beta_1 \;=\; \Delta\boldsymbol{h}^T\boldsymbol{Q}_2\Delta\boldsymbol{h}^* + 2\Re(\boldsymbol{q}_2^H\Delta\boldsymbol{h}^*) + c_2,$$

where

$$\begin{aligned}
\boldsymbol{Q}_2 &= p_1 \boldsymbol{G}_1^T \boldsymbol{D}_R^T \mathrm{diag}(\breve{\boldsymbol{h}}_1) \bar{\boldsymbol{A}} \mathrm{diag}(\breve{\boldsymbol{h}}_1^*) \boldsymbol{D}_R \boldsymbol{G}_1 \\
&\quad + p_1 \boldsymbol{G}_1^T \boldsymbol{D}_L^T \mathrm{diag}(\tilde{\boldsymbol{h}}_1) \bar{\boldsymbol{A}} \mathrm{diag}(\tilde{\boldsymbol{h}}_1^*) \boldsymbol{D}_L \boldsymbol{G}_1 \\
&\quad + p_1 \boldsymbol{G}_1^T \boldsymbol{D}_L^T \mathrm{diag}(\tilde{\boldsymbol{h}}_1) \bar{\boldsymbol{A}} \mathrm{diag}(\breve{\boldsymbol{h}}_1^*) \boldsymbol{D}_R \boldsymbol{G}_1 \\
&\quad + p_1 \boldsymbol{G}_1^T \boldsymbol{D}_R^T \mathrm{diag}(\breve{\boldsymbol{h}}_1) \bar{\boldsymbol{A}} \mathrm{diag}(\tilde{\boldsymbol{h}}_1^*) \boldsymbol{D}_L \boldsymbol{G}_1 \\
&\quad + \sigma_R^2 \boldsymbol{G}_1^T \boldsymbol{D}_L^T \boldsymbol{E} \odot \bar{\boldsymbol{A}} \boldsymbol{D}_L \boldsymbol{G}_1,
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{q}_2^H &= \sigma_R^2 \breve{\boldsymbol{h}}_1^T \boldsymbol{E} \odot \bar{\boldsymbol{A}} \boldsymbol{D}_L \boldsymbol{G}_1, \\
c_2 &= \sigma_R^2 \breve{\boldsymbol{h}}_1^T \boldsymbol{E} \odot \bar{\boldsymbol{A}} \breve{\boldsymbol{h}}_1^* + \sigma_1^2.
\end{aligned}$$

Now, we rewrite constraint (2.9) as

$$\begin{cases}
\Delta \boldsymbol{h}^T (\boldsymbol{Q}_1 - \gamma_1 \boldsymbol{Q}_2) \Delta \boldsymbol{h}^* \\
+2\Re\big((\boldsymbol{q}_1 - \gamma_1 \boldsymbol{q}_2)^H \Delta \boldsymbol{h}^*)\big) + c_1 - \gamma_1 c_2 \geq 0 \\
\Delta \boldsymbol{h}^T \boldsymbol{G}_1^H \boldsymbol{G}_1 \Delta \boldsymbol{h}^* \leq \epsilon_1^2 \\
\Delta \boldsymbol{h}^T \boldsymbol{G}_2^H \boldsymbol{G}_2 \Delta \boldsymbol{h}^* \leq \epsilon_2^2.
\end{cases}$$

According to the S-procedure in Theorem 1, the above quadratic three-inequality system in equivalent to the following LMI:

$$\begin{pmatrix}
\boldsymbol{Q}_1 - \gamma_1 \boldsymbol{Q}_2 + \lambda_1 \boldsymbol{G}_1^H \boldsymbol{G}_1 + \lambda_2 \boldsymbol{G}_2^H \boldsymbol{G}_2 & \boldsymbol{q}_1 - \gamma_1 \boldsymbol{q}_2 \\
\boldsymbol{q}_1^H - \gamma_1 \boldsymbol{q}_2^H & c_1 - \gamma_1 c_2 - \lambda_1 \epsilon_1^2 - \lambda_2 \epsilon_2^2
\end{pmatrix} \succeq 0$$

Similarly, for $\mathrm{SINR}_2$, we have

$$\begin{aligned}\alpha_2 &= \Delta\boldsymbol{h}^T\boldsymbol{Q_3}\Delta\boldsymbol{h}^* + 2\Re(\boldsymbol{q}_3^H\Delta\boldsymbol{h}^*) + c_3\\ \beta_2 &= \Delta\boldsymbol{h}^T\boldsymbol{Q}_4\Delta\boldsymbol{h}^* + 2\Re(\boldsymbol{q}_4^H\Delta\boldsymbol{h}^*) + c_4,\end{aligned}$$

where

$$\begin{aligned}\boldsymbol{Q}_3 &= p_1\boldsymbol{G}_1^T\boldsymbol{D}_R^T\mathrm{diag}(\breve{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\mathrm{diag}(\tilde{\boldsymbol{h}}_1^*)\boldsymbol{D}_L\boldsymbol{G}_2\\ &\quad +p_1\boldsymbol{G}_2^T\boldsymbol{D}_L^T\mathrm{diag}(\tilde{\boldsymbol{h}}_1)\bar{\boldsymbol{A}}\mathrm{diag}(\breve{\boldsymbol{h}}_2^*)\boldsymbol{D}_R\boldsymbol{G}_1\\ &\quad +p_1\boldsymbol{G}_1^T\boldsymbol{D}_R^T\mathrm{diag}(\breve{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\mathrm{diag}(\breve{\boldsymbol{h}}_2^*)\boldsymbol{D}_R\boldsymbol{G}_1\\ &\quad +p_1\boldsymbol{G}_2^T\boldsymbol{D}_L^T\mathrm{diag}(\tilde{\boldsymbol{h}}_1)\bar{\boldsymbol{A}}\mathrm{diag}(\tilde{\boldsymbol{h}}_1^*)\boldsymbol{D}_L\boldsymbol{G}_2\\ \boldsymbol{q}_3^H &= p_1(\tilde{\boldsymbol{h}}_1\odot\breve{\boldsymbol{h}}_2)^T\bar{\boldsymbol{A}}\mathrm{diag}(\breve{\boldsymbol{h}}_2^*)\boldsymbol{D}_R\boldsymbol{G}_1,\\ &\quad +p_1\big(\tilde{\boldsymbol{h}}_1\odot\breve{\boldsymbol{h}}_2\big)^T\bar{\boldsymbol{A}}\mathrm{diag}(\tilde{\boldsymbol{h}}_1^*)\boldsymbol{D}_L\boldsymbol{G}_2,\\ c_3 &= p_1\big(\tilde{\boldsymbol{h}}_1\odot\breve{\boldsymbol{h}}_2\big)^T\bar{\boldsymbol{A}}\big(\tilde{\boldsymbol{h}}_1\odot\breve{\boldsymbol{h}}_2\big)^*,\end{aligned}$$

and

$$\begin{aligned}\boldsymbol{Q}_4 &= p_2\boldsymbol{G}_2^T\boldsymbol{D}_R^T\mathrm{diag}(\breve{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\mathrm{diag}(\breve{\boldsymbol{h}}_2^*)\boldsymbol{D}_R\boldsymbol{G}_2\\ &\quad +p_2\boldsymbol{G}_2^T\boldsymbol{D}_L^T\mathrm{diag}(\tilde{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\mathrm{diag}(\tilde{\boldsymbol{h}}_2^*)\boldsymbol{D}_L\boldsymbol{G}_2\\ &\quad +p_2\boldsymbol{G}_2^T\boldsymbol{D}_L^T\mathrm{diag}(\tilde{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\mathrm{diag}(\breve{\boldsymbol{h}}_2^*)\boldsymbol{D}_R\boldsymbol{G}_2\\ &\quad +p_2\boldsymbol{G}_2^T\boldsymbol{D}_R^T\mathrm{diag}(\breve{\boldsymbol{h}}_2)\bar{\boldsymbol{A}}\mathrm{diag}(\tilde{\boldsymbol{h}}_2^*)\boldsymbol{D}_L\boldsymbol{G}_2\\ &\quad +\sigma_R^2\boldsymbol{G}_2^T\boldsymbol{D}_L^T\boldsymbol{E}\odot\bar{\boldsymbol{A}}\boldsymbol{D}_L\boldsymbol{G}_2,\\ \boldsymbol{q}_4^H &= \sigma_R^2\breve{\boldsymbol{h}}_2^T\boldsymbol{E}\odot\bar{\boldsymbol{A}}\boldsymbol{D}_L\boldsymbol{G}_2,\\ c_4 &= \sigma_R^2\breve{\boldsymbol{h}}_2^T\boldsymbol{E}\odot\bar{\boldsymbol{A}}\breve{\boldsymbol{h}}_2^* + \sigma_2^2.\end{aligned}$$

In a similar way, constraint (2.10) could be shown equivalent to

$$\begin{pmatrix} \boldsymbol{Q}_3 - \gamma_2 \boldsymbol{Q}_4 + \lambda_3 \boldsymbol{G}_1^H \boldsymbol{G}_1 + \lambda_4 \boldsymbol{G}_2^H \boldsymbol{G}_2 & \boldsymbol{q}_3 - \gamma_2 \boldsymbol{q}_4 \\ \boldsymbol{q}_3^H - \gamma_2 \boldsymbol{q}_4^H & c_3 - \gamma_2 c_4 - \lambda_3 \epsilon_1^2 - \lambda_4 \epsilon_2^2 \end{pmatrix} \succeq 0$$

At last, we address the power constraint in (2.11). Let $\boldsymbol{K}$ be the commutation matrix such that $\mathrm{vec}(\boldsymbol{A^T}) = \boldsymbol{K}\mathrm{vec}(\boldsymbol{A})$. Then according to (2.5), $p_R$ can be rewritten as

$$p_R = c_0 + 2\Re(\boldsymbol{q}_0^H \Delta \boldsymbol{h}) + \Delta \boldsymbol{h}^H \boldsymbol{Q}_0 \Delta \boldsymbol{h},$$

where

$$\begin{aligned} \boldsymbol{Q}_0 &= p_1 \boldsymbol{G}_1^T \boldsymbol{D}_L^T \boldsymbol{E} \odot [\boldsymbol{K}\bar{\boldsymbol{A}}\boldsymbol{K}^T] \boldsymbol{D}_L \boldsymbol{G}_1 \\ &\quad + p_2 \boldsymbol{G}_2^T \boldsymbol{D}_L^T \boldsymbol{E} \odot [\boldsymbol{K}\bar{\boldsymbol{A}}\boldsymbol{K}^T] \boldsymbol{D}_L \boldsymbol{G}_2, \\ \boldsymbol{q}_0^H &= p_1 \breve{\boldsymbol{h}}_1^H \boldsymbol{E} \odot [\boldsymbol{K}\bar{\boldsymbol{A}}\boldsymbol{K}^T] \boldsymbol{D}_L \boldsymbol{G}_1 \\ &\quad + p_2 \breve{\boldsymbol{h}}_2^H \boldsymbol{E} \odot [\boldsymbol{K}\bar{\boldsymbol{A}}\boldsymbol{K}^T] \boldsymbol{D}_L \boldsymbol{G}_2, \\ c_0 &= p_1 \breve{\boldsymbol{h}_1}^T \boldsymbol{E} \odot [\boldsymbol{K}\bar{\boldsymbol{A}}\boldsymbol{K}^T] \breve{\boldsymbol{h}}_1^* \\ &\quad + p_2 \breve{\boldsymbol{h}_2}^T \boldsymbol{E} \odot [\boldsymbol{K}\bar{\boldsymbol{A}}\boldsymbol{K}^T] \breve{\boldsymbol{h}}_1^* + \mathrm{tr}(\bar{\boldsymbol{A}}). \end{aligned}$$

As such, (2.11) is equivalent to

$$\begin{pmatrix} -\boldsymbol{Q}_0 + \kappa_1 \boldsymbol{G}_1^H \boldsymbol{G}_1 + \kappa_2 \boldsymbol{G}_2^H \boldsymbol{G}_2 & -\boldsymbol{q}_0 \\ -\boldsymbol{q}_0^H & t - c_0 - \kappa_1 \epsilon_1^2 - \kappa_2 \epsilon_2^2 \end{pmatrix} \succeq 0.$$

Therefore, the original problem $\mathcal{Q}_1$ can be transformed into the following form.

$$
\begin{aligned}
\mathcal{Q}_2: \quad & \min_{\bar{\boldsymbol{A}},t,\boldsymbol{\lambda},\boldsymbol{\kappa}} \ t \\
\text{s.t.} \quad & \begin{pmatrix} \boldsymbol{Q}_1 - \gamma_1\boldsymbol{Q}_2 + \lambda_1\boldsymbol{G}_1^H\boldsymbol{G}_1 + \lambda_2\boldsymbol{G}_2^H\boldsymbol{G}_1 & \boldsymbol{q}_1 - \gamma_1\boldsymbol{q}_2 \\ \boldsymbol{q}_1^H - \gamma_1\boldsymbol{q}_2^H & c_1 - \gamma_1 c_2 - \lambda_1\epsilon_1^2 - \lambda_2\epsilon_2^2 \end{pmatrix} \succeq 0 \\
& \begin{pmatrix} \boldsymbol{Q}_3 - \gamma_2\boldsymbol{Q}_4 + \lambda_3\boldsymbol{G}_1^H\boldsymbol{G}_1 + \lambda_4\boldsymbol{G}_2^H\boldsymbol{G}_2 & \boldsymbol{q}_3 - \gamma_2\boldsymbol{q}_4 \\ \boldsymbol{q}_1^H - \gamma_2\boldsymbol{q}_2^H & c_3 - \gamma_2 c_4 - \lambda_3\epsilon_1^2 - \lambda_4\epsilon_2^2 \end{pmatrix} \succeq 0 \\
& \begin{pmatrix} -\boldsymbol{Q}_0 + \kappa_1\boldsymbol{G}_1^H\boldsymbol{G}_1 + \kappa_2\boldsymbol{G}_2^H\boldsymbol{G}_2 & -\boldsymbol{q}_0 \\ -\boldsymbol{q}_0^H & t - c_0 - \kappa_1\epsilon_1^2 - \kappa_2\epsilon_2^2 \end{pmatrix} \succeq 0 \\
& \mathrm{Rank}(\bar{\boldsymbol{A}}) = 1, \qquad (2.13)
\end{aligned}
$$

where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_4)$, and $\boldsymbol{\kappa} = (\kappa_1, \kappa_2)$. Since constraint (2.13) is not convex, problem $\mathcal{Q}_2$ is still not a convex problem. Note that, however, if we ignore the rank-one constraint in (2.13), this problem becomes

$$
\begin{aligned}
\mathcal{Q}_3: \quad & \min_{\bar{\boldsymbol{A}},t,\boldsymbol{\lambda},\boldsymbol{\kappa}} \ t \\
\text{s.t.} \quad & \begin{pmatrix} \boldsymbol{Q}_1 - \gamma_1\boldsymbol{Q}_2 + \lambda_1\boldsymbol{G}_1^H\boldsymbol{G}_1 + \lambda_2\boldsymbol{G}_2^H\boldsymbol{G}_1 & \boldsymbol{q}_1 - \gamma_1\boldsymbol{q}_2 \\ \boldsymbol{q}_1^H - \gamma_1\boldsymbol{q}_2^H & c_1 - \gamma_1 c_2 - \lambda_1\epsilon_1^2 - \lambda_2\epsilon_2^2 \end{pmatrix} \succeq 0 \\
& \begin{pmatrix} \boldsymbol{Q}_3 - \gamma_2\boldsymbol{Q}_4 + \lambda_3\boldsymbol{G}_1^H\boldsymbol{G}_1 + \lambda_4\boldsymbol{G}_2^H\boldsymbol{G}_2 & \boldsymbol{q}_3 - \gamma_2\boldsymbol{q}_4 \\ \boldsymbol{q}_1^H - \gamma_2\boldsymbol{q}_2^H & c_3 - \gamma_2 c_4 - \lambda_3\epsilon_1^2 - \lambda_4\epsilon_2^2 \end{pmatrix} \succeq 0 \\
& \begin{pmatrix} -\boldsymbol{Q}_0 + \kappa_1\boldsymbol{G}_1^H\boldsymbol{G}_1 + \kappa_2\boldsymbol{G}_2^H\boldsymbol{G}_2 & -\boldsymbol{q}_0 \\ -\boldsymbol{q}_0^H & t - c_0 - \kappa_1\epsilon_1^2 - \kappa_2\epsilon_2^2 \end{pmatrix} \succeq 0,
\end{aligned}
$$

which is convex and actually a SDP problem. Such a relaxation procedure is called SDP relaxation. In general, the relaxed solution may not be of rank-one. In the next

section, we will discuss how to obtain an approximate rank-one solution from the relaxed problem.

## 2.4 Rank-one Approximation and Outage Condition

The SDP problem $\mathcal{Q}_3$ can be efficiently solved by numerical methods, such as the interior point method [37]. If the resulting solution $\bar{\boldsymbol{A}}^*$ is of rank-one, the optimal beamforming matrix at relay $\boldsymbol{A}^*$ can be easily obtained as $\boldsymbol{A}^* = \text{ivec}(\boldsymbol{a}_{opt})$, where $\boldsymbol{a}_{opt}\boldsymbol{a}_{opt}^H = \bar{\boldsymbol{A}}^*$. However, given the SINR thresholds $\gamma_1$ and $\gamma_2$, problem $\mathcal{Q}_3$ may not be feasible, which leads to outage. In addition, even when the problem $\mathcal{Q}_3$ is feasible, the resulting matrix $\bar{\boldsymbol{A}}^*$ may not be of rank-one. Next, we first propose the rank-one reconstruction approach for the case when $\mathcal{Q}_3$ is feasible.

### *2.4.1 Principle Eigenvector Rank-one Approximation*

Since matrix $\bar{\boldsymbol{A}}^*$ is Hermitian, we have $\bar{\boldsymbol{A}}^* = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^H$. Let $\boldsymbol{A}^* = \text{ivec}(\boldsymbol{u}_1)$, where $\boldsymbol{u}_1$ is the column vector in $\boldsymbol{U}$ that corresponds to the largest eigenvalue. In general, this approach is suboptimal unless $\text{Rank}(\bar{\boldsymbol{A}}^*) = 1$. However, this approach is easy to implement with reasonably good performance, which will be shown in Section 2.5.

