# Optimized Training Signal Design

**A Master's Thesis**

**Submitted to the Faculty of**

**Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona**

**Universitat Politècnica de Catalunya**

**by**

**Marie Maros**

**In partial fulfilment**

**of the requirements for the degree of**

**TELECOMMUNICATIONS ENGINEER**

**Advisor: Mats Bengtsson**

**Stockholm, February 2014**

# Optimized Training Signal Design

Marie Maros

February 28, 2014

## ABSTRACT

In this thesis, the problem of finding an optimal training sequence for estimating a MIMO flat fading channel with spatially and temporally correlated Gaussian noise is considered.

The methods analyzed tailor the training sequence not only according to the known statistical CSI but also to the specific purpose the channel estimate will fulfill. The task of obtaining the optimal training sequence is formulated in two different ways, either guaranteeing a specific performance or setting a maximum training power budget. Two different applications are considered, the ZF precoder and the MMSE equalizer. The performance of the training sequence obtained by minimizing a metric that is representative for this applications is compared to using the training sequence that minimizes the channel estimate mean square error. Additionally, since some approximations are required to solve the optimization problem when using the application-oriented metrics, the impact of these is analyzed.

Two different approximations that allow convexification and lead to SDP formulations are considered for each problem. The two approximations and problem formulations are analyzed in terms of performance, training power and outage probability. The SDP formulations are then compared to the solutions provided by builtin functions in MATLAB that converge to a minimum in order to obtain information about how far from optimal the solutions obtained from the SDP are.

# ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor, Mats Bengtsson at the Signal Processing department in KTH, for proposing this topic for my master thesis, for his guidance and for trusting my work methodology during the thesis. I would also like to thank my teacher, Josep Vidal at SPCOM in UPC, for providing me with a global view about the topics treated.

Thank you, Pol del Aguila Pla for your support and assistance. Finally, I would like to thank my mother and father for their unconditional support during my stay in Sweden and always.

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# NOTATION

**x** Lower case boldface is used to represent column vectors.

**X** Upper case boldface is used to represent matrices.

$\mathbf{X}^T$,$\mathbf{X}^*$, $\mathbf{X}^H$ denote transpose, conjugate and conjugate transpose respectively.

$\mathbf{X}^\dagger$ denotes **X**'s Moore-Penrose pseudoinverse.

$\mathrm{tr}(\mathbf{X})$ denotes the trace of **X**.

s.t. stands for subject to.

$\mathcal{CN}(\boldsymbol{\mu}, \mathbf{C})$ stands for circularly symmetric complex Gaussian random vectors, with mean $\boldsymbol{\mu}$ and covariance matrix **C**.

$\mathbf{Hess}(f)$ denotes the Hessian matrix of a function $f$

# ACRONYMS

**CDMA** Code Division Multiple Access

**CSI** Channel State Information

**FDM** Frequency Division Multiplexing

**FDMA** Frequency Division Multiple Access

**GPP** Guaranteed Performance Problem

**ISI** Inter Symbol Inteference

**KKT** Karush Kuhn Tucker. Referring to the KKT conditions of optimality described in chapter 3

**LMI** Linear Matrix Inequality

**MIMO** Multiple Input Multiple Output

**MISO** Multiple Input Single Output

**MMSE** Minimum Mean Square Error

**MPP** Maximized Performance Problem

**OFDM** Orthogonal Frequency-Division Multiplexing

**OFDMA** Orthogonal Frequency Division Multiple Access

**PAR** Peak to Average Ratio

**PDF** Probability Density Function

**SDMA** Space Division Multiple Access

**SDP** Semi Definite Program

**SIMO** Single Input Multiple Output

**SISO** Single Input Single Output

**SNR** Signal to Noise Ratio

**TDMA** Time Division Multiple Access

**ZF** Zero Forcing

# Part I

# INTRODUCTION

# 1

# INTRODUCTION

As technology continues to improve, mobile devices that require connectivity to the network increase their presence in our daily lives. Specifically, wireless access to the Internet and multimedia applications require a much higher volume of data traffic than speech communication for mobile phones. Additionally, the number of users has increased exponentially during the past years, entailing that not only the amount of data has increased, but also the amount of users that require data simultaneously.

Therefore, different strategies to separate the signals that are sent to different users are used. The most intuitive technique for this purpose is TDMA (Time Division Multiple Access). In a TDMA system all users are using the same set of frequencies to communicate but in different time moments. Alternatively, it is possible to have all users accessing the system at the same time but using different frequencies. This technique is called FDMA (Frequency Division Multiple Access).

Considering the demand on throughput and the scarcity of electromagnetic spectrum, other techniques are required to optimize as much as possible the capabilities of the wireless channel that is available. The technique used in contemporary systems (3G) is CDMA (Code Division Multiple Access), in which all users transmit at the same time using the same frequencies but are separated by user specific codes that allow to reduce the impact of user interference. The main inconvenient of this last technique is that it requires a strict control on the transmitting power used by each user and a very tight time synchronization.

MIMO (Multiple Input Multiple Output) systems, based on the use of several antennas at transmission and reception, have risen as a technology that allows to fulfill the demands, both in throughput and in simultaneous transmission, of modern communications. The use of several antennas can provide robustness or increased throughput when considering a single user communication, or in case we are considering a multiple user communication, an alternative of separating different users. This technique is referred to as SDMA (Space Division Multiple Access).