### *2.4.2 Outage Consideration: Robust-Hybrid Approach*

Besides the feasibility-caused outage, we have another source of outage if the above principle eigenvector based rank-one approximation leads to a solution that does not satisfy the SINR constraints in (13). This happens more often when the largest eigenvalue is not dominating the others. To reduce the overall outage probability, we propose the following robust-hybrid approach :

1. Solve problem $\mathcal{Q}_3$, if it is feasible, obtain $\boldsymbol{A}^* = \text{ivec}(\boldsymbol{u}_1)$.

2. If problem $\mathcal{Q}_3$ is not feasible, adopt the non-robust approach given in [5].

The robust-hybrid approach is motivated by the observation that the feasibility of the non-robust optimization problem is easier to satisfy given the less number of constraints.

## 2.5 Simulation Results

In the simulation, we make the following assumptions: 1) all nodes have unit noise power; 2) power values $p_1$ and $p_2$ are 10 W; 3) $\hat{\boldsymbol{h}}_i \sim \mathcal{CN}(0, \boldsymbol{I})$, $\Delta \boldsymbol{h}_i \sim \frac{a_i}{\sqrt{M}} e^{j\boldsymbol{\theta}}$, $i = 1, 2$, where $a_i$ is uniformly distributed in $[0, \epsilon_i]$ and $\boldsymbol{\theta}$ is uniformly distributed over $[0, 2\pi]$; and 4) $\epsilon_1 = \epsilon_2$. We compute the outage probability over 1000 channel realizations.

First, we show the relationship between the overall outage probability and the error bound. We compare the hybrid approach against the non-robust approach by assuming that the number of relay antennas are four. As we see in Fig. 2.1, a significantly smaller outage probability is achieved with the robust approach. In particular, when the error bound $\epsilon_1$ and $\epsilon_2$ are large, the outage probability almost remains the same as the error bound changes. When the error bound decreases to a certain critical point, the outage probability decreases significantly with the error bound. To further understand this phenomenon, we decompose the outage probability into two parts. One is the probability that problem $\mathcal{Q}_3$ is infeasible, denoted as $P_1$ and shown by the solid curve with squares in Fig. 2.2. The other is the conditional outage probability $P_2$ given that problem $\mathcal{Q}_3$ is feasible, shown by the solid curve in Fig. 2.2. As we see, $P_1$ increases when error bound increases, since it is more difficult for the robust approach to find feasible solutions given larger channel uncertainty. When the problem is feasible, the conditional outage probability $P_2$ is low compared to the non-robust approach. Interestingly, we see that $P_2$ first increases and then decreases when error bound decreases. The overall outage probability is given as $P_1 + P_2 - P_1 P_2 \approx P_1 + P_2$. As we see from Fig. 2.2, the

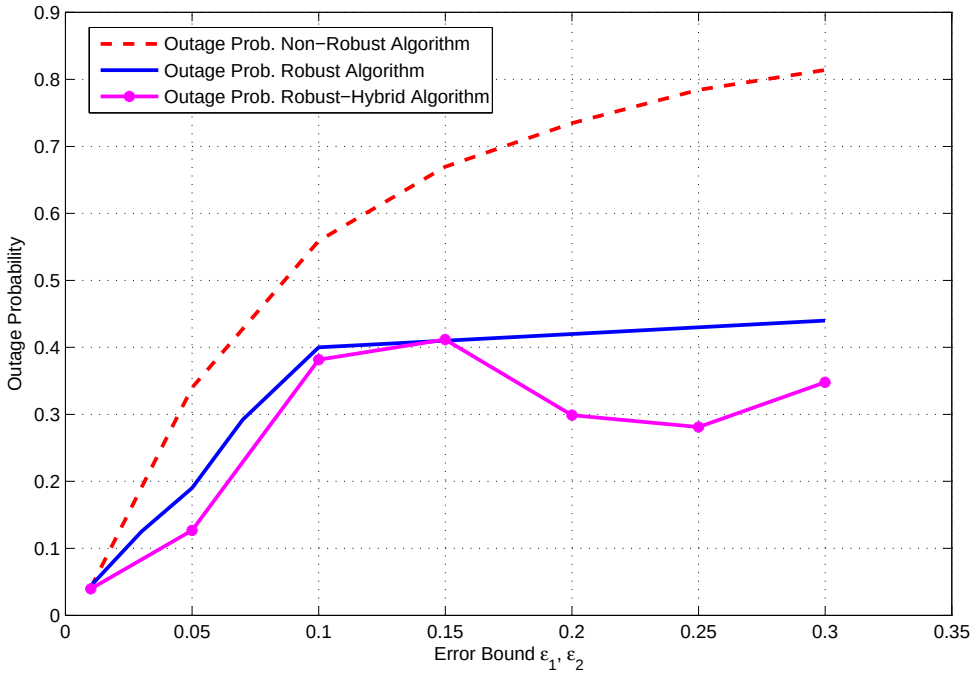


Figure 2.1: Outage Probability vs. Error Bound.

sum of $P_1$ and $P_2$ almost remains constant when the error bound is larger than 0.1. As a result, the overall outage probability almost remains unchanged till the error bound decreases to some critical point, after which the outage probability decreases significantly with the error bound. Fig. 2.3, shows the probability of finding an optimal solution as a function of channel estimation error bound. It is clear that as the channel estimation error bound increases, the probability of obtaining an optimal solution decreases drastically. This highlights the importance of exploring sub-optimal techniques that can provide "acceptable" performance. In addition, in Fig. 2.4, we show the relationship between the outage probability and the number of relay antennas. The errors are bounded at $\epsilon_1 = \epsilon_2 = .01$. As we see, the outage probabilities for both the robust and non-robust approaches decrease. This is due to

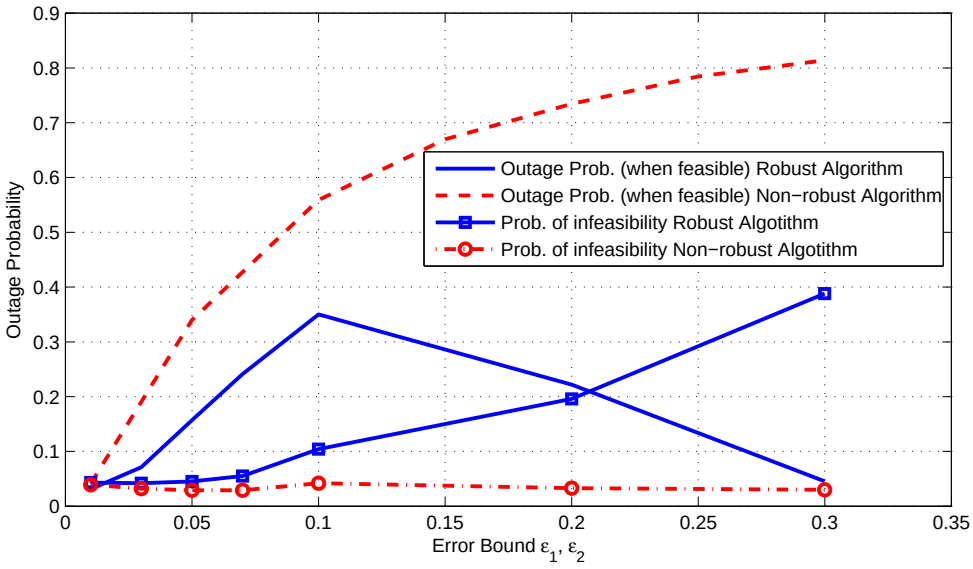


Figure 2.2: Decomposed View of Outage Probability : Conditional Outage Probability and Probability of Infeasibility.

the fact that higher spatial dimension offers a higher degree-of-freedom.

## 2.6 Conclusion

In this chapter we proposed the robust beamforming approach under realistic channel estimation conditions (i.e., with estimation errors) for a TWR system with ANC. By using the S-procedure, the original constraints with infinite dimensions are converted to several LMIs and then the relaxed SDP is solved by applying rank-one relaxation. A principle eigenvector based rank-one reconstruction approach is proposed to reconstruct the solution. To reduce the outage probability, we further proposed a hybrid approach that incorporates the non-robust approach when robust problem formulation is infeasible. Simulations are conducted to show that the proposed hybrid approach leads to significant improvement in terms of outage probability over the non-robust one.

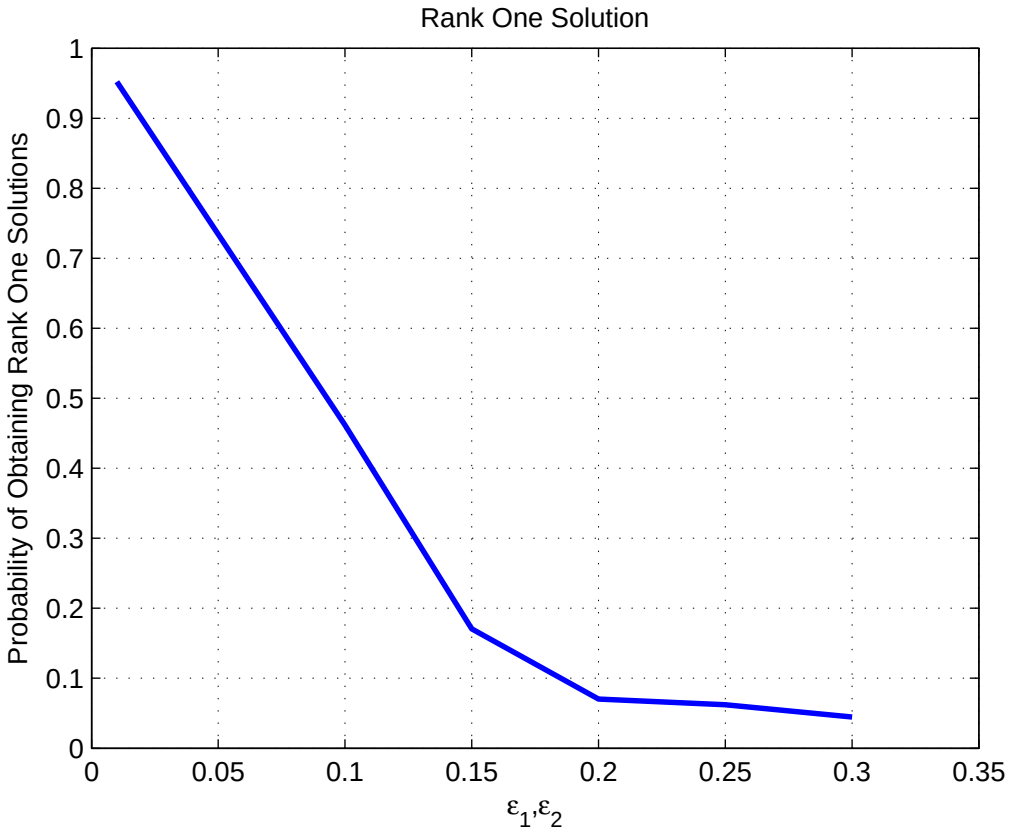


Figure 2.3: Probability of Obtaining Rank-one (optimal) Solution

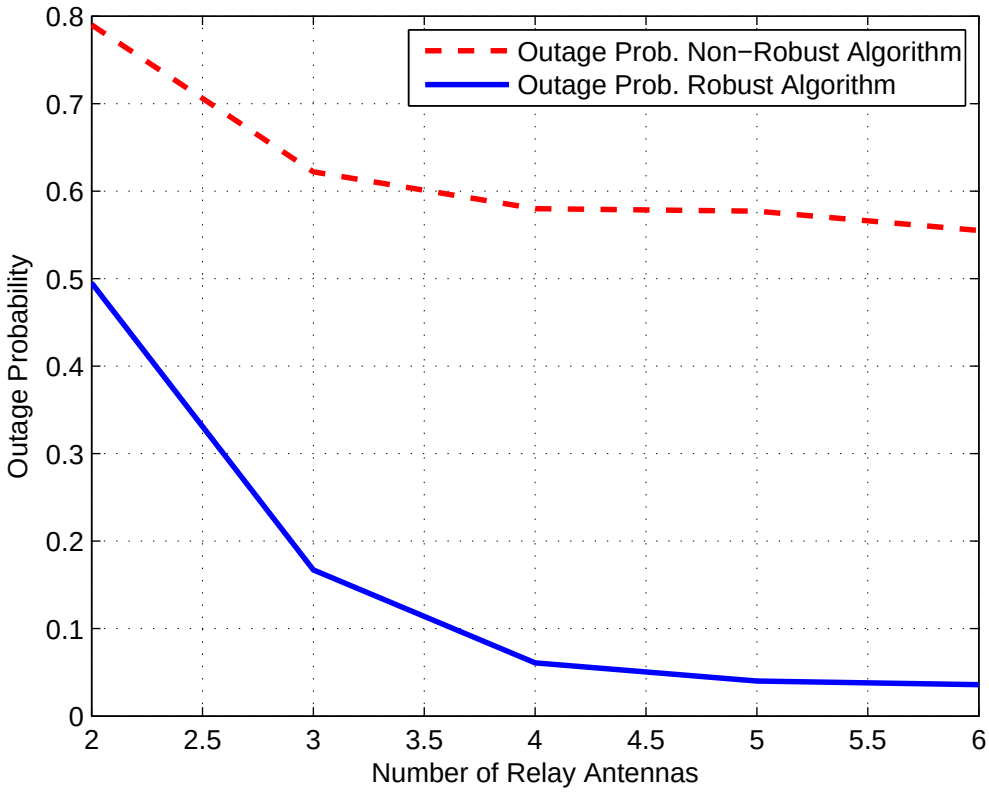


Figure 2.4: Outage Probability vs. Number of Relay Antennas.

# 3. LINEARIZED ROBUST BEAMFORMING

In this chapter, we present a novel robust beamforming method against channel estimation errors for TWR systems. The proposed method obtains a solution of the associated non-convex robust optimization problem by solving a set of closed-form linear equations. Simulations show a considerable performance gain over the rank-one relaxation-based SDP solutions, especially for the cases where the relaxed problem becomes infeasible. In addition, there is significant reduction in complexity, making this method very attractive for practical implementation.

## 3.1 Introduction

In previous chapter, we extended the results of [5] to incorporate channel estimation errors by adopting a robust optimization framework [16]. Similar problems for OWR robust beamforming were studied in [27] and [33] for single-antenna collaborative relays. This was further extended to the case of multiple antennas in [34]. Robust downlink beamforming design for a multiuser multiple input single output (MISO) cognitive radio network where multiple primary users coexist with multiple secondary users was studied in [38]. More recently, robust beamforming design for a TWR system was studied by the authors of [29], where they maximized the minimum SINR between two source nodes subject to a total relay power budget. The non-convex problem was decomposed into a series of relay power minimization problems under minimum SINR constraints by using bisection search. Then the relay power minimization problem was recast into a rank-relaxed SDP and a suboptimal solution was proposed. In their formulation, they ignored the impact of channel estimation error on the cost function (relay power) for the robust optimization problem.

In all these cases the robust optimization framework proposed in [16] was ap-

plied, where the channel estimation error was assumed to be norm-bounded and the optimization problem was solved by convex relaxation. It was observed in [39] that when the non-convex robust optimization problem for TWR was solved with SDP rank-one relaxation, frequently the optimum solution was not available due to either infeasibility or failure of relaxation conditions. In this chapter we explore a new technique that uses the non-robust beamforming solution of [5] as the initial guess and then obtains the robust solution by a first-order perturbation of the initial guess. This approach utilizes the gradients of both the constraint and cost functions along with the worst-case (norm-bounded) channel estimation errors to design the perturbation matrix. The resulting robust QCQP problem is linearized and recast into a LP problem, which is then used to find a beamforming solution under worst-case channel estimation error assumptions. The final beamformimg matrix is then obtained by solving a set of closed-form equations analytically, which significantly reduces the computational complexity, making the proposed method more practical for implementation.

The rest of the chapter is organized as follows. In Section 3.2, we present the system model and formulate the problem. In Section 4.3, we present our approach for transforming the original non-convex problem into a linearized robust optimization problem. Simulation results are presented in Section 3.4 to show the performance improvement and finally we conclude the chapter in Section 3.5.