Spread spectrum techniques used in 3G are being abandoned in the current communication standard 4G, in which MIMO systems play a relevant role. In the specific case of 4G the different antennas of a

MIMO system allow to easily transmit different orthogonal subcarriers, which are reffered to as OFDM or OFDMA, when used to transmit information from different users in different subcarriers. This may seem identical to the approach used in FDM, but in this case the subcarriers are selected to be orthogonal to each other, which erases the necessity of inter carrier guards or the use of frequency selecting filters in reception. The subcarrier frequencies are selected to be orthogonal to each other, but also to be as close in frequency as possible so that the spectral efficiency is increased. This implies that each of the subcarriers uses a very narrow bandwidth to transmit which leads to a large symbol time. This may seem as an inconvenient since a large symbol time might imply that the communications is slow, but this fact is compensated by the use of several subcarriers. Additionally, it provides a very significant advantage since the effects of the radio channel that impact the communication due to short symbol times are not as critical.

## 1.1 SISO AND SIMO COMMUNICATIONS

SISO (Single Input Single Output) systems have been the conventional choice in communications. The main inconvenience this systems have is that they are vulnerable to interferences present in the radio environment. This means that in some occasions, it is possible that the link actually offers a lower performance than expected or even breaks down. To reduce the likelihood of these events the so-called *diversity* techniques have been developed. These techniques rely on sending or receiving more than one copy of the signal of interest providing additional robustness to the system.

SIMO (Single Input Multiple Output) systems use one transmitting antenna while using several receiving antennas. Due to the different paths a signal may follow when being transmitted through the electromagnetic channel it is possible that the copies received in the different antennas are not identical, meaning that it is actually possible to either select or combine them so that the overall received signal quality is improved. The improvement on performance resulting from of an intelligent combination of this different received signals is referred to as *beamforming gain*.

It is also possible to obtain a beamforming gain when using a MISO (Multiple Input Single Output) system. In this case the different transmitting antennas are assigned different weights so that the energy can be sent to specific directions, and therefore an enhancement effect is perceived in reception. The use of this kind of systems strongly motivates the need for channel characterization and estimation since intelligent combination techniques require knowledge of the channel to achieve their full potential.

## 1.2 MIMO COMMUNICATIONS

MISO and SIMO systems allow to improve the quality of the communication link by allowing the use of diversity techniques, but they still would require techniques such as TDMA, FDMA or CDMA to allow the transmission of several data streams simultaneously. On the other hand, MIMO (Multiple Input Multiple Output) systems allows to perform *beamforming* on each of the antenna links obtaining a separate data stream for each. As mentioned previously, this can be used either to multiplex a multiuser communication or to provide diversity for a single user transmission. Additionally, as in the previous cases, for this to work it is necessary to obtain knowledge of the channel.

## 1.3 THESIS MOTIVATION

Channel estimation is required in order to exploit the capabilities of multi antenna environments. In order to be able to estimate the channel's response, pilot sequences that both the transmitter and receiver known are used. Current techniques rely on the use of fixed statistically white sequences to provide a channel estimate. This technique is optimal in both terms of entropy and resulting variance when using the Least Square estimator. This techniques do not assume a statistical knowledge on the channel, but only on the noise's distribution.

Most research in optimizing the training sequence relies on obtaining statistical information of the channel, which is later on used to tailor a training sequence that exploits it and allows to estimate the channel faster and with higher accuracy. These techniques do not consider the specific application that the channel estimate will have, since different diversity techniques will use the estimation differently. In this thesis the approach proposed in [10] and [11],that considers the specific application to which the channel estimate will be applied, is analyzed. Specifically, the problem of finding the training sequence can be expressed as an optimization problem as in (1).

$$\begin{aligned} &\underset{\mathbf{P}}{\text{maximize}} && \text{Application oriented performance} \\ &\text{subject to} && \text{Training power restriction} \end{aligned} \tag{1}$$

Where **P** represents the training sequence. Alternatively, the problem can be also formulated as in (2).

$$\begin{aligned} &\underset{\mathbf{P}}{\text{minimize}} && \text{Training power} \\ &\text{subject to} && \text{Application oriented performance restriction} \end{aligned} \tag{2}$$

Both problem formulations and the corresponding developments, following the steps in [11] and [10], for them to become efficiently solvable, are considered and analyzed in this thesis. Additionally, some application oriented performance functions, proposed in [10],

are analyzed and evaluated. Later on, keeping in mind that some approximations have been done to solve the original problems efficiently, using the techniques proposed in [10] and [11], a comparison to the solution obtained when using alternative and more resource consuming methods is done. The contributions of this thesis can be found in chapter 13.

## 1.4 THESIS GOALS

Both [10] and [11] propose approximations that are required in order to convexify the problem formulations that are derived from equations (1) and (2). An analysis of the concepts behind them and a performance comparison is provided through experimental results. Additionally, the application oriented performance functions mentioned in [10] and [11] require some approximations for their use as well. An analysis of the impact on the perceived performance is provided through numerical simulation. Finally, as proposed in [11], in order to be able to write the problem in the final form a relaxation is required. The impact of this is evaluated through the minimization using local minimum finding algorithms of the non-convex formulation of the problems.