## 3.2 System Model

We consider a two-way relay system similar to the one introduced in Chapter 2, Fig. 1.2, which consists of the relay node $R$ and two terminal nodes $S1$ and $S2$. In this chapter we introduce the following modifications to the notation for the ease of understanding. The transmit power at the relay $R$ is given by (3.1) and the SINRs

at node $S_i$ are given by $f_i(\boldsymbol{A}, \Delta\boldsymbol{h})$ $(i = 1, 2, \quad j = 3 - i)$ in (3.2)

$$\begin{aligned} G_p(\boldsymbol{A},\Delta\boldsymbol{h}) &= \|\boldsymbol{A}\boldsymbol{h}_1\|^2 p_1 + \|\boldsymbol{A}\boldsymbol{h}_2\|^2 p_2 + \mathrm{Tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \\ &= p_1 \hat{\boldsymbol{h}}_1^H \boldsymbol{A}^H \boldsymbol{A}\hat{\boldsymbol{h}}_1 + p_2 \hat{\boldsymbol{h}}_2^H \boldsymbol{A}^H \boldsymbol{A}\hat{\boldsymbol{h}}_2 + \mathrm{Tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \\ &+ 2\Re(p_1 \hat{\boldsymbol{h}}_1^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_1) + 2\Re(p_2 \hat{\boldsymbol{h}}_2^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_2) \\ &+ p_1 \Delta\boldsymbol{h}_1^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_1 + p_2 \Delta\boldsymbol{h}_2^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_2. \end{aligned} \tag{3.1}$$

$$f_i(\boldsymbol{A}, \Delta\boldsymbol{h}) = \frac{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j + \hat{\boldsymbol{h}}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_j + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j|^2 p_j}{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_i + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_i|^2 p_i + \|(\hat{\boldsymbol{h}}_i^T + \Delta\boldsymbol{h}_i^T)\boldsymbol{A}\|^2\sigma_R^2 + \sigma_i^2}. \tag{3.2}$$

Based on these definitions, the robust optimization problem to minimize the worst-case relay power under worst-case SINR constraints can be formulated as follows.

$$\begin{aligned} & \min_{\boldsymbol{A}} \left[ \max_{\|\Delta\boldsymbol{h}\| \le \epsilon} G_p(\boldsymbol{A}, \Delta\boldsymbol{h}) \right] \\ \text{s.t.} \quad & \min_{\|\Delta\boldsymbol{h}\| \le \epsilon} f_1(\boldsymbol{A}, \Delta\boldsymbol{h}) \ge \gamma_1, \\ & \min_{\|\Delta\boldsymbol{h}\| \le \epsilon} f_2(\boldsymbol{A}, \Delta\boldsymbol{h}) \ge \gamma_2, \end{aligned} \tag{3.3}$$

Where $\gamma_1$ and $\gamma_2$ are the respective SINR targets at $S_1$ and $S_2$. We note that the problem in (3.3) is not convex in general, which renders the globally optimal solution extremely hard to obtain. Therefore, we resort to sub-optimal solutions. The challenges of obtaining a feasible solution for a similar problem were observed in chapter II [39]. In particular, for some specific channel configurations and CSCG noises, the robust TWR beamforming problem of (3.3) has no feasible solutions. In addition, the particular method used for solving the robust optimization problem may fail to obtain a feasible solution. Overall, we call it as a "system outage" when we can not find a feasible solution for (3.3). Therefore, it is desirable to have an efficient

algorithm to solve (3.3), which achieves an "acceptable" performance in terms of power and bears a low outage probability. We will demonstrate the "goodness" of the proposed algorithm by comparing it against the optimal non-robust solution of [5] under perfect channel knowledge, with the same instantaneous channel realization and CSGC noises setup. We also compare the performance of the method presented in this chapter against the results of [5] and [39], under same channel estimation error setup.

## 3.3 Robust Beamforming Using Perturbation

We now propose an algorithm that linearizes the robust optimization problem of (3.3) in the neighborhood of the non-robust optimal solution. Such linearization enables us to convert the non-convex robust optimization problem into a linear optimization problem (under the worst-case channel estimation error assumption), which is then analytically solvable to obtain an optimal perturbation matrix $\boldsymbol{A_p}$. The final sub-optimal beamforming matrix $\tilde{\boldsymbol{A}}$ for (3.3) is the perturbed version of the non-robust beamforming matrix from [5] (by the optimal perturbation matrix $\boldsymbol{A_p}$). Specifically, the proposed algorithm is initialized by $\boldsymbol{A_0}$, the optimal non-robust solution [5], which is give by

$$\begin{aligned} \boldsymbol{A_0} =& arg \min_{\boldsymbol{A}} \left[G_p(\boldsymbol{A}, \boldsymbol{0})\right] \\ \text{s.t.} \quad & f_i(\boldsymbol{A}, \boldsymbol{0}) \geq \gamma_i, \quad i = 1, 2. \end{aligned} \tag{3.4}$$

It is important to note that the method in [5] always provides an optimal solution when the channel is perfectly known. As such, our objective is to perturb the solution $\boldsymbol{A}_0$ in such a way as to increase the relay power as little as possible, while trying to meet the SINR constraints in (3.3) under channel estimation errors.

In particular, after we obtain a non-robust solution for (3.4) using the method

suggested in [5], there are three possible cases that may be easily distinguished:

$$f_1(\boldsymbol{A}_0, 0) = \gamma_1; \quad f_2(\boldsymbol{A}_0, 0) = \gamma_2; \tag{3.5a}$$

$$f_1(\boldsymbol{A}_0, 0) = \gamma_1; \quad f_2(\boldsymbol{A}_0, 0) > \gamma_2; \tag{3.5b}$$

$$f_1(\boldsymbol{A}_0, 0) > \gamma_1; \quad f_2(\boldsymbol{A}_0, 0) = \gamma_2. \tag{3.5c}$$

Before examining these cases, we first introduce some simplified notations to represent the gradient functions

$$\begin{aligned} \vec{\boldsymbol{g}}_{\boldsymbol{A}} = \nabla_{\vec{\boldsymbol{A}}} G_p(\boldsymbol{A}, 0)\big|_{\boldsymbol{A}=\boldsymbol{A_0}}; \vec{\boldsymbol{g}}_{\boldsymbol{h}} = \nabla_{\vec{\boldsymbol{h'}}} G_p(\boldsymbol{A_0}, \boldsymbol{h'})\big|_{\boldsymbol{h'}=0}; \\ \vec{\boldsymbol{f}}_{\boldsymbol{A},i} = \nabla_{\vec{\boldsymbol{A}}} f_i(\boldsymbol{A}, 0)\big|_{\boldsymbol{A}=\boldsymbol{A_0}}; \vec{\boldsymbol{f}}_{\boldsymbol{h},i} = \nabla_{\vec{\boldsymbol{h'}}} f_i(\boldsymbol{A_0}, \boldsymbol{h'})\big|_{\boldsymbol{h'}=0}; \end{aligned} \tag{3.6}$$

where $\vec{\boldsymbol{g}}_{\boldsymbol{A}}, \vec{\boldsymbol{f}}_{\boldsymbol{A},i} \in \mathcal{R}^{2M^2\times 1}$ and $\vec{\boldsymbol{g}}_{\boldsymbol{h}}, \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \in \mathcal{R}^{4M\times 1}$. Next we present the steps for linearizing the optimization problem. The first-order Taylor expansion for the relay power function $G_p(\boldsymbol{A}_0 + \boldsymbol{A_p}, \Delta\boldsymbol{h})$ and the SINR constraint functions $f_i(\boldsymbol{A}_0 + \boldsymbol{A_p}, \Delta\boldsymbol{h})$ under a small channel estimation error $\Delta\boldsymbol{h}$ and a small perturbation matrix $\boldsymbol{A_p} \in \mathcal{C}^{M\times M}$ is given by:

$$\begin{aligned} G_p(\boldsymbol{A}_0 + \boldsymbol{A_p}, \Delta\boldsymbol{h}) \approx G_p(\boldsymbol{A}_0, 0) + \vec{\boldsymbol{A}}_{\boldsymbol{p}} \cdot \vec{\boldsymbol{g}}_{\boldsymbol{A}} + \Delta\vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{g}}_{\boldsymbol{h}}, \\ f_i(\boldsymbol{A}_0 + \boldsymbol{A_p}, \Delta\boldsymbol{h}) \approx f_i(\boldsymbol{A}_0, 0) + \vec{\boldsymbol{A}}_p \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},i} + \Delta\vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i}, \end{aligned}$$

Where,

$$\begin{aligned} \vec{\boldsymbol{A}}_{\boldsymbol{p}} = [vec(\Re(\boldsymbol{A_p})); vec(\Im(\boldsymbol{A_p}))]^T \in \mathcal{R}^{1\times 2M^2}, \\ \Delta\vec{\boldsymbol{h}} = [\Re(\Delta\boldsymbol{h}_1); \Im(\Delta\boldsymbol{h}_1); \Re(\Delta\boldsymbol{h}_2); \Im(\Delta\boldsymbol{h}_2)]^T \in \mathcal{R}^{1\times 4M}. \end{aligned}$$

Note that (3.7) is a good approximation only when $\vec{\boldsymbol{A}}_{\boldsymbol{p}}$ and $\Delta\vec{\boldsymbol{h}}$ are sufficiently small

such that higher-order terms in the Taylor expansions are negligible. Here we do not provide rigorous bounds on the resulting approximation error; however, we later will show from simulations that such approximations are valid. Using the above linear approximations, the problem for choosing the optimal $\boldsymbol{A_p}$ can now be cast as

$$\begin{aligned} \boldsymbol{A_p}^* = arg \min_{\vec{\boldsymbol{A}}_{\boldsymbol{p}}} \left[ \max_{\|\Delta \boldsymbol{h}\| \leq \epsilon} \left[ G_p(\boldsymbol{A}_0, 0) + \vec{\boldsymbol{A}}_{\boldsymbol{p}} \cdot \vec{\boldsymbol{g}}_{\boldsymbol{A}} + \Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{g}}_{\boldsymbol{h}} \right] \right] \\ \text{s.t.} \quad \min_{\|\Delta \boldsymbol{h}\| \leq \epsilon} \left[ f_i(\boldsymbol{A}_0, 0) + \vec{\boldsymbol{A}}_p \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},i} + \Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \right] \geq \gamma_i, i = 1, 2, \end{aligned} \tag{3.7}$$

where $\boldsymbol{A_p}^*$ is the optimal perturbation matrix. Since only the middle term in the cost function of (3.7) depends on $\boldsymbol{A_p}$, it follows that all the other terms in the cost function can be dropped. Similarly the constraints can be rearranged as

$$\max_{\|\Delta \boldsymbol{h}\| \leq \epsilon} \left[ \Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \right] - (f_i(\boldsymbol{A}_0, 0) - \gamma_i) \leq \vec{\boldsymbol{A}}_{\boldsymbol{p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},i}, i = 1, 2. \tag{3.8}$$

Where we used the fact that

$$- \min_{\|\Delta \boldsymbol{h}\| \leq \epsilon} \left[ \Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \right] = \quad \max_{\|\Delta \boldsymbol{h}\| \leq \epsilon} \left[ \Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \right], \tag{3.9}$$

Considering the worst-case channel estimation error bound ($\|\Delta \boldsymbol{h}\| = \epsilon$ ) and applying Cauchy-Schwarz inequality, the first term in the left-hand side of the inequality (3.8) can be replaced by

$$\Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \leq \|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\| \cdot \|\Delta \boldsymbol{h}\|,$$

and simplified using the fact

$$\max_{\|\Delta \boldsymbol{h}\| \leq \epsilon} \left[ \Delta \vec{\boldsymbol{h}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{h},i} \right] = \epsilon \|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\|. \quad i = 1, 2.$$

The optimization problem in (3.7) can then be converted to the following LP problem

$$\begin{aligned} \boldsymbol{A_p}^* &= arg\min_{\vec{\boldsymbol{A_p}}} \left[\vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{g}}_{\boldsymbol{A}}\right] \\ \text{s.t.} \quad & \epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\| - (f_i(\boldsymbol{A}_0, 0) - \gamma_i) \leq \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},i}, \quad i = 1, 2. \end{aligned} \tag{3.10}$$

We now examine the implications of the three cases in (3.5a)-(3.5c). Using our simplified notations, the KKT conditions for the non-robust beamforming solution of problem (3.4) are

$$\vec{\boldsymbol{g}}_{\boldsymbol{A}} = \lambda_1 \vec{\boldsymbol{f}}_{\boldsymbol{A},1} + \lambda_2 \vec{\boldsymbol{f}}_{\boldsymbol{A},2}, \tag{3.11}$$

$$(\gamma_i - f_i(\boldsymbol{A}_0, 0))\lambda_i = 0, \quad i = 1, 2, \tag{3.12}$$

where $\lambda_1, \lambda_2 \geq 0$ are the Lagrange multipliers.

In case of (3.5a) and by substituting (3.11) into the objective function of (3.10), the linearized problem (3.10) becomes

$$\begin{aligned} \boldsymbol{A_p}^* &= arg\min_{\vec{\boldsymbol{A_p}}} \left[\lambda_1 \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},1} + \lambda_2 \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},2}\right] \\ \text{s.t.} \quad & \epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\| \leq \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},i}, \quad i = 1, 2. \end{aligned} \tag{3.13}$$

Since $\lambda_1, \lambda_2 \geq 0$ and the fact that the right-hand side of the constraints in (3.13) also appears in the cost function, implies that the cost function will be minimized as long as both constraints are satisfied with equality. This also follows from basic properties of linear programming.

In case of (3.5b), from (3.12) it follows that $\lambda_1 \geq 0, \lambda_2 = 0$. The corresponding

linearized problem becomes

$$\begin{aligned} \boldsymbol{A_p}^* &= arg\min_{\vec{\boldsymbol{A_p}}}[\lambda_1 \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},1}] \\ \text{s.t.} \quad & \epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\| \leq \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},1}, \\ & \epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\| - (f_2(\boldsymbol{A}_0, 0) - \gamma_2) \leq \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},2}. \end{aligned} \tag{3.14}$$

The cost function in (3.14) is proportional to the right-hand side of the first constraint, such that the cost function will be minimized when the first constraint is satisfied with equality. Furthermore, since $f_2(\boldsymbol{A}_0, 0) - \gamma_2 > 0$, any $\boldsymbol{A_p}$ that satisfies the second constraint in (3.13) will also satisfy the second constraint in (3.14). Similar arguments could be applied to case (3.5c). Therefore to find the solution, we could just focus in the problem in (3.13). By the properties of linear programming, we conclude that any $\boldsymbol{A_p}$ that satisfies

$$\epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\| = \vec{\boldsymbol{A_p}} \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},i}, \quad i = 1, 2, \tag{3.15}$$

is an optimal solution ($\boldsymbol{A_p}^*$) to the linearized problem in cases (3.5a), (3.5b), and (3.5c). It is also observed that there are many possible solutions for $\vec{\boldsymbol{A_p}}$ in (3.15). Specifically, we choose $\vec{\boldsymbol{A_p}}$ to be in the subspace spanned by the gradient vectors $\vec{\boldsymbol{f}}_{\boldsymbol{A},1}$ and $\vec{\boldsymbol{f}}_{\boldsymbol{A},2}$ :

$$\vec{\boldsymbol{A_p}} = c_1 \left[\vec{\boldsymbol{f}}_{\boldsymbol{A},1}\right] + c_2 \left[\vec{\boldsymbol{f}}_{\boldsymbol{A},2}\right]. \tag{3.16}$$

Substituting (3.16) into (3.15), we have

$$\epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},1}\| = c_1\|\vec{\boldsymbol{f}}_{\boldsymbol{A},1}\|^2 + c_2\vec{\boldsymbol{f}}_{\boldsymbol{A},2}^T \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},1}, \tag{3.17}$$

$$\epsilon\|\vec{\boldsymbol{f}}_{\boldsymbol{h},2}\| = c_1\vec{\boldsymbol{f}}_{\boldsymbol{A},1}^T \cdot \vec{\boldsymbol{f}}_{\boldsymbol{A},2} + c_2\|\vec{\boldsymbol{f}}_{\boldsymbol{A},2}\|^2. \tag{3.18}$$

From (3.17) and (3.18), we could solve for $c_1$ and $c_2$ to construct the optimal perturbation matrix $\boldsymbol{A_p}$. Given the perturbation matrix $\boldsymbol{A_p}$, the proposed robust beamforming matrix is obtained as $\tilde{\boldsymbol{A}} = \boldsymbol{A_0} + \boldsymbol{A_p}$.

## 3.4 Simulation Results

We now simulate a TWR system with channel estimation errors, and compare the performance of the proposed beamforming method with those presented in [5] and [39]. We used the true channel in [5] to serve as the upper bound for both the performance metric of outage and power. The setups are $M = 4$, ${\sigma_R}^2 = 1W$, ${\sigma_i}^2 = 1W$; i=1,2, $p_i = 10W; i = 1, 2$, $\gamma_i = 10; i = 1, 2$, and $\epsilon_i = [0.05, 0.4]$ with increments of 0.05. The channel was generated as $\hat{\boldsymbol{h}}_i \sim \mathcal{CN}(0, \boldsymbol{I})$ and the channel estimation error is set as $\Delta \boldsymbol{h}_i \sim \frac{a_i}{\sqrt{M}} e^{j\boldsymbol{\theta}}$, $i = 1, 2$, with$a_i$ uniformly distributed in $[0, \epsilon_i]$ and $\boldsymbol{\theta}$ uniformly distributed over $[0, 2\pi]^M$.

In this setup an outage is declared when any of the SINRs at the source nodes fall below $\gamma_i$. In Fig. (3.1), and (3.2), we respectively plot the outage probability and $95-th$ percentile of the empirical cumulative density function (cdf) [40] of the transmit power required to achieve the corresponding outage performance. The performance of the algorithm in [5] under perfect channel knowledge is indicated by "Perfect-channel" which serves as performance bound. Also, the curve labeled "Non-Robust" indicates the performance of the algorithm in [5] under estimated CSI and the curves labeled "Robust-Hybrid" indicates the performance of the algorithm in [39]. We observe in Fig. (3.1) that significantly smaller outage probabilities are achieved with the linearized robust beamforming approach compared to the ones in [5] and [39]. In particular, when the channel estimation error increases, the outage probability for the robust-hybrid method [39] becomes very large. This can be attributed to the fact that, as the channel estimation error increases, the SDP relaxation approach

fails to find an optimal solution and has to resort to the principal eigenvector based suboptimal solution [39]. In terms of power, it is seen from Fig. (3.2) that our linearized robust method outperforms the robust-hybrid method significantly.

## 3.5 Conclusion

In this chapter we presented a linearized robust beamforming scheme for the TWR system that operates under realistic channel estimation error assumptions. We linearized the non-convex robust optimization problem and analytically obtained the optimal perturbation matrix to perturb the corresponding non-robust optimal beamforming matrix. Simulations showed that the proposed approach led to significant improvement in both outage probability and relay power.