The goals of this thesis are, therefore:

- Evaluate the resulting performance when applying the methodology proposed on [11] in order to solve the problem formulations proposed in [10].
- Compare the performance to the cases in which an analytical solution can be found in order to evaluate the ideas proposed in [11].
- Evaluate the approximations required for simplicity and causality of the application oriented performance functions comparing them to the performance obtained by using the non approximated functions after obtaining the training sequence.
- Evaluate which of the two different approximations required for convexification provides better results in terms of performance and training power.
- Evaluate how far from optimal the solution of the convexified problem using the methodology provided in [11] is when compared to minimizing the non-convex problem.

In order to carry out the mentioned goals, a theoretical analysis of the different cases, refer to chapters 7 and 9, is done in order to discard or justify the necessity of the experiments later described in chapter 10. In chapter 10 the results obtained are contrasted with the previous theoretical analysis and further conclusions are extracted in

the cases in which additional information from the experiments can be obtained.

# 2

## THESIS OUTLINE

This thesis is organized in three different parts. In the first part some preliminary knowledge that allows deeper understanding of the thesis and that allows to reader to comprehend the motivation behind some of the derivations is provided. Some convex optimization concepts, required for the understanding of interior point methods to solve SDPs, and channel modeling basics, together with the channel model used during the entire thesis, are provided.

In the second part the theoretical derivations that allow finding the training sequences, as well as the experiments carried out to evaluate the solutions and the conclusions extracted from each of them are explained.

In the third part, a global summary of the thesis, the main conclusions and possible further work is included.

# Part II

# PRELIMINARIES

# 3

# SDP AND INTERIOR POINT METHODS

In this chapter some preliminary knowledge is provided to understand the motivation behind using relaxations to obtain convex optimization problems. Additionally, a general formulation of the SDP is included so that the reader can verify that the problems in chapter 7 fulfill the conditions. Finally, a description of some algorithms used to solve this kind of problems is provided. The concepts required to understand the formulations and algorithms are developed, but not in depth, in this section. For a deeper insight on the topic refer to [5].

## 3.1 OPTIMIZATION

Constrained optimization problems are of interest in any field of engineering, since they allow to achieve an optimal solution provided some constraints or bounds. A general optimization problem can be expressed as in equation (3).

$$\begin{aligned} &\text{minimize} && f_0(\mathbf{x}) \\ &\text{subject to} && f_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m. \\ & && h_i(\mathbf{x}) = 0\ i = 1, \ldots, p, \end{aligned} \tag{3}$$

where $f_0(\mathbf{x})$ is the objective function, $\mathbf{x}$ is the optimization variable, $f_i(\mathbf{x})$ are the inequality constraint functions and $h_i(\mathbf{x})$ are the equality constraint functions.

Generally, the objective function $f_0(\mathbf{x})$ represents the cost of choosing a specific value of $\mathbf{x}$, and the constraints represent some boundaries this value must respect. Therefore, the aim is to make the best possible choice of $\mathbf{x}$.

A solution for a general optimization problem can be very hard to find, requiring very large amounts of memory and long computational times. Fortunately, some classes of optimization problems have efficient algorithms that reliably solve them.

## 3.2 CONVEX OPTIMIZATION

Convex optimization problems do not generally have an analytical formula to solve them but have very effective algorithms that allow

solving them. Interior-point methods can in most cases solve these problems up to a specified accuracy and with relatively low complexity. Therefore, being able to formulate a problem as a convex optimization problem provides a strategy to obtain a relatively fast and accurate solution.

Some basic definitions and concepts are required to understand the restrictions over a convex optimization problem. These will be provided in the following subsections.

### 3.2.1 *Convex sets: basic definitions*

*Affine sets*: A set $\mathcal{C} \subseteq \mathbb{R}^n$ is affine if the line through any two distinct points in $\mathcal{C}$ lies in $\mathcal{C}$. Mathematically this can be expressed as in equation (4).

$$\forall\, \mathbf{x_1}, \mathbf{x_2} \in \mathcal{C}, \forall \theta \in \mathbb{R} \quad \theta(\mathbf{x_1}) + (1-\theta)\mathbf{x_2} \in \mathcal{C} \tag{4}$$

*Convex sets*: A set $\mathcal{C} \subseteq \mathbb{R}^n$ is convex if the line segment between two distinct points in $\mathcal{C}$ lies in $\mathcal{C}$, which can mathematically be expressed as in equation (5).

$$\forall\, \mathbf{x_1}, \mathbf{x_2} \in \mathcal{C}, \forall \theta \in [0,1] \quad \theta(\mathbf{x_1}) + (1-\theta)\mathbf{x_2} \in \mathcal{C} \tag{5}$$

*Cones*: A set $\mathcal{C} \subseteq \mathbb{R}^n$ is a cone or nonnegative homogeneous if $\forall \mathbf{x} \in \mathcal{C}$ and $\forall \theta \geq 0$, $\theta \mathbf{x} \in \mathcal{C}$.

*Convex cone*: A set is a convex cone if it is both a cone and convex.