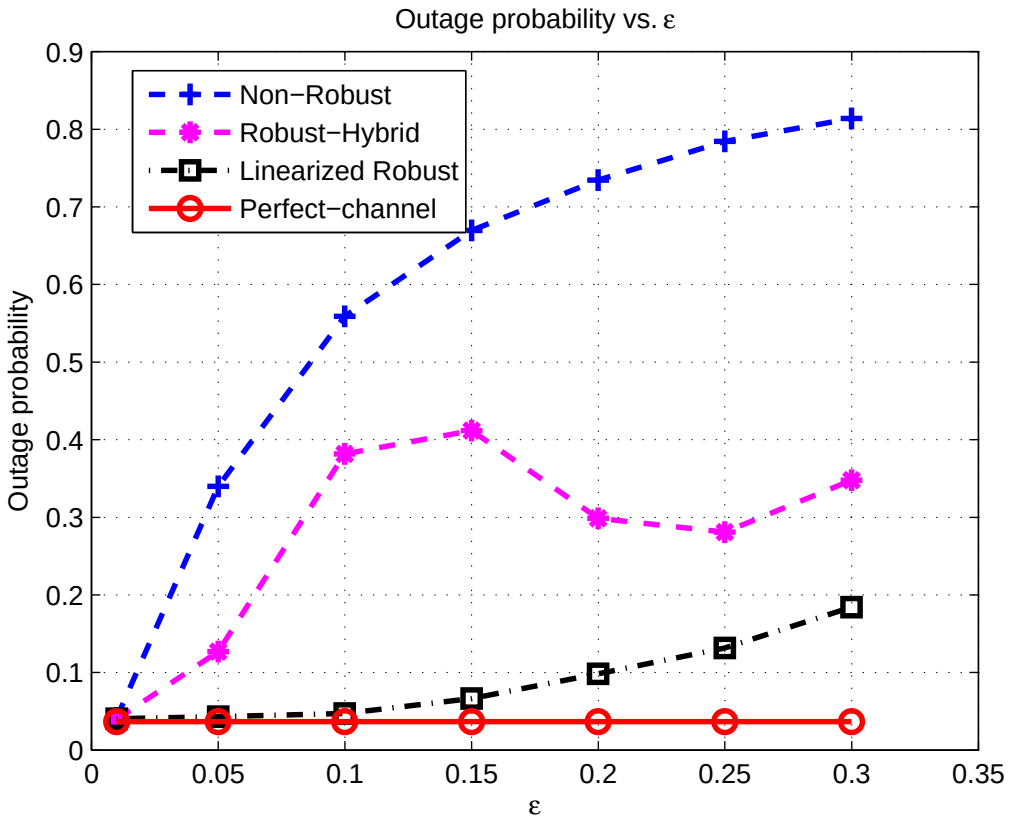


Figure 3.1: Performance of TWR System with Linearized Robust Beamforming Under Channel Estimation Error : Outage vs. Channel Estimation Error.

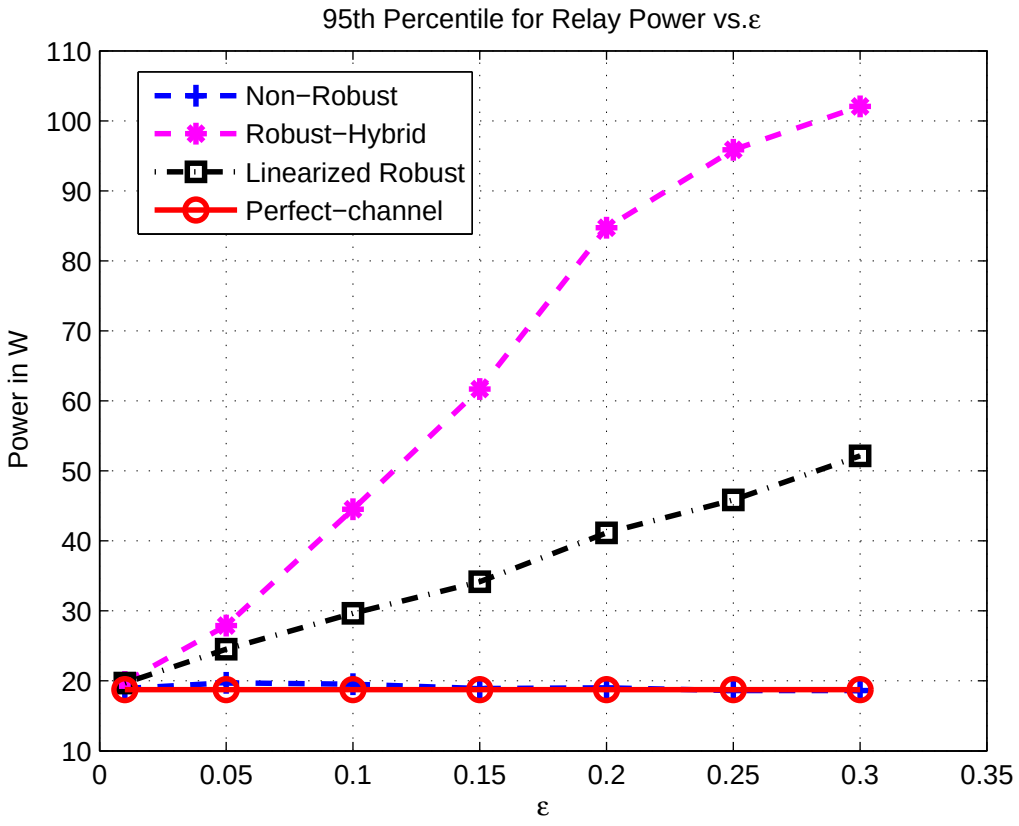


Figure 3.2: Performance of TWR System with Linearized Robust Beamforming Under Channel Estimation Error : Relay Power vs. Channel Estimation Error.

# 4. A PRACTICAL VIEW OF ROBUST BEAMFORMING FOR TWO WAY RELAY SYSTEMS

In this chapter we present a practical view of robust TWR beamforming. A TWR system operating on a hardware target has limited resources and time to compute its beamforming matrix. It is therefore desirable to investigate a method that addresses the problem from this practical stand point that also has an "acceptable" performance in terms of outage probability and relay power requirements. With this motivation, we develop an iterative technique that uses Newton's method and the steepest-descent method (with zero and first order approximations) to solve the TWR robust beamforming problem by a process of alternating between a worst-case CSI error vector search and solving an optimization problem to obtain a beamforming matrix. We assume that the algorithms only operate for an arbitrary number of iterations in order to meet practical real time execution requirements. The algorithm terminates once the KKT conditions for optimality are satisfied or the maximum number of allowable iterations are reached. We provide empirical results to show the performance of the iterative-robust algorithm compared to the techniques presented in the earlier chapters.

## 4.1 Introduction

The importance of considering a robust optimization framework for practical systems which exhibit parameter uncertainty was shown in [16, 25]. In the previous chapters, we presented two different methods for obtaining a robust beamforming solution for a TWR system. The SDP relaxation approach of Chapter 2, provided some useful insights into the upper bound on relay power consumption. The method presented in Chapter 3 extended the optimal non-robust solution [5] to a robust

solution using a low complexity first-order perturbation method that showed very attractive system performance. In the linearized robust method of Chapter 2, it was assumed that the non-robust solution was available. In addition, we assumed that the CSI error was small enough for the linearization to hold.

While both these methods demonstrated very good performance, they both rely on solving an SDP problem using interior point methods. In recent years, there has been some interesting work [35] on looking at the practical implication of implementing such methods on hardware targets such as FGPAs, GPUs and DSPs. Solving these types of problems efficiently with limited resources on a hardware target which also meets the real-time execution requirements remain an area of active research [35]. In a practical system, the size of the problem at hand typically justifies the choice of one solution method over another. Many times the overhead of setting up a more sophisticated computational framework can be significant compared to the problem itself. While we do not compare the relative implementation complexity of our proposed method in this chapter to that of an interior point based one, it is fair to say that the hardware resource utilization for solving a non-convex QCQP with rank relaxed SDP via such a method can be significant [41, 42] on a hardware target.

In this chapter we take a practical view for solving the robust TWR beamforming problem by an iterative approach. The robust optimization problem for TWR beamforming under study has two explicit constrains and typically such a TWR system will have small number of antennas. We have also seen in previous chapters, that the robust TWR beamforming design problem is not convex, therefore obtaining an optimal solution is extremely hard. Therefore, a simple method to solve this problem could very likely outweigh the benefits of faster convergence of an interior point method that is observed for a typical convex problem. Our method is similar in

spirit to [25], where the authors address the robust downlink beamforming problem. They use an iterative method to solve an alternating sequence of optimization and worst-case analysis, where in each step, a convex optimization problem is solved with interior-point method.

At each iteration of the method presented here, we first solve three optimization problems (relay power maximization and two SINR minimizations), each under the norm-bounded channel uncertainty constraint, in order to identify the worst-case CSI's corresponding to a fixed beamforming matrix solution. In the next step, we find a beamforming matrix that achieves the desired outage performance with minimum power for the optimization problem under these worst-case CSI conditions. This process is iterated, until a final beamforming matrix that solves the robust TWR beamforming problem of Chapter 2 is found.

In the following we present a detailed description of our iterative method. We then present simulation results to compare the performance of our proposed method to the others presented in the previous chapters.

## 4.2 System Model

We use the same system model as in Chapter 2, but for completeness we present the relay transmit power and source node SINR equations from the previous chapter and also restate the robust optimization problem under study. The transmit power at the relay R is given by

$$\begin{aligned} G_p(\boldsymbol{A},\Delta\boldsymbol{h}) &= \|\boldsymbol{A}\boldsymbol{h}_1\|^2 p_1 + \|\boldsymbol{A}\boldsymbol{h}_2\|^2 p_2 + \mathrm{Tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \\ &= p_1 \hat{\boldsymbol{h}}_1^H \boldsymbol{A}^H \boldsymbol{A}\hat{\boldsymbol{h}}_1 + p_2 \hat{\boldsymbol{h}}_2^H \boldsymbol{A}^H \boldsymbol{A}\hat{\boldsymbol{h}}_2 + \mathrm{Tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \\ &+ 2\Re(p_1 \hat{\boldsymbol{h}}_1^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_1) + 2\Re(p_2 \hat{\boldsymbol{h}}_2^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_2) \\ &+ p_1 \Delta\boldsymbol{h}_1^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_1 + p_2 \Delta\boldsymbol{h}_2^H \boldsymbol{A}^H \boldsymbol{A}\Delta\boldsymbol{h}_2. \end{aligned} \tag{4.1}$$

We define $f_{SINR_1}(\boldsymbol{A}, \Delta\boldsymbol{h})$ and $f_{SINR_2}(\boldsymbol{A}, \Delta\boldsymbol{h})$ as the SINRs at $S_1$ and $S_2$ respectively, where $f_{SINR_i}$, $i = 1, 2$ are given by

$$f_{SINR_i}(\boldsymbol{A}, \Delta\boldsymbol{h}) = \frac{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j + \hat{\boldsymbol{h}}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_j + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j|^2 p_j}{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_i + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_i|^2 p_i + \|(\hat{\boldsymbol{h}}_i^T + \Delta\boldsymbol{h}_i^T)\boldsymbol{A}\|^2\sigma_R^2 + \sigma_i^2}. \tag{4.2}$$

Based on the definitions of SINR and transmit power, the robust optimization problem can be formulated similar to Chapter 3 as

$$\begin{aligned} & \min_{\boldsymbol{A}} \left[ \max_{\|\Delta\boldsymbol{h}\| \leq \epsilon} G_p(\boldsymbol{A}, \boldsymbol{h}) \right] \\ \text{s.t.} \quad & \min_{\|\Delta\boldsymbol{h}\| \leq \epsilon} f_{SINR_1}(\boldsymbol{A}, \Delta\boldsymbol{h}) \geq \gamma_1 \\ & \min_{\|\Delta\boldsymbol{h}\| \leq \epsilon} f_{SINR_2}(\boldsymbol{A}, \Delta\boldsymbol{h}) \geq \gamma_2, \end{aligned} \tag{4.3}$$

where $\|\Delta\boldsymbol{h}\|^2 \equiv \|\Delta\boldsymbol{h}_1\|^2 + \|\Delta\boldsymbol{h}_2\|^2$ and $\Delta\boldsymbol{h}$ is the CSI error vector. Given the SINR thresholds $\gamma_1$ and $\gamma_2$, the system outage can be defined as the event that results in $f_{SINR_1}(\boldsymbol{A}, \Delta\boldsymbol{h}) < \gamma_1$ and/or $f_{SINR_2}(\boldsymbol{A}, \Delta\boldsymbol{h}) < \gamma_2$, i.e. no feasible solution to problem (4.3) could be found.

It is convenient to re-express the optimization problem in terms of the revised SINR functions (note that $f_i(\boldsymbol{A}, \Delta\boldsymbol{h})$ in this chapter is expressed in a different from Chapter 3)

$$\begin{aligned} f_i(\boldsymbol{A}, \Delta\boldsymbol{h}) \equiv \quad & |\boldsymbol{h}_i^T \boldsymbol{A}\boldsymbol{h}_k + \boldsymbol{h}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_k + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\boldsymbol{h}_k|^2 p_k \\ & -\gamma_i |\boldsymbol{h}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_i + \Delta\boldsymbol{h}_i^T \boldsymbol{A}\boldsymbol{h}_i|^2 p_i \\ & -\gamma_i \sigma_R^2 \|(\boldsymbol{h}_i^T + \Delta\boldsymbol{h}_i^T)\boldsymbol{A}\|^2, \quad i = 1, 2 \quad k = 3 - i. \end{aligned} \tag{4.4}$$

Note that, this formulation reduces the need for divisions in the iterative formulation. Divisions are expensive in a hardware target. In terms of $f_1(\boldsymbol{A}, \Delta\boldsymbol{h})$ and $f_2(\boldsymbol{A}, \Delta\boldsymbol{h})$,

the optimization problem becomes

$$\begin{aligned} & \min_{\boldsymbol{A}} \left[ \max_{\|\Delta \boldsymbol{h}\| \leq \epsilon} G_p(\boldsymbol{A}, \boldsymbol{h}) \right] \\ \text{s.t.} \quad & \min_{\|\Delta \boldsymbol{h}\| \leq \epsilon} f_1(\boldsymbol{A}, \Delta \boldsymbol{h}) \geq 0 \\ & \min_{\|\Delta \boldsymbol{h}\| \leq \epsilon} f_2(\boldsymbol{A}, \Delta \boldsymbol{h}) \geq 0. \end{aligned} \tag{4.5}$$

## 4.3 Robust Beamforming Using an Iterative Method

We now propose an iterative method to solve the non-convex robust TWR beamforming problem (4.5). In Chapter 3, it was observed that the linearized robust beamforming solution works very well in practice, as long as the channel estimation error is moderate and an optimal non-robust solution is available to serves as an initial guess. In contrast the method presented in this chapter does not require previous computation of an optimal non-robust solution. Instead, we initialize the algorithm using a zero-forcing solution similar to the one proposed in [39].

The iterative technique can be primarily broken down into the following steps:

1. Initialize the algorithm with a beamforming matrix $\boldsymbol{A}$ obtained using the zero-forcing method proposed in [5].

2. Set $\epsilon$ the radius of the error sphere according to design requirements.

3. For the beamforming matrix $\boldsymbol{A}$, find three "worst case" CSI error vectors $\Delta\hat{\boldsymbol{h}}_{G_p}$, $\Delta\hat{\boldsymbol{h}}_{f_1}$ and $\Delta\hat{\boldsymbol{h}}_{f_2}$, that maximize the relay transmit power and minimize the two source node SINRs, respectively. (This step is referred to as "Stage I" in the subsequent description of the algorithm.)

4. Perturb the beamforming matrix $\boldsymbol{A}$ to find a new beamforming matrix $\bar{\boldsymbol{A}}$ that satisfies the target SINR at the two source nodes $S_1$ and $S_2$, and also minimizes

the transmit power at the relay node under the three worst-case CSI error vector $\Delta\hat{\boldsymbol{h}}_{G_p}$, $\Delta\hat{\boldsymbol{h}}_{f_1}$ and $\Delta\hat{\boldsymbol{h}}_{f_2}$ (this step is referred to as "Stage II" in the detailed algorithm description).

5. Check if $\|\vec{\bar{\boldsymbol{A}}} - \vec{\boldsymbol{A}}\| < \tau$, where $\tau$ is an arbitrarily chosen small constant. If this check returns true then we conclude a feasible and "acceptable" beamforming solution to (4.5) is found. Otherwise, $\boldsymbol{A}$ is set to $\bar{\boldsymbol{A}}$ and the algorithm returns to step-3 and continues until $\|\bar{\boldsymbol{A}} - \boldsymbol{A}\| < \tau$ is satisfied. (this step is referred to as "Stage III" in the detailed algorithm description.)

6. If $\|\bar{\boldsymbol{A}} - \boldsymbol{A}\| < \tau$ is not satisfied after a predefined number of iteration, then the design requirement on epsilon is relaxed (i.e. $\epsilon$ is reduced) and the algorithm returns to Step-3 and the iteration continues until a feasible solution to (4.3) is found.

In Stage-I, the three worst-case CSI error vectors $\Delta\hat{\boldsymbol{h}}_{G_p}$, $\Delta\hat{\boldsymbol{h}}_{f_1}$ and $\Delta\hat{\boldsymbol{h}}_{f_2}$ are specified by the following three optimization problems:

$$\Delta\hat{\boldsymbol{h}}_{G_p} = arg\max_{\Delta\boldsymbol{h}} [G_p(\boldsymbol{A}, \Delta\boldsymbol{h})] \tag{4.6a}$$

$$\text{s.t.} \quad \|\Delta\boldsymbol{h}\| \leq \epsilon,$$

$$\Delta\hat{\boldsymbol{h}}_{f_1} = arg\min_{\Delta\boldsymbol{h}} [f_1(\boldsymbol{A}, \Delta\boldsymbol{h})] \tag{4.6b}$$

$$\text{s.t.} \quad \|\Delta\boldsymbol{h}\| \leq \epsilon,$$

$$\Delta\hat{\boldsymbol{h}}_{f_2} = arg\min_{\Delta\boldsymbol{h}} [f_2(\boldsymbol{A}, \Delta\boldsymbol{h})] \tag{4.6c}$$

$$\text{s.t.} \quad \|\Delta\boldsymbol{h}\| \leq \epsilon.$$

We now prove that the inequalities in (4.6a, 4.6b and 4.6c) can all be replaced with equalities; in other words, the worst case CSI error vector that maximizes the

power function (4.1) and the two worst case CSI error vectors that minimize the two SINR functions in (4.4), all lie on the surface of the CSI error sphere. The proof for this proceeds as follows.