*Proper cone*: A cone $\mathcal{K} \in \mathbb{R}^n$ is a proper cone if it is convex, closed, has a non empty interior, and is pointed[1]

A proper cone can be used to define a generalized inequality which allows to define an ordering in $\mathbb{R}^n$ that will conserve some properties of the ordering in $\mathbb{R}$. The ordering, and inequality, can be defined as in equation (6).

$$\mathbf{x} \preceq_K \mathbf{y} \iff \mathbf{y} - \mathbf{x} \in \mathcal{K} \tag{6}$$

All derivations in this section are easily extended to generalized inequalities. Refer to [5] for details.

### 3.2.2 *Convex functions*

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $\mathrm{dom}(f)$ is a convex set and $\forall \mathbf{x}$, $\mathbf{y} \in \mathrm{dom}(f)$, $\forall \theta \in [0,1]$ equation (7) is fulfilled.

$$f(\theta \mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}) \tag{7}$$

This means that the cord from point $(\mathbf{x}, f(\mathbf{x}))$ to $(\mathbf{y}, f(\mathbf{y}))$ is above the graphical representation of $f$. From this definition, it can also

1 Pointed: does not contain any lines.

be extracted that if a function is convex, a local minimum is also the global minimum.

A function $f$ is concave if the function $-f$ is convex. Additionally when the inequality holds with equality, the function is affine. Affine functions are both convex and concave.

#### 3.2.2.1 *Some operations that preserve convexity*

The operations that preserve convexity and are relevant for the understanding of this chapter are described below.

*Intersection*: the intersection of convex sets is a convex set.

*Affine functions*:

1. If $\mathcal{S} \subseteq \mathbb{R}^n$ is convex and $f : \mathbb{R}^n \to \mathbb{R}^m$ is an affine function the image of $\mathcal{S}$ under $f$, $f(\mathcal{S}) = \{f(\mathbf{x}) | \mathbf{x} \in \mathcal{S}\}$ is convex.
2. If $\mathcal{S} \subseteq \mathbb{R}^m$ and $f : \mathbb{R}^n \to \mathbb{R}^m$ is an affine function, the inverse image of $\mathcal{S}$ under $f$, $f^{-1}(\mathcal{S}) = \{\mathbf{x} | f(\mathbf{x}) \in \mathcal{S}\}$ is convex.

*Pointwise maximum*: If $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ are convex functions their pointwise maximum $f(\mathbf{x}) = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$ is also convex. In the same way, the pointwise infimum of a set of concave functions is concave.

#### 3.2.2.2 *Sublevel sets*

The $\alpha$-sublevel set of a function $f : \mathbb{R}^n \to \mathbb{R}$ is defined as $\mathcal{C}_\alpha = \{\mathbf{x} \in \text{dom}(f) | f(\mathbf{x}) \leq \alpha\}$. An important property of convex functions is that all sublevel sets of a convex function are convex.

### 3.2.3 *Problem formulation*

A convex optimization problem is one that can be written as:

$$\begin{aligned} &\text{minimize} && f_0(\mathbf{x}) \\ &\text{s.t.} && f_i(\mathbf{x}) \leq 0 \;\; i = 1, \ldots, m \\ &&& \mathbf{a_i}^T \mathbf{x} = b_i \;\; i = 1, \ldots, p \end{aligned} \tag{8}$$

where $f_i$, $i = 0, \ldots, m$ are convex functions.

To be able to continue, some new terminology is required:

*Feasible point*: All points that fulfill the constraints and at the same time belong to the domain of the objective function $f_0(\mathbf{x})$ are feasible points.

*Feasible set*: Set that contains all the feasible points.

*Feasible problem*: The problem (8) is feasible if there exists one or more feasible points.

*Infeasible problem*: The problem is infeasible if it is not feasible.

A convex problem adds the following requirements over a general optimization problem:

1. The objective function must be convex.
2. The inequality constraint functions are convex.
3. The equality constraint functions must be affine.

From this additional requirements it is possible to say that the feasible set of a convex optimization is a convex set, since:

1. the domain of a convex function is convex,
2. any $\alpha$-sublevel set of convex functions is a convex set,
3. affine functions preserve convexity
4. and the intersection of convex sets is a convex set.

A generalization of the standard convex optimization problem can be done using general inequalities if the inequality constraints are allowed to be vector or matrix valued.

$$\begin{aligned} &\text{minimize} && f_0(\mathbf{x}) \\ &\text{s.t.} && f_i(\mathbf{x}) \preceq_{K_i} \mathbf{0} \;\; i = 1, \ldots, m \\ & && \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{9}$$

Where $f_0 : \mathbb{R}^n \to \mathbb{R}$, $K_i \subseteq \mathbb{R}^{k_i}$ are proper cones and $f_i : \mathbb{R}^n \to \mathbb{R}^{k_i}$ are $K_i$-convex.