- To establish equality in (4.6a), we observe from (4.1) that the expression for $G_p$ is quadratic in $\Delta\boldsymbol{h}$, and the second-order terms $p_1\Delta\boldsymbol{h}_1^H\boldsymbol{A}^H\boldsymbol{A}\Delta\boldsymbol{h}_1 + p_2\Delta\boldsymbol{h}_2^H\boldsymbol{A}^H\boldsymbol{A}\Delta\boldsymbol{h}_2$ depend on the hermitian positive semidefinite form $\boldsymbol{A}^H\boldsymbol{A}$. It follows that $G_p$ as a function of $\Delta\boldsymbol{h}$ has no local maxima. Thus, the maximum of $G_p$ on the set $\{\|\Delta\boldsymbol{h}\| \leq \epsilon\}$ cannot occur in the interior of $\Delta\boldsymbol{h}$ sphere i.e $\{\|\Delta\boldsymbol{h}\| < \epsilon\}$, it occurs on the surface of the sphere i.e $\{\|\Delta\boldsymbol{h}\| = \epsilon\}$.

- To establish equality in (4.6b) and (4.6c), we observe that the function $f_i(\boldsymbol{A}, \Delta\boldsymbol{h})$ depends on $\Delta\boldsymbol{h}_j$ through a single term in the numerator. It follows that $f_i(\boldsymbol{A}, \Delta\boldsymbol{h})$ as a function of $\Delta\boldsymbol{h}_j$, is a constant on a hyperplane $\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\Delta\boldsymbol{h}_j = \zeta$ where $\zeta \in \mathcal{C}$. Any such plane that passes through the interior of the $\Delta\boldsymbol{h}$ sphere i.e $\|\Delta\boldsymbol{h}\| < \epsilon$ will also intersect the sphere at the boundary $\|\Delta\boldsymbol{h}\| = \epsilon$ (since sphere is bounded and the plane is unbounded). This implies that, any value which $f_{SINR_i}(\boldsymbol{A}, \Delta\boldsymbol{h})$ assumes in the interior of the $\Delta\boldsymbol{h}$ sphere, it also assumes the same value on the boundary of the sphere. In particular, the minimum value of $f_i(\boldsymbol{A}, \Delta\boldsymbol{h})$ on $\|\Delta\boldsymbol{h}\| \leq \epsilon$ must be assumed over the boundary of the $\Delta\boldsymbol{h}$ sphere.

Hence equality in (4.6a, 4.6b and 4.6c) has been established. For convenience of

reference, we restate the optimization problems with equality:

$$\Delta\hat{\boldsymbol{h}}_{G_p} = arg\max_{\Delta\boldsymbol{h}} [G_p(\boldsymbol{A}, \Delta\boldsymbol{h})] \tag{4.7a}$$

$$\text{s.t.} \quad \|\Delta\boldsymbol{h}\| = \epsilon,$$

$$\Delta\hat{\boldsymbol{h}}_{f_1} = arg\min_{\Delta\boldsymbol{h}} [f_1(\boldsymbol{A}, \Delta\boldsymbol{h})] \tag{4.7b}$$

$$\text{s.t.} \quad \|\Delta\boldsymbol{h}\| = \epsilon,$$

$$\Delta\hat{\boldsymbol{h}}_{f_2} = arg\min_{\Delta\boldsymbol{h}} [f_2(\boldsymbol{A}, \Delta\boldsymbol{h})] \tag{4.7c}$$

$$\text{s.t.} \quad \|\Delta\boldsymbol{h}\| = \epsilon.$$

Before proceeding further, we introduce the following simplified notations for the gradient functions

$$\begin{aligned} \vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}} &\equiv \nabla_{\vec{\Delta\boldsymbol{h}}} G_p(\boldsymbol{A}, \Delta\boldsymbol{h}); \\ \vec{f}_{\Delta\boldsymbol{h},i} &\equiv \nabla_{\vec{\Delta\boldsymbol{h}}} f_i(\boldsymbol{A}, \Delta\boldsymbol{h}) \quad i = 1, 2. \end{aligned} \tag{4.8}$$

Where $\Delta\boldsymbol{h} \in \mathcal{C}^{1\times 2M}$ and $\vec{\Delta\boldsymbol{h}} \equiv [\Re(\Delta\boldsymbol{h}_1); \Im(\Delta\boldsymbol{h}_1); \Re(\Delta\boldsymbol{h}_2); \Im(\Delta\boldsymbol{h}_2)]^T \in \mathcal{R}^{1\times 4M}$ is the CSI error vector. Also, $\vec{\boldsymbol{A}} = [vec(\Re(\boldsymbol{A})); vec(\Im(\boldsymbol{A}))]^T \in \mathcal{R}^{1\times 2M^2}$ is the beamforming matrix $\boldsymbol{A} \in \mathcal{C}^{M\times M}$ represented as a vector with real entries. In the following we provide detailed description for the three main stages of the algorithm.

### *4.3.1 Stage I: Worst-case CSI Error Vector*

Throughout this stage, the matrix $\boldsymbol{A}$ is assumed to be fixed. Starting with a beamforming matrix $\boldsymbol{A}$, this stage of the algorithm finds three "worst-case" CSI errors $\Delta\hat{\boldsymbol{h}}_{G_p}$, $\Delta\hat{\boldsymbol{h}}_{f_1}$ and $\Delta\hat{\boldsymbol{h}}_{f_2}$ corresponding to the optimization problems (4.7a) for relay power and (4.7b,4.7c) for the SINR functions. We adopt Newton's method to compute $\Delta\hat{\boldsymbol{h}}_{G_p}$, and steepest-descent methods to compute $\Delta\hat{\boldsymbol{h}}_{f_1}$ and $\Delta\hat{\boldsymbol{h}}_{f_2}$. In the

following we present Stage-I of our algorithm in details.

**Worst-case CSI error vector for relay power** In this section we will discuss how Newton's method (with lowest non-vanishing order i.e zero or first order) is used to obtain the worst-case CSI error vector associated with the relay power. For ease of representation we use $\Delta\boldsymbol{h} = [\Delta\boldsymbol{h}_1;\ \Delta\boldsymbol{h}_2] \in \mathcal{C}^{1\times 2M}$ to represent $\Delta\hat{\boldsymbol{h}}_{G_p}$. Also, $\Delta\boldsymbol{h}(k)$ and $\Delta\boldsymbol{h}(k+1)$ are used to represent an estimate of $\Delta\hat{\boldsymbol{h}}_{G_p}$ at the $k$'th and $k+1$'th iteration of the following algorithm.

**Algorithm Initialization:** The iterative TWR beamforming algorithm is initialized with a beamforming matrix $\boldsymbol{A}$, obtained by the zero-forcing method proposed in [5]. We use, $\epsilon\vec{\boldsymbol{g}}_{\boldsymbol{h}}/\|\vec{\boldsymbol{g}}_{\boldsymbol{h}}\|$ and $\epsilon\vec{\boldsymbol{f}}_{\boldsymbol{h},i}/\|\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\|$ (with i=1,2) as initial guess for $\Delta\hat{\boldsymbol{h}}_{G_p}$ and $\Delta\hat{\boldsymbol{h}}_{f_i}$ (with i=1,2) respectively. These guesses correspond to the linearized approximations of gradient of $G_p(\boldsymbol{A}, \Delta\boldsymbol{h})$ and $f_i(\boldsymbol{A}, \Delta\boldsymbol{h})$ with respect to $\Delta\boldsymbol{h}$ (Appendix A). During subsequent executions of Stage-I, the initialization step is omitted, and updated value of $\boldsymbol{A}$ obtained in Stage-III and previously computed value of $\Delta\hat{\boldsymbol{h}}_{G_p}$ and $\Delta\hat{\boldsymbol{h}}_{f_i}$ (with i=1,2) are used to start the iteration for Stage-I.

**Subsequent Iterations:** Using our notations defined above, we assume that $\Delta\boldsymbol{h}(k)$ approximates $\Delta\hat{\boldsymbol{h}}_{G_p}$ at the $k$'th iteration of the algorithm, such that $\Delta\hat{\boldsymbol{h}}_{G_p} \approx \Delta\boldsymbol{h}(k) + \boldsymbol{\delta}$ where $\boldsymbol{\delta} \in \mathcal{C}^{1\times 2M}$ is a small perturbation. By replacing $\Delta\boldsymbol{h}$ in (4.7a) by $\Delta\boldsymbol{h}(k)+\boldsymbol{\delta'}$, where $\boldsymbol{\delta'} \in \mathcal{C}^{1\times 2M}$ is an independent variable defined by $\boldsymbol{\delta'} \equiv \Delta\hat{\boldsymbol{h}}_{G_p} - \Delta\boldsymbol{h}(k)$ and considering $\Delta\boldsymbol{h}(k)$ to be a constant in each iteration, the optimization problem (4.7a) for finding $\Delta\hat{\boldsymbol{h}}_{G_p}$ can be recast in terms of the $\boldsymbol{\delta}$ as: find $\Delta\boldsymbol{h}(k) + \boldsymbol{\delta}$ where $\boldsymbol{\delta}$ is the

solution to

$$\begin{aligned} \vec{\boldsymbol{\delta}} &= arg \max_{\vec{\boldsymbol{\delta}}'} [G_p(\boldsymbol{A}, \Delta\boldsymbol{h}(k) + \boldsymbol{\delta}')] \\ \text{s.t.} \quad & \|\Delta\boldsymbol{h}(k) + \boldsymbol{\delta}'\|^2 = \epsilon^2. \end{aligned} \tag{4.9}$$

Where $\vec{\boldsymbol{\delta}} = [\Re(\boldsymbol{\delta_1});\ \Im(\boldsymbol{\delta_1});\ \Re(\boldsymbol{\delta_2});\ \Im(\boldsymbol{\delta_2})] \in \mathcal{R}^{1\times 4M}$. The KKT condition of the optimization problem in (4.9) along with the constraint is given by

$$\vec{\boldsymbol{g}}_{\Delta\boldsymbol{h}(k)+\boldsymbol{\delta}} = \lambda(\Delta\vec{\boldsymbol{h}}(k) + \vec{\boldsymbol{\delta}}) \tag{4.10a}$$

$$\|\Delta\vec{\boldsymbol{h}}(k) + \vec{\boldsymbol{\delta}}\|^2 = \epsilon^2. \tag{4.10b}$$

A value for $\vec{\boldsymbol{\delta}}$ can be obtained by solving (4.10a) and (4.10b) iteratively. Once $\vec{\boldsymbol{\delta}}$ is found in each iteration, an improved estimate of $\Delta\hat{\boldsymbol{h}}_{G_p}$ can be obtained by $\Delta\hat{\boldsymbol{h}}_{G_p} = \Delta\boldsymbol{h}(k) + \boldsymbol{\delta}$. In order to solve the above set of equation we linearize the KKT condition and the constraint (4.10a and 4.10b by using approximations to the lowest non-vanishing order). The Lagrange multiplier in (4.10a) is approximated to lowest order (zero'th order) by $\lambda'$ and obtained by taking the inner product of both sides of (4.10a) with $\Delta\vec{\boldsymbol{h}}(k)$ and solving for "approximate" value of $\lambda$ denoted by $\lambda'$ (approximated to the zero'th order in $\vec{\boldsymbol{\delta}}$ i.e. in particular $\vec{\boldsymbol{\delta}} \approx 0$) as follows

$$\lambda' \equiv \frac{\Delta\vec{\boldsymbol{h}}(k)^T \cdot \vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}(k)}}{\Delta\vec{\boldsymbol{h}}(k)^T \cdot \Delta\vec{\boldsymbol{h}}(k)}. \tag{4.11}$$

Note that the exact value of $\lambda$ is given by $\lambda \equiv \Delta\lambda + \lambda'$, where $\Delta\lambda$ is the estimation error and we assume $\Delta\lambda << \lambda'$.

Next, using the fact that $\|\Delta\boldsymbol{h}(k)\| = \epsilon$, equation (4.10b) can be approxi-

mated to the lowest order (first order in this case assuming higher order terms are negligible) as follows:

$$\begin{aligned} &\|\vec{\boldsymbol{\delta}}\|^2 + 2\Delta\vec{\boldsymbol{h}}(k)\cdot\vec{\boldsymbol{\delta}} = 0 \\ \implies &\Delta\vec{\boldsymbol{h}}(k)\cdot\vec{\boldsymbol{\delta}} \approx 0 \\ \implies &\Delta\vec{\boldsymbol{h}}_1(k)\cdot\vec{\boldsymbol{\delta}}_1 + \Delta\vec{\boldsymbol{h}}_2(k)\cdot\vec{\boldsymbol{\delta}}_2 \approx 0. \end{aligned} \tag{4.12}$$

Now, using the gradient function $\vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}}$ given in Appendix A, with $\Delta\vec{\boldsymbol{h}} = \Delta\vec{\boldsymbol{h}}(k) + \boldsymbol{\delta}$, the KKT condition in (4.10a) can be expanded and written as

$$\begin{aligned} &2\left[p_1\overrightarrow{\left(\boldsymbol{A}_0^H\boldsymbol{A}_0(\boldsymbol{h}_1 + \Delta\boldsymbol{h}_1(k) + \boldsymbol{\delta}_1)\right)}\ ;\ p_2\overrightarrow{\left(\boldsymbol{A}_0^H\boldsymbol{A}_0(\boldsymbol{h}_2 + \Delta\boldsymbol{h}_2(k) + \boldsymbol{\delta}_2)\right)}\right] \\ &= \lambda[\Delta\vec{\boldsymbol{h}}_1(k) + \vec{\boldsymbol{\delta}}_1\ ;\ \Delta\vec{\boldsymbol{h}}_2(k) + \vec{\boldsymbol{\delta}}_2]. \end{aligned} \tag{4.13}$$

Next, replacing $\lambda$ with $\lambda' + \Delta\lambda$ in (4.13), and retaining only lowest order terms (first order) in $\vec{\boldsymbol{\delta}}$ and $\Delta\lambda$ we obtain

$$\begin{aligned} &2p_1\text{vec}\left(\boldsymbol{A}_0^H\boldsymbol{A}_0\boldsymbol{\delta}_1\right) - \lambda'\vec{\boldsymbol{\delta}}_1 - (\Delta\lambda)\Delta\vec{\boldsymbol{h}}_1(k) = \\ &\qquad\qquad \lambda'\Delta\vec{\boldsymbol{h}}_1(k) - 2p_1\text{vec}\left(\boldsymbol{A}_0^H\boldsymbol{A}_0(\boldsymbol{h}_1 + \Delta\boldsymbol{h}_1(k))\right), \\ &2p_2\text{vec}\left(\boldsymbol{A}_0^H\boldsymbol{A}_0\boldsymbol{\delta}_2\right) - \lambda'\vec{\boldsymbol{\delta}}_2 - (\Delta\lambda)\Delta\vec{\boldsymbol{h}}_2(k) = \\ &\qquad\qquad \lambda'\Delta\vec{\boldsymbol{h}}_2(k) - 2p_2\text{vec}\left(\boldsymbol{A}_0^H\boldsymbol{A}_0(\boldsymbol{h}_2 + \Delta\boldsymbol{h}_1(k))\right). \end{aligned} \tag{4.14}$$

Equations (4.14) can be rearranged into $4N+1$ real equations, which can than be solved for the $4N+1$ real unknowns $\Re[\boldsymbol{\delta}_1], \Im[\boldsymbol{\delta}_1], \Re[\boldsymbol{\delta}_2], \Im[\boldsymbol{\delta}_2]$ and $\Delta\lambda$. Using the solution of $\vec{\boldsymbol{\delta}}$ from above, we obtain $\Delta\boldsymbol{h}(k) + \boldsymbol{\delta}$ which is an improved estimate for $\Delta\hat{\boldsymbol{h}}_{G_p}$. However, we know from (4.7a) that $\Delta\hat{\boldsymbol{h}}_{G_p}$ has a norm equal to $\epsilon$, so we apply re-scaling to ensure that our estimate meets this condition. The final expression for $\Delta\boldsymbol{h}(k+1)$ (improved

estimate of $\Delta\hat{\boldsymbol{h}}_{G_p}$ at $k+1$'th iteration) is given by

$$\Delta\boldsymbol{h}(k+1) = \epsilon\frac{\Delta\boldsymbol{h}(k)+\boldsymbol{\delta}}{\|\Delta\boldsymbol{h}(k)+\boldsymbol{\delta}\|}. \tag{4.15}$$

The process described above is iterated until $\|\Delta\boldsymbol{h}(k+1)-\Delta\boldsymbol{h}(k)\| < \tau$, where $\tau$ is an arbitrary chosen small tolerance value.

**Worst-case CSI error vector for the destination SINRs:** In this section of the algorithm, we will show how the CSI error vector $\Delta\hat{\boldsymbol{h}}_{f_i}$ associated with the worst-case SINR value at source nodes $S_i$ (where $i = 1, 2$) are estimated using the steepest descent method. Once again, for ease of representation, we will use $\Delta\boldsymbol{h}_i \in \mathcal{C}^{M\times 1}$ to represent $\Delta\hat{\boldsymbol{h}}_{f_i}$. Also, $\Delta\boldsymbol{h}_i(k)$ and $\Delta\boldsymbol{h}_i(k+1)$ are used to represent an estimate of $\Delta\hat{\boldsymbol{h}}_{f_i}$ at the $k$'th and $k+1$'th iteration of the following algorithm.

**Initialization:** The initial guess for the worst-case CSI error vector is given by

$$\Delta\hat{\boldsymbol{h}}_{f_i}(0) = \epsilon\frac{\nabla_{\vec{\boldsymbol{h}'}} f_i(\boldsymbol{A},\boldsymbol{h'})\big|_{\boldsymbol{h'}=0}}{\|\nabla_{\vec{\boldsymbol{h}'}} f_i(\boldsymbol{A},\boldsymbol{h'})\big|_{\boldsymbol{h'}=0}\|}. \quad i = 1, 2$$

The exact expression for the gradient function $\nabla_{\vec{\boldsymbol{h}'}} f_i(\boldsymbol{A},\boldsymbol{h'})\big|_{\boldsymbol{h'}=0}$ is given in Appendix A.