### 3.2.4 *Conic problems*

A conic problem is a generalized convex optimization problem that has a linear objective function and an inequality constraint function that is affine. Mathematically it can be expressed as in (10).

$$\begin{aligned} &\text{minimize} && \mathbf{c}^T\mathbf{x} \\ &\text{s.t.} && \mathbf{Fx} + \mathbf{g} \preceq_K \mathbf{0} \\ & && \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{10}$$

#### 3.2.4.1 *SDP*

A Semidefinite Program is a problem within the family of conic problems. SDPs have the additional property that the cone $K$ is the cone of positive semidefinite matrices. Mathematically this can be expressed as in (11).

$$\begin{aligned} &\text{minimize} && \mathbf{c}^T\mathbf{x} \\ &\text{s.t.} && x_1\mathbf{F_1} + \cdots + x_n\mathbf{F_n} + \mathbf{G} \preceq \mathbf{0} \\ & && \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{11}$$

Where $\mathbf{G}, \mathbf{F_1}, \ldots, \mathbf{F_n}$ are semidefinite positive matrices.

It is common to refer to problems that have a linear objective function, several LMIs, inequality constrains and equality constraints as

SDPs as well, since they can be easily transformed into an SDP by using the methodology described in [5, page 169].

SDPs are efficiently solved by using Interior point methods which rely on duality to achieve the solution. Duality will be dealt with in the next section.

### 3.2.5 *Duality*

Recall the general optimization problem formulation in (3). The Lagrangian function is defined by augmenting the objective function taking into account the constraints as in (12).

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^{p} \nu_i h_i(\mathbf{x}) \tag{12}$$

Then, the Lagrange dual function is defined as (13).

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathcal{D}} (f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i f_i(\mathbf{x}) + \sum_{i=1}^{p} \nu_i h_i(\mathbf{x})) \tag{13}$$

where $\mathcal{D}$ is the feasible set. Considering that all points that belong to the feasible set fulfill the inequality and equality constraints, $h_i(\mathbf{x})$ is 0, making the last term of the expression in (13) equal to 0, and $f_i(\mathbf{x})$ is non-positive $\forall \mathbf{x} \in \mathcal{D}$.

If $\boldsymbol{\lambda} \succeq \mathbf{0}$ is assumed, the second term in (13) is non-positive, meaning that the Lagrange dual function for any value of $\boldsymbol{\nu}$ and $\boldsymbol{\lambda} \succeq \mathbf{0}$ is a non-trivial lower bound for the optimal value, i.e., $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leq f_0(\mathbf{x}_{opt})$.

The Lagrange dual function can be interpreted as a softened version of the expression in (14) which is equivalent to the optimization problem at hand but it bunches together cost function and restrictions.

$$\underset{\mathbf{x}}{\text{minimize}} \, (f_0(\mathbf{x}) + \sum_{i=1}^{m} I_-(f_i(\mathbf{x})) + \sum_{i=1}^{p} I_0(h_i(\mathbf{x}))) \tag{14}$$

where $I_-(u)$ is $\infty$ for non-negative values of $u$ and 0 for the rest, and $I_0(u)$ is 0 when $u = 0$ and $\infty$ otherwise.

As mentioned previously, the Lagrange dual function provides a lower bound of $f_0(\mathbf{x}_{opt})$. The question that arises from this is if it is possible to find the best lower bound, or even better, a lower bound that is actually tight. This leads to the dual problem which is defined as:

$$\begin{aligned} &\underset{(\boldsymbol{\lambda}, \boldsymbol{\nu})}{\text{maximize}} && g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \\ &\text{s.t.} && \boldsymbol{\lambda} \succeq 0 \end{aligned} \tag{15}$$

the solution for this problem $(\boldsymbol{\lambda}_{opt}, \boldsymbol{\nu}_{opt})$ is referred to as dual optimal or optimal Lagrange multipliers. Note that the dual problem

is always convex even if the original problem is not, since it is the maximization of a concave function[2].

#### 3.2.5.1 *Weak duality and strong duality*

Considering the previous, $g(\boldsymbol{\lambda}_{opt}, \boldsymbol{\nu}_{opt})$ is by definition the best lower bound on $f_0(\mathbf{x}_{opt})$ that can be obtained of the Lagrange dual function. Therefore defining $g(\boldsymbol{\lambda}_{opt}, \boldsymbol{\nu}_{opt}) = d_{opt}$ and $f_0(\mathbf{x}_{opt}) = p_{opt}$, the inequality $d_{opt} \leq p_{opt}$ holds even if the original problem is not convex. When this inequality is not met with equality there is *weak duality* and the difference $p_{opt} - d_{opt}$ is the *duality gap*. On the other hand, if the inequality holds with equality there is *strong duality*. Unfortunately, that a problem is convex does not always imply strong duality, but under more restrictive conditions strong duality is guaranteed.

The pair $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is referred to as dual feasible points, and it provides information of how suboptimal a feasible point is without necessarily knowing the value of $p_{opt}$ as follows:

$$f_0(\mathbf{x}) - p_{opt} \leq f_0(\mathbf{x}) - g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \tag{16}$$

from which it can be said that $\mathbf{x}$ is $\epsilon$ suboptimal if $\epsilon = f_0(\mathbf{x}) - g(\boldsymbol{\lambda}, \boldsymbol{\nu})$. This will provide a good heuristic stopping criteria in the optimization algorithms.

#### 3.2.5.2 *Slater's condition*

If a problem is convex, Slater's condition states that if the non-affine inequalities, meaning that this condition is not necessary on affine inequalities, hold strictly, the duality gap is 0, i.e. there is strong duality. A more detailed explanation of this can be found in [5, page 226].

#### 3.2.5.3 *KKT Conditions*

Assume that the functions $f_0, f_1, ..., f_m$ are differentiable and have open domains. Assume also that strong duality holds. The primal and dual optimal points $\mathbf{x}_{opt}$ and $(\boldsymbol{\lambda}_{opt}, \boldsymbol{\nu}_{opt})$ by definition minimize the Lagrangian $L(\mathbf{x}_{opt}, \boldsymbol{\lambda}_{opt}, \boldsymbol{\nu}_{opt})$ (12), implying that the Lagrangian's gradient at $\mathbf{x}_{opt}$ is 0. This is mathematically expressed as in (17).

$$\nabla f_0(\mathbf{x}_{opt}) + \sum_{i=1}^{m} \lambda_{opt_i} \nabla f_i(\mathbf{x}_{opt}) + \sum_{i=1}^{p} \nu_{opt_i} \nabla h_i(\mathbf{x}_{opt}) = 0 \tag{17}$$

2 As mentioned previously, the pointwise infimum of a set of concave functions is concave

From which, considering that $\mathbf{x}_{opt}$ is necessarily feasible the KKT conditions are derived:

$$\begin{aligned}
&f_i(\mathbf{x}_{opt}) \leq 0 && i = 1, \ldots, m \\
&h_i(\mathbf{x}_{opt}) = 0 && i = 1, \ldots, p \\
&\lambda_{opt_i} \geq 0 && i = 1, \ldots, m \\
&\lambda_{opt_i} f_i(\mathbf{x}_{opt}) = 0 && i = 1, \ldots, m \\
&\nabla f_0(\mathbf{x}_{opt}) + \sum_{i=1}^{m} \lambda_{opt_i} \nabla f_i(\mathbf{x}_{opt}) + \sum_{i=1}^{p} \nu_{opt_i} \nabla h_i(\mathbf{x}_{opt}) = 0
\end{aligned} \tag{18}$$

Where the two first restrictions are the fulfillment of the restrictions, the third is the restriction on the dual problem and the fourth is required for strong duality.

Note that convexity or any other kind of restriction has not been forced upon the problem.

#### 3.2.5.4 *KKT conditions for convex problems*

If the problem is convex, any points $\hat{\mathbf{x}}$ and $(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\nu}})$ that fulfill the KKT conditions are the primal and dual optimal, and strong duality holds.

The first two conditions force $\hat{\mathbf{x}}$ to be primal feasible. Additionally, since the problem is convex the Lagrangian function will be as well. Therefore, a point $\mathbf{x}$ that fulfills the KKT conditions minimizes the Lagrangian. From this, $g(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\nu}}) = L(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\nu}}) = f_0(\hat{\mathbf{x}}) + \sum_{i=1}^{m} \hat{\lambda}_i f_i(\hat{\mathbf{x}}) + \sum_{i=1}^{p} \hat{\nu}_i h_i(\hat{\mathbf{x}}) = f_0(\hat{\mathbf{x}})$. Meaning that $\hat{\mathbf{x}}$ and $(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\nu}})$ are the primal and dual optimal respectively, and strong duality holds.

All facts considered, it is concluded that solving the KKT conditions is equivalent to solving the original problem if it is convex.

## 3.3 ALGORITHMS

The aim of this section is to provide a small insight of the Interior-point methods used later on in the master thesis. As a general idea, interior point methods solve the original problems by applying Newton's method to a sequence of modified KKT conditions, which are equality constrained optimization problems. Therefore, a brief explanation of Newton's algorithm when solving unconstrained problems will be provided and later extended to problems with equality constraints. Finally, Interior-point methods will be dealt with in the last section of this chapter.

### 3.3.1 *Unconstrained minimization*

Consider the following optimization problem:

$$\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) \tag{19}$$

assuming $\mathbf{x} \in \text{dom}(f)$ the *newton step* can be defined as in (20).

$$\boldsymbol{\Delta x_{nt}} = -\boldsymbol{Hess}(f(\mathbf{x}))^{-1}\boldsymbol{\nabla} f(\mathbf{x}) \tag{20}$$

The newton step defines a descent direction unless $\boldsymbol{\nabla} f(\mathbf{x}) = 0$, which guarantees convergence to a local or global minimum. The newton step can be interpreted as:

1. the minimizer of the second order approximation of $f$ at $\mathbf{x}$:

$$\hat{f}(\mathbf{x}+\mathbf{v}) = f(\mathbf{x}) + \boldsymbol{\nabla} f(\mathbf{x})^T\mathbf{v} + \frac{1}{2}\mathbf{v}^T\boldsymbol{Hess}(f(\mathbf{x}))\mathbf{v} \tag{21}$$

2. the steepest descent direction in the Hessian norm
3. a linear approximation of the optimality condition $\boldsymbol{\nabla} f_0(\mathbf{x}_{opt}) = 0$:

$$\boldsymbol{\nabla} f(\mathbf{x}+\mathbf{v}) = \boldsymbol{\nabla} f(\mathbf{x}) + \boldsymbol{\nabla}^2 f(\mathbf{x})\mathbf{v} = 0 \tag{22}$$

The *Newton decrement* is defined as (23).

$$\lambda(\mathbf{x}) = (\boldsymbol{\nabla} f(\mathbf{x})^T\boldsymbol{Hess}(f(\mathbf{x}))^{-1}\boldsymbol{\nabla} f(\mathbf{x}))^{1/2} \tag{23}$$

The Newton decrement can written as (24).

$$\frac{1}{2}\lambda(\mathbf{x})^2 = \hat{f}(\mathbf{x}) - \hat{f}(\mathbf{x} + \boldsymbol{\Delta x_{nt}}) \tag{24}$$

Meaning that the quantity $\lambda^2/2$ is an estimate of the suboptimality of $\mathbf{x}$ since it can be interpreted as an approximation of $f_0(\mathbf{x}) - p_{opt}$.