**Subsequent Iteration:** The estimate for the worst-case CSI related to the SINR function at the $k$'th iteration is denoted by $\Delta\boldsymbol{h}_i(k)$. The normalized gradient $\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h_i}(k)} \in \mathcal{C}^{2M\times 1}$ at the $k$'th iteration is given by

$$\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h_i}(k)} \equiv \frac{\nabla_{\boldsymbol{\Delta h_i}(k)} f_i(\boldsymbol{A},\boldsymbol{\Delta h_i}(k))}{\|\nabla_{\boldsymbol{\Delta h_i}(k)} f_i(\boldsymbol{A},\boldsymbol{\Delta h_i}(k))\|} \quad i = 1, 2. \tag{4.16}$$

The exact expression for the gradient function $\nabla_{\boldsymbol{\Delta h_i}(k)} f_i(\boldsymbol{A},\boldsymbol{\Delta h_i}(k))$ is given in Appendix A. Using the normalized gradient values obtained

above and the current estimate of the CSI error vector, $\boldsymbol{\Delta h_i}(k)$, we evaluate three new linear estimates of CSI error vectors associated with each destination node SINR as follows

$$\Delta\hat{\boldsymbol{h}}_{\ell,i} = \epsilon \frac{\left[\boldsymbol{\Delta h_i}(k) - \ell\mu\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h_i}(k)}\right]}{\|\boldsymbol{\Delta h_i}(k) - \ell\mu\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h_i}(k)})\|}; \quad \ell = 0, 1, 2, \quad i = 1, 2, \quad (4.17)$$

and $\mu$ is a fixed (small) parameter.

Using the three CSI error estimates from above and three corresponding values of the SINR $f_i(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{\ell,i})$, we apply quadratic interpolation to obtain a better estimate for the CSI error vector. This is done by fitting a parabola through the three points $(\ell, f_i(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{\ell,i}))$, and finding the $x$ coordinate where the parabola's vertex is located. Using this $x$ coordinate, an interpolated estimate for the CSI error vector is obtained as follows

$$\Delta\hat{\boldsymbol{h}}_{3,i} = \epsilon \frac{\left[\boldsymbol{\Delta h_i}(k) - x\mu\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h_i}(k)}\right]}{\|\boldsymbol{\Delta h_i}(k) - x\mu\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h_i}(k)}\|}, \quad i = 1, 2. \quad (4.18)$$

Given four CSI error vector estimates, the best value for the CSI error vector $\Delta\hat{\boldsymbol{h}}_{\ell^*,i}$ is chosen from $\{\Delta\hat{\boldsymbol{h}}_{0,i}, \Delta\hat{\boldsymbol{h}}_{1,i}, \Delta\hat{\boldsymbol{h}}_{2,i}, \Delta\hat{\boldsymbol{h}}_{3,i}\}$ such that $f_i(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{\ell^*,i})$ is minimized. Finally $\Delta\boldsymbol{h}_i(k+1)$ (the improved estimate of $\Delta\hat{\boldsymbol{h}}_{f_i}$ at iteration $k+1$)is set to $\Delta\hat{\boldsymbol{h}}_{l^*,i}$. The iteration continues until $\|\Delta\boldsymbol{h}_i(k+1) - \Delta\boldsymbol{h}_i(k)\|$ is less than some fixed tolerance.

### *4.3.2 Stage-II: Optimal Beamforming Matrix Search Under Worst-case CSI Error Vectors*

After identifying the three estimates of worst-case CSI error vectors $\Delta\hat{\boldsymbol{h}}_{G_p}$, $\Delta\hat{\boldsymbol{h}}_{f_1}$ and $\Delta\hat{\boldsymbol{h}}_{f_2}$ corresponding to a beamforming matrix $\boldsymbol{A}$, the next step is to find a beamforming matrix $\bar{\boldsymbol{A}}$ that solves the following optimization problem

$$\begin{aligned} \bar{\boldsymbol{A}} =& arg\min_{\boldsymbol{A}} \left[ G_p(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{G_p}) \right] \\ \text{s.t.} \quad & f_i(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{f_i}) \geq \gamma_i, \quad i = 1, 2. \end{aligned} \tag{4.19}$$

Starting with an initial guess for the beamforming matrix $\boldsymbol{A}(0) = \boldsymbol{A}$, the iterative search proceeds to find $\bar{\boldsymbol{A}} \in \mathcal{C}^{M\times M}$, the solution for the optimization problem (4.19). The solution of the beamforming matrix at the $m$'th iteration of this stage is given by $\boldsymbol{A}(m) \in \mathcal{C}^{M\times M}$. The $m$'th estimate of the beamforming matrix $\boldsymbol{A}(m)$, can give rise to two conditions : (a) $\boldsymbol{A}(m)$ is feasible (i.e both SINR constraints are met) but not optimal (that is, the power is not minimized) (b) $\boldsymbol{A}(m)$ is infeasible (that is, the two SINR constraints are not met). In the following, we describe how the algorithm proceeds for these two conditions

1. If $\boldsymbol{A}(m)$ is feasible but not optimal, we proceed to reduce the power by a fixed (small) percentage while remaining feasible. We obtain the next estimate of the beamforming matrix $\boldsymbol{A}(m+1)$ at the $m+1$'th iteration by perturbing the beamforming matrix by a perturbation matrix $\boldsymbol{A_p}$ that is parallel to the steepest-descent direction of the function $G_p(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{G_p})$.

2. If $\boldsymbol{A}(m)$ is not feasible, then we obtain $\boldsymbol{A}(m+1)$ by adding a perturbation $\boldsymbol{A}_p \in \mathcal{C}^{M\times M}$ along a direction that is perpendicular to the steepest-descent direction with respect to $G_p(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{G_p})$ the power gradient, but at the same time, in a

direction which improves both the SNR (values of functions $f_1(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{f_1})$ and $f_2(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{f_2})$) towards feasibility. If it is not possible to improve the worst-case SINR's (within tolerance) while remaining perpendicular to the gradient of the power function at the relay node, then the perpendicularity condition is temporarily relaxed.

Finally for both of the above two cases, the $m+1$'th estimate of the beamforming matrix is given by $\boldsymbol{A}(m+1) = \boldsymbol{A}(m) + \boldsymbol{A}_p$. In order to compute the optimal beamforming solution for (4.19) using steepest descent method, we represent all the matrices in vector forms i.e. $\vec{\boldsymbol{A}} = [vec(\Re(\boldsymbol{A})); vec(\Im(\boldsymbol{A}))]^T \in \mathcal{R}^{1\times 2M^2}$ and $\vec{\boldsymbol{A}}_p = [vec(\Re(\boldsymbol{A}_p)); vec(\Im(\boldsymbol{A}_p))]^T \in \mathcal{R}^{1\times 2M^2}$ and in addition to that, we use the following gradient vectors (all are real $2N^2 \times 1$ vectors)

$$\begin{aligned} \vec{\boldsymbol{g}}_{\boldsymbol{A}(m)} &\equiv \nabla_{\vec{\boldsymbol{A}}} G_p(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{G_p})\Big|_{\boldsymbol{A}=\boldsymbol{A}(m)}, \\ \vec{\boldsymbol{f}}_{i,\boldsymbol{A}(m)} &\equiv \nabla_{\vec{\boldsymbol{A}}} f_i(\boldsymbol{A}, \Delta\hat{\boldsymbol{h}}_{f_i})\Big|_{\boldsymbol{A}=\boldsymbol{A}(m)}, \ i = 1, 2. \end{aligned} \tag{4.20}$$

An explicit expression for $\vec{\boldsymbol{g}}_{\boldsymbol{A}(m)}$ and $\vec{\boldsymbol{f}}_{i,\boldsymbol{A}(m)}$ are given in Appendix A.

During the search for $\vec{\boldsymbol{A}}_p$, the perpendicularity condition to the steepest-descent direction of the power gradient $\vec{\boldsymbol{g}}_m$ is imposed by choosing a perturbation $\vec{\boldsymbol{A}}_p$ such that

$$\vec{\boldsymbol{g}}_m \cdot \vec{\boldsymbol{A}}_p = 0; \qquad \vec{\boldsymbol{f}}_{i,\boldsymbol{A}(m)} \cdot \vec{\boldsymbol{A}}_p = \xi_i, \ i = 1, 2.$$

where

$$\xi_i \equiv \gamma_i - f_i(\boldsymbol{A}(m), \Delta\hat{\boldsymbol{h}}_{f_i}) \qquad (i = 1, 2),$$

and $\xi_i$ represents the "shortfalls" between the minimum target SINRs and the SINRs achieved with worst-case CSI error vectors $\Delta\hat{\boldsymbol{h}}_{f_i}$, $(i = 1, 2)$. Based on the above

mentioned conditions, $\vec{\boldsymbol{A}}_p$ is obtained by

$$\vec{\boldsymbol{A}}_p = \Gamma(\Gamma^T\Gamma)^{-1}[0\ \xi_1\ \xi_1]^T,$$

where

$$\Gamma \equiv [\vec{\boldsymbol{g}}_m\ \vec{\boldsymbol{f}}_{1,\boldsymbol{A}(m)}\ \vec{\boldsymbol{f}}_{2,\boldsymbol{A}(m)}] \in \mathcal{R}^{2M^2\times 3}.$$

Thus, a small (first order) perturbation of $\vec{\boldsymbol{A}}(m)$ along the $\vec{\boldsymbol{A}}_p$ direction will produce almost negligible increase in power, and will increase the two SINRs in the same ratio as their respective shortfalls.

When perpendicularity condition is not imposed, the direction of the perturbation $\vec{\boldsymbol{A}}_p$ is found by

$$\vec{\boldsymbol{A}}_p = \Gamma(\Gamma^T\Gamma)^{-1}[\xi_1\ \ \xi_2]^T,$$

where

$$\Gamma \equiv [\vec{\boldsymbol{f}}_{1,\boldsymbol{A}(m)}\ \vec{\boldsymbol{f}}_{2,\boldsymbol{A}(m)}] \in \mathcal{R}^{2M^2\times 2}.$$

In this case, we have

$$\vec{\boldsymbol{f}}_{i,\boldsymbol{A}(m)} \cdot \vec{\boldsymbol{A}}_p = \xi_i,\ \ i = 1, 2.$$

Once again, a small (first order) perturbation of $\vec{\boldsymbol{A}}(m)$ along the $\vec{\boldsymbol{A}}_p$-direction will produce small increase in power but it will increase the two SINRs in the same ratio as their respective shortfalls.

The magnitude of $\vec{\boldsymbol{A}}_p$ is determined by extrapolation as follows. Two trial perturbations are defined as

$$\vec{\boldsymbol{A}}_{p,\ell} = \ell\mu\frac{\vec{\boldsymbol{A}}_p}{\|\vec{\boldsymbol{A}}_p\|}, \qquad \ell = 1, 2,$$

where $\mu << 1$ is a small parameter. Then, two interpolating parabolas $\wp_i$ (one for each SINR function with $i = 1, 2$) are computed through the three corresponding points

$$\left(0, f_i\left(\boldsymbol{A}(m), \Delta\hat{\boldsymbol{h}}_{f_i}\right)\right), \left(1, f_i\left(\boldsymbol{A}(m) + \boldsymbol{A}_{p,1}, \Delta\hat{\boldsymbol{h}}_{f_i}\right)\right) \textit{and} \left(2, f_i\left(\boldsymbol{A}(m) + \boldsymbol{A}_{p,2}, \Delta\hat{\boldsymbol{h}}_{f_i}\right)\right).$$

If $\wp_i$ attains a $y$-value of $\xi_i$, then $x_i$ is defined as the corresponding $x$-value. Otherwise, $x_i$ is defined as the vertex of $\wp_i$. In terms of $x_i$ $(i = 1, 2)$, the perturbation $\vec{\boldsymbol{A}}_p$ is finally defined as

$$\vec{\boldsymbol{A}}_p \equiv x\vec{\boldsymbol{A}}_{p,l}, \text{ where } x \equiv \begin{cases} x_2 & \text{if } \xi_1 \leq 0; \\ x_1 & \text{if } \xi_1 > 0 \text{ and } \xi_2 \leq 0; \\ \min(x_1, x_2) & \text{if } \xi_1 > 0 \text{ and } \xi_2 > 0, \end{cases}$$

and finally $\boldsymbol{A}(m+1) = \boldsymbol{A}(m) + \boldsymbol{A}_p$. The iterative search for the beamforming matrix $\bar{\boldsymbol{A}}$ terminates if the KKT conditions for (4.19) are met within tolerance for a feasible solution, or if maximum number of allowable iterations $M_{BF}$ is reached. In either case, $\bar{\boldsymbol{A}}$ is set to the final value of the beamforming matrix $\boldsymbol{A}$ obtained in this stage.

### *4.3.3 Stage III - Finding Robust Beamforming Solution for TWR System*

In the final stage (Stage-III) of the algorithm we check for convergence of the algorithm by computing

$$\|\vec{\bar{\boldsymbol{A}}} - \vec{\boldsymbol{A}}\| \leq \tau \tag{4.21}$$

where $\vec{\bar{\boldsymbol{A}}}$ is the latest updated value of the beamforming matrix and $\vec{\boldsymbol{A}}$ is the "old" (prior to the update) beamforming matrix and $\tau$ is an arbitrary chosen small tolerance value. If this condition is satisfied the algorithm terminates. If this condition is

not satisfied we check if the maximum allowable number of iteration $N_{optBF}$ has been reached. If we have reached the maximum number of iteration and we have not found a solution, we then lower the worst-case design requirement, i.e. set $\epsilon = \epsilon/\epsilon_{step}$ and iterate through the full algorithm (Stage -I,II and III) until (4.21) is met and a feasible solution to (4.5) is found. If no feasible solution can be obtained at $\epsilon = \epsilon/\epsilon_{step}$, the relaxation in $\epsilon$ will continue until we approximately reach the non-robust problem ($\epsilon \approx 0$); which we have seen in the previous chapters, presents the highest chance of providing a feasible solution. Note that when we relax the design requirements, we are relying on best effort transmission. On the contrary, when (4.21) is satisfied, the KKT condition for the TWR robust beamforming problem holds by the virtue of the fact that the beamforming matrix that satisfied KKT condition in Stage-II of the algorithm did not change and hence the condition remains valid in this final stage as well, implying we have obtained a locally optimal solution.

## 4.4 Simulation Results

We now simulate a TWR system with channel estimation errors, and compare the performance of the proposed beamforming method in this chapter with those presented in Chapters II and III. We used the true channel in [5] to serve as the upper bound for both the performance metric of outage and power. The setups are $M = 4$, ${\sigma_R}^2 = 1W$, ${\sigma_i}^2 = 1W$; i=1,2, $p_i = 10W; i = 1, 2$, $\gamma_i = 10; i = 1, 2$, and $\epsilon_i = [0.05, 0.4]$ with increments of 0.05. The channel was generated as $\hat{\boldsymbol{h}}_i \sim \mathcal{CN}(0, \boldsymbol{I})$ and the channel estimation error is set as $\Delta\boldsymbol{h}_i \sim \frac{a_i}{\sqrt{M}} e^{j\boldsymbol{\theta}}$, $i = 1, 2$, with $a_i$ uniformly distributed in $[0, \epsilon_i]$ and $\boldsymbol{\theta}$ uniformly distributed over $[0, 2\pi]^M$.

In this setup an outage is declared when any of the SINRs at the source nodes fall below $\gamma_i$. In Fig. (4.1), and (4.2), we plot the outage probability and $95-th$ percentile of the empirical cumulative density function (cdf) [40] of the transmit power required

to achieve the corresponding outage performance. The performance of the iterative-

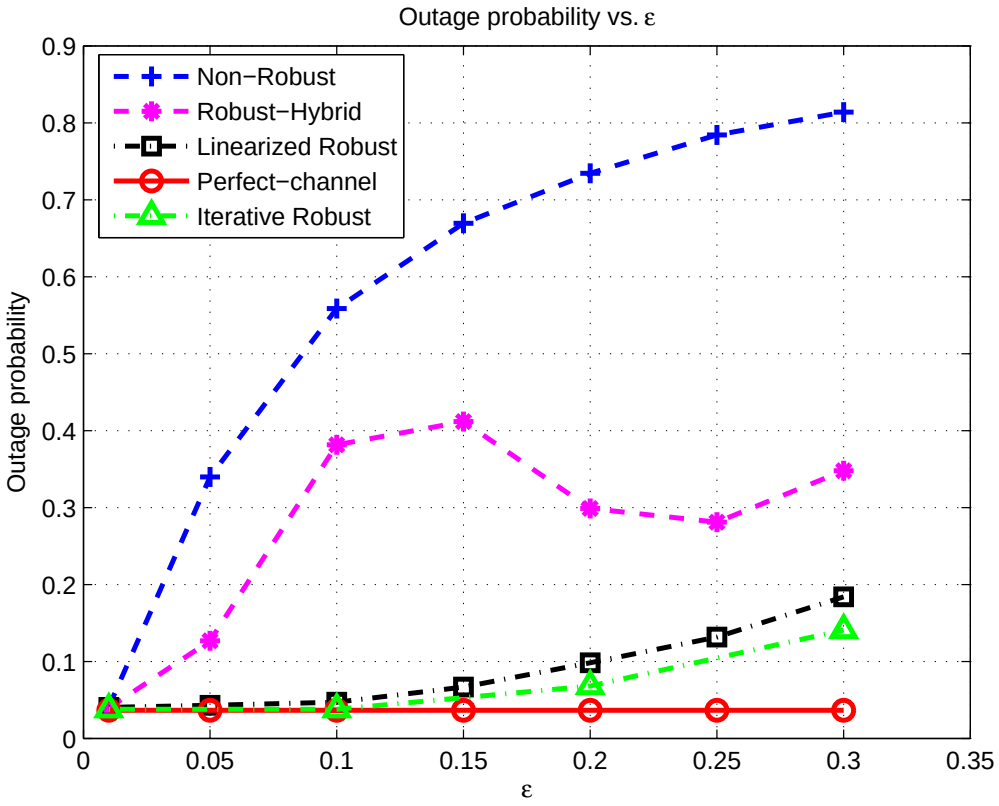


Figure 4.1: Performance of TWR System with Iterative Robust Beamforming Under Channel Estimation Error : Outage vs. Channel Estimation Error.