Newton's method follows the steps in (1).

**Algorithm 1** Newton's method for unconstrained problems.

**Require:** $\mathbf{x} \in \text{dom}(f)$, tolerance $\epsilon > 0$

**repeat**

1. Compute Newton step and decrement.
2. Choose step size $t$ using line search.
3. Update $\mathbf{x} = \mathbf{x} + t\boldsymbol{\Delta x_{nt}}$

**until** $\lambda^2/2 \leq \epsilon$

Where line search algorithms[3] are meant to select an appropriate step size $t$.

3 Refer to the reference for further information

### 3.3.2 *Problems with equality constraints*

Imagine now that the problem to solve is the one in (25).

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{s.t.} && \mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned} \tag{25}$$

For the purpose of solving this kind of problems the definition of the Newton step is modified to take into account the equality constraint. Considering the addition of the equality constraint, the initial points of each iteration need to be feasible, and the directions in which the Newton step moves need to be feasible as well.

In this case the newton step has its origins in the following minimization problem:

$$\begin{aligned} &\underset{\mathbf{v}}{\text{minimize}} && \hat{f}(\mathbf{x}+\mathbf{v}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T\mathbf{v} + \frac{1}{2}\mathbf{v}^T \boldsymbol{Hess}(f(\mathbf{x}))\mathbf{v} \\ &\text{s.t.} && \mathbf{A}(\mathbf{x}+\mathbf{v}) = \mathbf{b} \end{aligned} \tag{26}$$

which is a quadratic convex problem with one equality constraint that can be solved analytically. Considering the duality theory explained previously, solving this problem is equivalent to solving its KKT conditions (27).

$$\begin{aligned} \mathbf{A}\mathbf{v}_{opt} &= \mathbf{b} \\ \nabla^2 f(\mathbf{x})\mathbf{v}_{opt} + \nabla f(\mathbf{x}) + \mathbf{A}^T\boldsymbol{\nu}_{opt} &= 0 \end{aligned} \tag{27}$$

where $\mathbf{v}_{opt} = \boldsymbol{\Delta x_{nt}}$ and $\boldsymbol{\nu}_{opt}$ is the associated dual optimal.

Therefore, the newton decrement can be characterized by the KKT system:

$$\begin{bmatrix} \nabla^2 f(\mathbf{x}) & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta x_{nt}} \\ \boldsymbol{\nu} \end{bmatrix} = \begin{bmatrix} -\nabla f(\mathbf{x}) \\ 0 \end{bmatrix} \tag{28}$$

The Newton's step in this case is only defined where the KKT matrix is non-singular, forcing a feasible direction. Newton's method for problems with equality constraints follows the steps in algorithm (2).

**Algorithm 2** Newton's method for equality constrained problems.

**Require:** $\mathbf{x} \in \text{dom}(f)$ with $\mathbf{A}\mathbf{x} = \mathbf{b}$, tolerance $\epsilon > 0$

**repeat**

1. Compute Newton step and decrement.
2. Choose step size $t$ using line search.
3. Update $\mathbf{x} = \mathbf{x} + t\boldsymbol{\Delta x_{nt}}$

**until** $\lambda^2/2 \leq \epsilon$

### 3.3.3 *Interior-point methods*

As mentioned previously, interior-point methods successively apply Newton's method to solve an optimization problem with linear equality constraints. Therefore, the first step is to write a problem like (8) as an equality constrained problem to which Newton's method can be applied. This can be done as in 29.

$$\begin{aligned} &\text{minimize} \quad f_0(\mathbf{x}) + \sum_{i=1}^{m} I_-(f_i(\mathbf{x})) \\ &\text{s.t.} \qquad\qquad\qquad\quad \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{29}$$

The objective function of the optimization problem in (29) is generally not differentiable and therefore Newton's method can not be used. Hence, an approximation of the indicator function $I_-$ is introduced:

$$\hat{I}_-(u) = -\frac{1}{t}\log(-u) \quad \text{with } t > 0 \tag{30}$$

The function in 30 is convex and non-decreasing. Replacing the indicator in (29), the objective function obtained is now convex.

$$\begin{aligned} &\text{minimize} \quad f_0(\mathbf{x}) - \sum_{i=1}^{m} \frac{1}{t}\log(-f_i(\mathbf{x})) \\ &\text{s.t.} \qquad\qquad\qquad\quad \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{31}$$

Where $\Phi(\mathbf{x}) = -\sum_{i=1}^{m}\log(-f_i(\mathbf{x}))$ is the logarithmic barrier.