robust beamforming algorithm presented in this chapter is indicated as "Iterative Robust" in Fig. (4.1) and Fig. (4.2). The performance of the algorithm in [5] under perfect channel knowledge is indicated by "Perfect-channel" which serves as performance bound. Also, the curve labeled "Non-Robust" indicates the performance of the algorithm in [5] under estimated CSI and the curves labeled "Robust-Hybrid" indicates the performance of the algorithm in [39]. In Fig. (4.1) and Fig. (4.2), curve labeled "Linearizer Robust" refers to the performance of the algorithm presented in

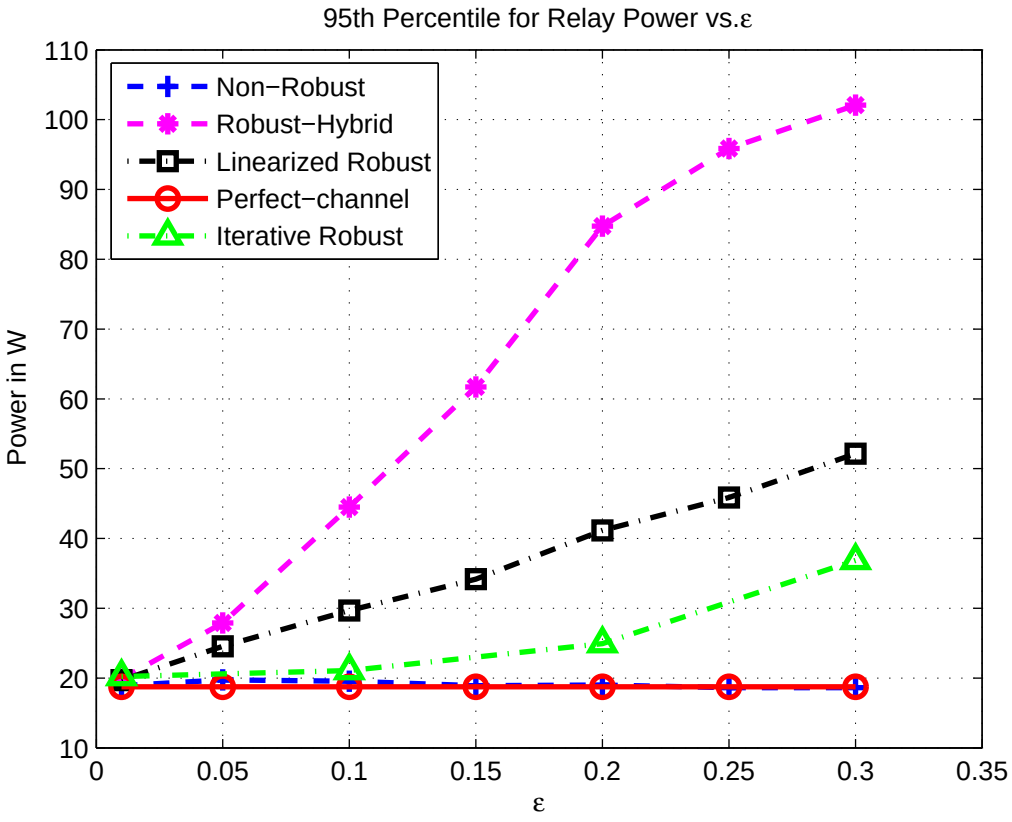


Figure 4.2: Performance of TWR System with Iterative Robust Beamforming Under Channel Estimation Error : Relay Power vs. Channel Estimation Error.

chapter III. We observe in Fig. (4.1) that significantly smaller outage probabilities are achieved with the "iterative robust" beamforming approach compared with the ones presented in previous chapters. In particular, when the channel estimation error increases, the outage probability for the iterative method outperforms all the other methods. This can be attributed to the fact that, as the channel estimation error increases, the SDP relaxation approach fails to find an optimal solution and has to resort to the principal eigenvector based suboptimal solution [39]. Similarly for the linearized-robust method the assumption of small CSI error for the linearization to hold, starts to deviate further and this causes loss in performance. In the case of "iterative-robust" method, since the worst-case CSI error is estimated, we do not have to make any assumption about the magnitude of estimation error $\epsilon$ for the linearization to hold. In terms of power, it is seen from Fig. (4.2) that the iterative-robust method outperforms both the robust-hybrid method and the linearized-robust method significantly.

## 4.5 Conclusion

In this chapter we presented an iterative-robust beamforming scheme for a TWR system that operates under realistic channel estimation error assumptions. Our algorithm solves a series of optimization problems (relay power maximization and two SINR minimization), each under a norm-bounded channel uncertainty constraint in order to identify the worst-case CSI corresponding to a common fixed beamforming matrix. In the next step of the algorithm, a new beamforming matrix is found that minimizes the relay transmit power and achieve the desired outage performance under these three worst-case CSI conditions. This process of estimating the worst-case CSI and finding a beamforming matrix is alternated, until a final beamforming matrix that satisfies the KKT condition of the robust TWR beamforming problem

under worst-case CSI is found. Simulations showed that the proposed approach led to significant improvement in both outage probability and relay power. Since the "iterative robust" method estimates the worst-case CSI at the relay node, that information can be used to draw some conclusion about the probability of a meeting a target SINR. In a practical system, the ability of estimating a priori (at the relay node) the probability of meeting a target SINR can have very significant impact in terms of the overall system performance. This is because, the relay can decide on a transmission strategy (e.g higher transmission power, different modulation/coding schemes, adding transmission redundancy such as HARQ (Hybrid Automatic Repeat Request) etc.) without waiting for an ACK/NACK (acknowledgement) from the receiver. This type of performance estimate at the transmitter side can also simplify the amount of feedback information needed from the receiver side. We point out that, the performance gain of the iterative-robust method over the linearized-robust method of Chapter 3, comes at the expense of added computational complexity, especially if we only consider the linearized solution and we do not consider the computation of the non-robust beamforming solution. Therefore, combining the two method can reduce the complexity of the over all algorithm further.

# 5. CONCLUSION AND FUTURE WORK

In this dissertation, we studied the design of an optimal beamforming solution for a TWR system equipped with multiple antennas at the relay node and a single antenna at each source node under channel estimation errors. We followed the robust optimization framework [16], where the channel estimation errors are assumed to be norm bounded. We designed a beamforming solution for a TWR system that minimizes the worst-case relay transmit power under the constraints of meeting predefined SINR targets at the two source nodes. This optimization problem is a non-convex QCQP with infinitely many constraints, making it extremely hard to obtain a globally optimal solution. In Chapter 2, we presented a design method where we used the S-procedure and converted the problem into a SDP with three LMIs. The SDP problem with rank-one relaxation was solved using standard SDP solvers [43,44]. Finally, the beamforming solution was obtained using the principal eigenvector based reconstruction. We observed that for some cases, the optimization problem was infeasible and in some other cases the solution was not optimal due to failure of rank-one condition. To overcome these difficulties, we proposed a hybrid-robust approach based on the best effort principle that combined the robust and non-robust beamforming solutions.

In Chapter 3, we proposed a linearized robust beamforming method. This method used perturbation and worst-case analysis techniques to design a robust beamforming solution. We designed an optimal perturbation matrix that was used to perturb the optimal non-robust TWR beamforming solution of [5]. The sub-optimal robust solution obtained by this method showed significant improvement for relay transmit power and system outage probability compared to the SDP method at a considerably

lower complexity. When compared with the optimal non-robust solutions with no CSI errors, the linearized robust solution (under CSI error) showed only moderate increase in relay transmit power requirements and outage probability.

In Chapter 4, we introduced an iterative technique for TWR robust beamforming that is more suitable for practical hardware implementation. In the iterative method, we solved a series of three optimization problems (relay power maximization, and two SINR minimization), each under the norm-bounded channel uncertainty constraint to identity the worst-case CSI corresponding to a beamforming matrix solution. In the next step we found a beamforming matrix that achieved the desired performance under these worst-case CSI condition. This process was alternated, until a final beamforming matrix that solved the robust TWR beamforming problem is found. This method showed the best performance amongst all the methods we considered.

In our current treatment of the robust beamforming, we started from a QCQP with infinitely many constraints and using S-procedure we converted that to SDP. An alternative way of treating these types of problems were proposed by Bertsimas & Sim [45]. In their treatment, the robust counterpart of any problem exhibits the same structure, i.e., the robust counterpart of a QCQP is also a QCQP with more constraints and variables. It would be interesting to assess the performance of this new treatment.

Another interesting extension from a practical standpoint would be to explore a hybrid algorithm, that combines the linearized robust method and iterative robust method to further reduce complexity. The practical view of robust beamforming for systems with large numbers of antenna arrays can be very beneficial topic to study for future wireless systems. Finally it will be very interesting to implement these techniques on a wireless testbed and obtain more insight into the problem. This experience can fuel new research topics.

# REFERENCES


[1] G. L. Stuber, *Principles of Mobile Communication (2nd Edition)*. Boston: Kluwer Academic Publishers, 2001.

[2] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: Analog networking coding." Computer Science and Artificial Intelligence Laboratory Technical Report, MIT-CSAIL-TR-2007-012, Feb. 2007.

[3] P. Popovski and H. Yomo, "Physical network coding in two-way wireless relay channels," in *IEEE International Conference on Communications (ICC)*, pp. 707–712, 2007.

[4] B. Rankov and A. Wittneben, "Spectral efficient protocols for half-duplex fading relay channels," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 2, pp. 379–389, 2007.

[5] R. Zhang, Y.-C. Liang, C. Choy, and S. Cui, "Optimal beamforming for two-way multi-antenna relay channel with analogue network coding," *IEEE J. Sel. Areas Commun.*, vol. 27, pp. 699–712, Jun. 2009.

[6] M. P. Wilson, K. Narayanan, H. D. Pfister, and A. Sprintson, "Joint physical layer coding and network coding for bidirectional relaying," *Information Theory, IEEE Transactions on*, vol. 56, no. 11, pp. 5641–5654, 2010.

[7] L. C. Godara, "Applications of antenna arrays to mobile communications I. performance improvement, feasibility, and systemconsiderations," *Proc. IEEE.*, vol. 85, no. 7, pp. 1029 – 1060, 1997.

[8] L. C. Godara, "Applications of antenna arrays to mobile communications II. beamfroming and direction of arrival considerations," *Proc. IEEE.*, vol. 85, no. 8,

pp. 1193 – 1245, 1997.

[9] R. A. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*. New York: John Wiley & Sons, 1980.

[10] B. D. V. Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Mag.*, vol. 5, no. 2, pp. 4 – 24, 1988.

[11] F. Ono, P. Mitran, H. Poor, and V. Tarokh, "Collaborative beamforming for distributed wireless ad hoc sensor networks," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4110–4124, 2005.

[12] S. Zhou and G. Giannakis, "How accurate channel prediction needs to be for transmit-beamforming with adaptive modulation over rayleigh mimo channels?," *IEEE Transactions on Wireless Communications*, vol. 3, no. 4, pp. 1285–1294, 2004.

[13] J.-J. van de Beek, O. Edfors, M. Sandell, S. Wilson, and P. Ola Borjesson, "On channel estimation in ofdm systems," in *IEEE Vehicular Technology Conference*, vol. 2, pp. 815–819 vol.2, 1995.

[14] D. Love, R. Heath, V. K. N. Lau, D. Gesbert, B. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 8, pp. 1341–1365, 2008.

[15] Y. Isukapalli, R. Annavajjala, and B. Rao, "Performance analysis of transmit beamforming for miso systems with imperfect feedback," *IEEE Transactions on Communications*, vol. 57, no. 1, pp. 222–231, 2009.

[16] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski, *Robust Optimization*. Princeton, NJ: Princeton University Press, 2009.

[17] A. Ben-Tal and A. Nemirovski, "Robust convex optimization," *Mathematics of Operations Research.*, vol. 23, No.4, pp. 769–1024, November 1998.

[18] R. T. Rockafellar, *Convex Analysis*. Princeton, NJ: Princeton University Press, 1970.

[19] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Method in Convex Programming*. Society for Industrial and Applied Mathematics, http://epubs.siam.org/doi/book/10.1137/1.9781611970791, 1994.

[20] B. O. Mats Bengtsson, "Transmit beamforming for physical-layer multicasting," vol. 37, pp. 987 – 996, September 1999.

[21] A. Gershman, N. Sidiropoulos, S. Shahbazpanahi, M. Bengtsson, and B. Ottersten, "Convex optimization-based beamforming," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 62–75, 2010.

[22] B. Chalise, S. Shahbazpanahi, A. Czylwik, and A. Gershman, "Robust downlink beamforming based on outage probability specifications," *IEEE Transactions on Wireless Communications*, vol. 6, no. 10, pp. 3498–3503, 2007.

[23] M. Shenouda and T. Davidson, "Convex conic formulations of robust downlink precoder designs with quality of service constraints," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 714–724, 2007.

[24] E. Karipidis, N. Sidiropoulos, and Z.-Q. Luo, "Far-field multicast beamforming for uniform linear antenna arrays," *IEEE Transactions on Signal Processing*, vol. 55, no. 10, pp. 4916–4927, 2007.

[25] A. Mutapcic, S. Kim, and S. Boyd, "A tractable method for robust downlink beamforming in wireless communications," in *Conference Record of the Forty-*

*First Asilomar Conference on Signals, Systems and Computers, 2007. ACSSC 2007.*, pp. 1224–1228, 2007.

[26] P. Ubaidulla and A. Chockalingam, "Robust distributed beamforming for wireless relay networks," in *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications,*, pp. 2345–2349, 2009.

[27] G. Zheng, K. Wong, A. Paulraj, and B. Ottersten, "Robust collaborative relay beamforming," *IEEE Transactions on Signal Processing*, vol. 57, no. 8, pp. 3130–3143, 2009.

[28] E. Gharavol and E. Larsson, "Robust joint optimization of mimo two-way relay channels with imperfect csi," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pp. 1657–1664, 2011.

[29] M. Tao and R. Wang, "Robust relay beamforming for two-way relay networks," *IEEE Communications Letters*, vol. 16, no. 7, pp. 1052–1055, 2012.

[30] V. Havary-Nassab, S. Shahbazpanahi, and A. Grami, "Optimal distributed beamforming for two-way relay networks," *IEEE Transactions on Signal Processing*, vol. 58, pp. 1238 – 1250, Mar. 2010.

[31] K. Lee, H. Sung, E. Park, and I. Lee, "Joint optimization for one and two-way MIMO AF multiple-relay systems," *IEEE Transactions Wireless Communications*, vol. 9, pp. 3671 – 3681, Dec. 2010.

[32] M. Zeng, R. Zhang, and S. Cui, "On design of distributed beamforming for two-way relay networks," *IEEE Transactions on Signal Processing*, vol. 59, pp. 2284–2295, May 2011.

[33] B. K. Chalise and L. Vandendorpe, "MIMO relay design for multipoint-to-multipoint communications with imperfect channel state information," *IEEE*

*Transactions on Signal Processing*, vol. 57, no. 7, pp. 2785–2796, 2009.

[34] P. Ubaidulla and A. Chockalingam, "MIMO relay precoder optimization in MIMO-relay networks with imperfect CSI," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5473–5484, 2011.

[35] D. P. Palomar and Y. C. Eldar, *Convex Optimization in Signal Processing and Communications*. Cambridge, UK ; New York: Cambridge University Press, 2010.

[36] A. Beck and Y. Eldar, "Strong duality in nonconvex quadratic optimization with two quadratic constraints," *SIAM J. Opt.*, vol. 17, no. 3, pp. 844–860, 2006.

[37] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[38] E. Gharavol, Y.-C. Liang, and K. Mouthaan, "Robust downlink beamforming in multiuser miso cognitive radio networks with imperfect channel-state information," *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 2852 –2860, July 2010.

[39] A. Aziz, M. Zeng, J. Zhou, C. Georghiades, and S. Cui, "Robust beamforming with channel uncertainty for two-way relay networks," in *IEEE International Conference on Communications (ICC)*, pp. 3632 –3636, June 2012.

[40] A. W. van der Vaart, *Asymptotic Statistics*. Cambridge ; New York: Cambridge University Press, 1998.

[41] J. Jerez, G. Constantinides, and E. Kerrigan, "Fpga implementation of an interior point solver for linear model predictive control," in *2010 International Conference on Field-Programmable Technology (FPT)*, pp. 316–319, 2010.

[42] B. Borchers and J. Young, "How far can we go with primal–dual interior point methods for sdp?," *Optimization online, http://www. optimization-online. org/DB FILE/2005/03/1082. pdf*, 2005.

[43] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control* (V. Blondel, S. Boyd, and H. Kimura, eds.), Lecture Notes in Control and Information Sciences, pp. 95–110, Springer-Verlag Limited, 2008.

[44] I. CVX Research, "CVX: Matlab software for disciplined convex programming, version 2.0 beta," Sept. 2012.

[45] D. Bertsimas and M. Sim *Tractable Approximations to Robust Conic Optimization Problems*, vol. 107(1), pp. 5–36, 2006.

# APPENDIX A

## GRADIENT FUNCTIONS

In this section we will present the gradient functions explicitly and also show how they are derived. Following is a list of all the gradient functions.

$$\begin{aligned}
\vec{\boldsymbol{g}}_{\boldsymbol{A}} &= \nabla_{\vec{\boldsymbol{A}}} G_p(\boldsymbol{A}, 0)\big|_{\boldsymbol{A}=\boldsymbol{A_0}}; \\
\vec{\boldsymbol{g}}_{\boldsymbol{h}} &= \nabla_{\vec{\boldsymbol{h'}}} G_p(\boldsymbol{A_0}, \boldsymbol{h'})\big|_{\boldsymbol{h'}=0}; \\
\vec{\boldsymbol{f}}_{\boldsymbol{A},i} &= \nabla_{\vec{\boldsymbol{A}}} f_i(\boldsymbol{A}, 0)\big|_{\boldsymbol{A}=\boldsymbol{A_0}}; \\
\vec{\boldsymbol{f}}_{\boldsymbol{h},i} &= \nabla_{\vec{\boldsymbol{h'}}} f_i(\boldsymbol{A_0}, \boldsymbol{h'})\big|_{\boldsymbol{h'}=0}; \\
\vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}} &= \nabla_{\vec{\Delta \boldsymbol{h}}} G_p(\bar{\boldsymbol{A}}, \Delta \boldsymbol{h}); \\
\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h}(k),i} &= \nabla_{\Delta \boldsymbol{h}} f_i(\boldsymbol{A}, \Delta \boldsymbol{h}(k)), \quad i = 1, 2; \\
\vec{\boldsymbol{g}}_m &= \nabla_{\vec{\boldsymbol{A}}} G_p(\boldsymbol{A}, \Delta \hat{\boldsymbol{h}}_{G_p})\Big|_{\boldsymbol{A}=\boldsymbol{A}(m)}; \\
\vec{\boldsymbol{f}}_{i,A(m)} &= \nabla_{\vec{\boldsymbol{A}}} f_i(\boldsymbol{A}, \Delta \hat{\boldsymbol{h}}_{f_i})\Big|_{\boldsymbol{A}=\boldsymbol{A}(m)}, \ i = 1, 2.
\end{aligned} \tag{A.1}$$

We use the following matrix identities extensively in our gradient computations

$$\begin{aligned}
\vec{\boldsymbol{A}} \cdot \vec{\boldsymbol{B}} &\equiv \Re\mathrm{Tr}[\boldsymbol{A}^H \boldsymbol{B}] = \Re\mathrm{Tr}[\boldsymbol{B}^H \boldsymbol{A}] \\
\mathrm{Tr}[\boldsymbol{ABC}] &\equiv \mathrm{Tr}[\boldsymbol{BCA}]\mathrm{Tr}[\boldsymbol{A}^T] \equiv \mathrm{Tr}[\boldsymbol{A}]
\end{aligned} \tag{A.2}$$

The gradients are computed by perturbing the variable of interest by a perturbed value, collecting the first order terms in the perturbation, and representing first order terms in vector form $\vec{\boldsymbol{B}} \cdot \vec{\boldsymbol{\delta}}$, where $\boldsymbol{\delta}$ is the perturbation. In this case $\vec{\boldsymbol{B}}$ will represent the gradient vector.

## A.1 Computation of $\vec{\boldsymbol{g}}_{\boldsymbol{A}}$

From the definition of $G_p(\boldsymbol{A}, 0)$ we have

$$G_p(\boldsymbol{A}, 0) \;=\; \|\boldsymbol{A}\boldsymbol{h}_1\|^2 p_1 + \|\boldsymbol{A}\boldsymbol{h}_2\|^2 p_2 + \mathrm{Tr}(\boldsymbol{A}^H\boldsymbol{A})\sigma_R^2 \tag{A.3}$$

In order to compute $\vec{\boldsymbol{g}}_{\boldsymbol{A}}$, we replace $\boldsymbol{A}$ by $\boldsymbol{A}+\boldsymbol{\delta}$ and take only the first order terms in $\boldsymbol{\delta}$. We will do this term-by-term for the three terms on the right-hand side of (A.3).

The first term on the right hand side of (A.3) under this replacement can be written as

$$\|(\boldsymbol{A}+\boldsymbol{\delta})\boldsymbol{h}_1\|^2 p_1 = \mathrm{Tr}\left[(\boldsymbol{h}_1)^H(\boldsymbol{A}+\boldsymbol{\delta})^H(\boldsymbol{A}+\boldsymbol{\delta})\boldsymbol{h}_1\right] p_1, \tag{A.4}$$

which when expanded gives first-order terms

$$\begin{aligned}\mathrm{Tr}\left[\boldsymbol{h}_1^H\boldsymbol{\delta}^H\boldsymbol{A}\boldsymbol{h}_1 + \boldsymbol{h}_1^H\boldsymbol{A}^H\boldsymbol{\delta}\boldsymbol{h}_1\right] p_1 &= 2\Re\mathrm{Tr}\left[\boldsymbol{\delta}^H\boldsymbol{A}\boldsymbol{h}_1\boldsymbol{h}_1^H\right] p_1 \\ &= 2p_1\mathrm{vec}\left(\boldsymbol{A}\boldsymbol{h}_1\boldsymbol{h}_1^H\right)\cdot\vec{\boldsymbol{\delta}}.\end{aligned} \tag{A.5}$$

Similarly, the second term in the right hand side of (A.3) when expanded yields first-order terms

$$2p_2\mathrm{vec}\left(\boldsymbol{A}\boldsymbol{h}_2\boldsymbol{h}_2^H\right)\cdot\vec{\boldsymbol{\delta}}.$$

The last term in (A.3) similarly yields first-order terms.

$$2\sigma_R^2\mathrm{vec}\left(\boldsymbol{A}\right)\cdot\vec{\boldsymbol{\delta}}.$$

In summary, we obtain

$$\vec{\boldsymbol{g}}_{\boldsymbol{A}} = \mathrm{vec}\left(2p_1\boldsymbol{A}\boldsymbol{h}_1\boldsymbol{h}_1^H + 2p_2\boldsymbol{A}\boldsymbol{h}_2\boldsymbol{h}_2^H + 2\sigma_R^2\boldsymbol{A}\right).$$

## A.2 Computation of $\vec{g}_{\boldsymbol{h}}$

$$G_p(\boldsymbol{A}, \boldsymbol{h}') \;=\; \|\boldsymbol{A}(\boldsymbol{h}_1 + \boldsymbol{h}'_1)\|^2 p_1 + \|\boldsymbol{A}(\boldsymbol{h}_2 + \boldsymbol{h}'_2)\|^2 p_2 + \mathrm{tr}(\boldsymbol{A}^H \boldsymbol{A})\sigma_R^2 \quad \text{(A.6)}$$

In order to compute $\vec{\boldsymbol{g}}_{\boldsymbol{h}}$, we replace $\boldsymbol{h}'$ by $\boldsymbol{\delta} \equiv [\boldsymbol{\delta}_1; \boldsymbol{\delta}_2]$ and take only the first order terms in $\boldsymbol{\delta}$. We will do this term-by-term for the three terms on the right-hand side of (A.6).

The first term on the right hand side of (A.6) under this replacement can be written as

$$\|\boldsymbol{A}(\boldsymbol{h}_1 + \boldsymbol{\delta}_1)\|^2 p_1 = \mathrm{Tr}\left[(\boldsymbol{A}(\boldsymbol{h}_1 + \boldsymbol{\delta}_1))^H \boldsymbol{A}(\boldsymbol{h}_1 + \boldsymbol{\delta}_1)\right] p_1, \quad \text{(A.7)}$$

which when expanded gives first-order terms

$$\begin{aligned}\left[\mathrm{Tr}[\boldsymbol{\delta}_1^H \boldsymbol{A}^H \boldsymbol{A}\boldsymbol{h}_1] + \mathrm{Tr}[(\boldsymbol{h}_1^H \boldsymbol{A}^H \boldsymbol{A}\boldsymbol{\delta}_1]\right] p_1 &= 2\Re\mathrm{Tr}[\boldsymbol{\delta}_1^H \boldsymbol{A}^H \boldsymbol{A}\boldsymbol{h}_1] p_1 \\ &= 2p_1 \mathrm{vec}\left(\boldsymbol{A}^H \boldsymbol{A}\boldsymbol{h}_1\right) \cdot \vec{\boldsymbol{\delta}}_1.\end{aligned} \quad \text{(A.8)}$$

Similarly, the second term in the right hand side of (A.6) when expanded yields first-order terms

$$2p_2 \mathrm{vec}\left(\boldsymbol{A}^H \boldsymbol{A}\boldsymbol{h}_2\right) \cdot \vec{\boldsymbol{\delta}}_2.$$

The last term in (A.6) is independent of the variable of interest ($\Delta\boldsymbol{h}$), hence its gradient with respect to $\Delta\boldsymbol{h}$ is zero.

In summary, by replacing $\boldsymbol{A}$ with $\boldsymbol{A}_0$ and combining vectors, we obtain

$$\vec{\boldsymbol{g}}_{\boldsymbol{h}} = 2\mathrm{vec}\left[p_1(\boldsymbol{A}_0^H \boldsymbol{A}_0 \boldsymbol{h}_1) \;;\; p_2(\boldsymbol{A}_0^H \boldsymbol{A}_0 \boldsymbol{h}_2)\right].$$

## A.3 Computation of $\vec{\boldsymbol{f}}_{\boldsymbol{A},i}$

From the definition of $f_{SINR_i}(\boldsymbol{A}, 0)$ we have

$$f_{SINR_i}(\boldsymbol{A},0) = \frac{|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\hat{\boldsymbol{h}}_j|^2 p_j}{\|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\|^2\sigma_R^2 + \sigma_i^2}, \quad i = 1,2, \quad j = 3-i. \tag{A.9}$$

$$\tag{A.10}$$

In order to compute $\vec{\boldsymbol{f}}_{\boldsymbol{A},i}$, we first replace $\boldsymbol{A}$ by $\boldsymbol{A}+\boldsymbol{\delta}$ in (A.9) to obtain

$$f_{SINR_i}(\boldsymbol{A}+\boldsymbol{\delta},0) = \frac{\hat{\boldsymbol{h}}_j^H(\boldsymbol{A}+\boldsymbol{\delta})^H\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T(\boldsymbol{A}+\boldsymbol{\delta})\hat{\boldsymbol{h}}_j p_j}{\|\hat{\boldsymbol{h}}_i^T(\boldsymbol{A}+\boldsymbol{\delta})\|^2\sigma_R^2 + \sigma_i^2}, \quad i = 1,2, \quad j = 3-i. \tag{A.11}$$

which when expanded gives first-order terms

$$\begin{aligned} &\frac{2p_j}{\|\hat{\boldsymbol{h}}_i^T \boldsymbol{A}\|^2\sigma_R^2 + \sigma_i^2}\Re\mathrm{Tr}\left[\boldsymbol{\delta}^H\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j\hat{\boldsymbol{h}}_j^H\right] \\ -&\frac{2p_j\sigma_R^2|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j|^2}{\left(\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2\right)^2}\Re\mathrm{Tr}\left[\boldsymbol{\delta}^H\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\right], \quad i=1,2, \quad j=3-i. \end{aligned} \tag{A.12}$$

This equation can be written in vector form as $\vec{\boldsymbol{f}}_{\boldsymbol{A},i}\cdot\boldsymbol{\delta}$, where

$$\vec{\boldsymbol{f}}_{\boldsymbol{A},i} = \mathrm{vec}\left(\frac{2p_j}{\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2}\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j\hat{\boldsymbol{h}}_j^H - \frac{2p_j\sigma_R^2|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j|^2}{\left(\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2\right)^2}\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\right), \ i=1,2, \ j=3-i.$$

## A.4 Computation of $\vec{\boldsymbol{f}}_{\boldsymbol{h},i}$

From the definition of $f_{SINR_i}(\boldsymbol{A}, \boldsymbol{h}')$ we have

$$f_{SINR_i}(\boldsymbol{A},\boldsymbol{h}') = \frac{|(\hat{\boldsymbol{h}}_i+\boldsymbol{h}_i')^T\boldsymbol{A}(\hat{\boldsymbol{h}}_j+\boldsymbol{h}_j')|^2 p_j}{\|(\hat{\boldsymbol{h}}_i+\boldsymbol{h}_i')^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2}, \quad i=1,2, \quad j=3-i. \tag{A.13}$$

In order to compute $\vec{\boldsymbol{f}}_{\boldsymbol{h},i}$, we first replace $\boldsymbol{h}'$ by $\boldsymbol{\delta}$ in (A.13) to obtain

$$f_i(\boldsymbol{A},\boldsymbol{\delta}) \;=\; \frac{|(\hat{\boldsymbol{h}}_i+\boldsymbol{\delta}_i)^T\boldsymbol{A}(\hat{\boldsymbol{h}}_j+\boldsymbol{\delta}_j)|^2 p_j}{\|(\hat{\boldsymbol{h}}_i+\boldsymbol{\delta}_i)^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2}, \quad i=1,2, \quad j=3-i. \tag{A.14}$$

which when expanded and rearranged gives first-order terms ($i=1,2,\ j=3-i$)

$$\begin{aligned}\frac{2p_j}{\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2}\Re\mathrm{Tr}\left[\boldsymbol{\delta}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j\hat{\boldsymbol{h}}_j^H\boldsymbol{A}^H\hat{\boldsymbol{h}}_i^* + \hat{\boldsymbol{\delta}}_j^H\boldsymbol{A}^H\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j\right]\\ -\frac{2p_j\sigma_R^2|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j|^2}{\left(\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2\right)^2}\Re\mathrm{Tr}\left[\boldsymbol{\delta}_i^T\boldsymbol{A}\boldsymbol{A}^H\hat{\boldsymbol{h}}_i^*\right].\end{aligned} \tag{A.15}$$

This equation can be written in vector form as $\vec{\boldsymbol{f}}_{\boldsymbol{h},i}\cdot\vec{\boldsymbol{\delta}}$, where ($i=1,2,\ j=3-i$)

$$\begin{aligned}\vec{\boldsymbol{f}}_{\boldsymbol{h},i} = \mathrm{vec}[\frac{2p_j}{\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2}\boldsymbol{A}^*\hat{\boldsymbol{h}}_j^*\hat{\boldsymbol{h}}_j^T\boldsymbol{A}^T\hat{\boldsymbol{h}}_i - \frac{2p_j\sigma_R^2|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j|^2}{\left(\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2\right)^2}\boldsymbol{A}^*\boldsymbol{A}^T\hat{\boldsymbol{h}}_i\ ;\\ \frac{2p_j}{\|\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\|^2\sigma_R^2+\sigma_i^2}\boldsymbol{A}^H\hat{\boldsymbol{h}}_i^*\hat{\boldsymbol{h}}_i^T\boldsymbol{A}\hat{\boldsymbol{h}}_j].\end{aligned} \tag{A.16}$$

## A.5 Expression for $\vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}}$

The computation of $\vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}}$ follows a similar procedure to the computations of $\vec{\boldsymbol{g}}_{\boldsymbol{h}}$ above. The result is

$$\vec{\boldsymbol{g}}_{\boldsymbol{\Delta h}} = \left[2p_1\mathrm{vec}(\boldsymbol{A}^H\boldsymbol{A}(\boldsymbol{h}_1+\Delta\boldsymbol{h}_1))\ ;\ 2p_2\mathrm{vec}(\boldsymbol{A}^H\boldsymbol{A}(\boldsymbol{h}_2+\Delta\boldsymbol{h}_2))\right].$$

## A.6 Expression for $\vec{\boldsymbol{f}}_{m,i}$

The results for $\vec{\boldsymbol{f}}_{m,i}$ is given by

$$\begin{aligned}
\vec{\boldsymbol{f}}_{i,A(m)} = & \quad [\boldsymbol{w}_i; \boldsymbol{w}_j] \quad i = 1, 2;\ j = 3 - i; \\
\boldsymbol{w}_i = & \quad 2p_j(\boldsymbol{h}_i^T \boldsymbol{A}\boldsymbol{h}_j + \boldsymbol{h}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_j + \boldsymbol{h}_j^T \boldsymbol{A}^T \Delta\boldsymbol{h}_i)\boldsymbol{A}^* {\boldsymbol{h}_j}^* \\
& -2p_i\gamma_i(\boldsymbol{h}_i^T(\boldsymbol{A} + \boldsymbol{A}^T)\Delta\boldsymbol{h}_i)(\boldsymbol{A}^H + \boldsymbol{A})^* {\boldsymbol{h}_i}^* \\
& -2\gamma_i\sigma_R^2 A^* A^T(\boldsymbol{h}_i + \Delta\boldsymbol{h}_i). \\
\boldsymbol{w}_j = & \quad 2p_j(\boldsymbol{h}_i^T \boldsymbol{A}\boldsymbol{h}_j + \boldsymbol{h}_i^T \boldsymbol{A}\Delta\boldsymbol{h}_j + \boldsymbol{h}_j^T \boldsymbol{A}^T \Delta\boldsymbol{h}_i)\boldsymbol{A}^H {\boldsymbol{h}_i}^*.
\end{aligned} \tag{A.17}$$

## A.7 Expression for $\vec{\boldsymbol{g}}_{\boldsymbol{A}(m)}$

$$\vec{\boldsymbol{g}}_{\boldsymbol{A}(m)} = vec\left[\boldsymbol{A}[(p_i/\sigma_R^2)\|(\boldsymbol{h}_i + \Delta\boldsymbol{h}_i)\| + (p_k/\sigma_R^2)\|(\boldsymbol{h}_k + \Delta\boldsymbol{h}_k)\| + \boldsymbol{I}_{M\times M}]\right]. \tag{A.18}$$

## A.8 Expression for $\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h}(k),i}$

$$\begin{aligned}
\vec{\hat{\boldsymbol{f}}}_{\boldsymbol{\Delta h}(k),i} = & [\boldsymbol{w}_i;\ \boldsymbol{w}_j] \quad i = 1, 2;\ j = 3 - i; \\
\boldsymbol{w}_i = & p_j[{\boldsymbol{h}_i}^T \boldsymbol{A}\boldsymbol{h}_j + \epsilon(\boldsymbol{h}_i^T \boldsymbol{A})\Delta\boldsymbol{h}_j + \epsilon(\boldsymbol{h}_j^T \boldsymbol{A}^T)\Delta\boldsymbol{h}_i]{(\boldsymbol{h}_j^T \boldsymbol{A}^T)}^H \\
& -(p_i\gamma_i\epsilon_i^2)[(\boldsymbol{h}_j^T \boldsymbol{A}^T)\Delta\boldsymbol{h}_i]{(\boldsymbol{h}_i^T(\boldsymbol{A} + \boldsymbol{A}^T))}^H \\
& -(\sigma_R^2\gamma_i)(\boldsymbol{A}^*)(\boldsymbol{A}^T\boldsymbol{h}_i + \epsilon\boldsymbol{A}^T\Delta\boldsymbol{h}_i) \\
\boldsymbol{w}_j = & p_j[{\boldsymbol{h}_i}^T \boldsymbol{A}\boldsymbol{h}_j + \epsilon(\boldsymbol{h}_i^T \boldsymbol{A})\Delta\boldsymbol{h}_j + \epsilon(\boldsymbol{h}_j^T \boldsymbol{A}^T)\Delta\boldsymbol{h}_i]{(\boldsymbol{h}_i^T \boldsymbol{A})}^H
\end{aligned}$$