Multiplying by $t$ for simplicity in future derivations the problem can now be formulated as in (32).

$$\begin{aligned} &\text{minimize} \quad t f_0(\mathbf{x}) + \Phi(\mathbf{x}) \\ &\text{s.t} \qquad\qquad\quad \mathbf{Ax} = \mathbf{b} \end{aligned} \tag{32}$$

The problem (32) can be solved using Newton's method. Note that this is an approximation, and the quality of the approximation will of course depend on the parameter $t$. A reasonable solution would seem to be to select high values of $t$, but that would make the objective function difficult to minimize using Newton's method since the Hessian would vary rapidly in the feasible set's boundaries. This problem can be avoided by solving a sequence of problems in the form of (32) with an increase of $t$ from problem to problem.

The set of strictly feasible optimal points $\mathbf{x}_{sol}(t)$ for $t > 0$ is referred to as central path. Of course, each point in $\mathbf{x}_{sol}(t)$ must satisfy the optimality conditions in (33).

$$\begin{aligned} &\mathbf{A}\mathbf{x}(t) = \mathbf{b} \\ &f_i(\mathbf{x}(t)) \leq 0 \;\; i = 1, \ldots, m \\ &\boldsymbol{\lambda} \succeq 0 \\ &0 = t\boldsymbol{\nabla} f_0(\mathbf{x_{sol}}(t)) + \boldsymbol{\nabla}\Phi(\mathbf{x_{sol}}(t)) + \mathbf{A}^T\hat{\boldsymbol{\nu}} \\ &\quad = t\boldsymbol{\nabla} f_0(\mathbf{x}_{sol}(t)) + \sum_{i=1}^{m} \lambda_i \boldsymbol{\nabla} f_i(\mathbf{x}_{sol}(t)) + \mathbf{A}^T\hat{\boldsymbol{\nu}} \end{aligned} \tag{33}$$

Where the value $\lambda_i = \frac{1}{-f_i(\mathbf{x}_{sol}(t))}$ is selected. Dividing the last expression by $t$ and comparing it with the last expression in the KKT conditions (18) it is possible to say that the dual optimal points can be expressed as in (34).

$$\begin{aligned} \lambda_{sol_i}(t) = \frac{-1}{t f_i(\mathbf{x_{sol}}(t))} \;\; i = 1, \ldots, m \\ \boldsymbol{\nu}_{sol}(t) = \hat{\boldsymbol{\nu}}/t \end{aligned} \tag{34}$$

All facts considered, using the selected value of $\lambda_i$, it is possible to say that this dual optimal points are dual feasible points of the original problem. Therefore, recalling expression (16), once obtained this points it is possible to know how suboptimal the primal feasible point is. This can easily be seen by obtaining the dual function of the original problem and evaluating the optimal points found:

$$\begin{aligned} &g(\boldsymbol{\lambda}_{sol}, \boldsymbol{\nu}_{sol}) = f_0(\mathbf{x}_{sol}(t)) + \sum_{i=1}^{m} \lambda_{sol_i}(t) f_i(\mathbf{x_{sol}}(t)) + \boldsymbol{\nu_{sol}}(t)^T(\mathbf{A}\mathbf{x}_{sol}(t) - b) \\ &= f_0(\mathbf{x}_{sol}(t)) - \frac{m}{t} \end{aligned} \tag{35}$$

and therefore $f_0(\mathbf{x}_{sol}(t)) - p_{opt} \leq \frac{m}{t}$.

Without forcing the value of $\lambda_i$, a new set of modified KKT conditions that represent the problem can be defined as follows:

$$\begin{aligned} &\mathbf{A}\mathbf{x} = \mathbf{b}, \;\; f_i(\mathbf{x}) \leq 0 && i = 1, \ldots, m \\ &\boldsymbol{\lambda} \succeq 0 \\ &\boldsymbol{\nabla} f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \boldsymbol{\nabla} f_i(\mathbf{x}) + \mathbf{A}^T\boldsymbol{\nu} = 0 \\ &-\lambda_i f_i(\mathbf{x}) = 1/t && i = 1, \ldots, m \end{aligned} \tag{36}$$

#### 3.3.3.1 *Primal - dual search direction*

The modified KKT conditions in (36) can be also expressed as in (37).

$$\begin{aligned} r_t(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) &= \begin{bmatrix} r_{dual} \\ r_{cent} \\ r_{pri} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\nabla} f_0(\mathbf{x}) + \mathbf{Df}(\mathbf{x})^T \boldsymbol{\lambda} + \mathbf{A}^T \boldsymbol{\nu} \\ -\mathbf{diag}(\boldsymbol{\lambda})\mathbf{f}(\mathbf{x}) - (1/t)\mathbf{1} \\ \mathbf{Ax} - \mathbf{b} \end{bmatrix} \end{aligned} \tag{37}$$

where $\mathbf{f}(\mathbf{x})$ is a vertical vector that contains $f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})$.

As mentioned earlier, the Newton step can be extracted from the linear approximation of the optimality condition $\mathbf{r_t}(\mathbf{y} + \boldsymbol{\Delta y}) \approx \mathbf{r_t}(\mathbf{y}) + \mathbf{Dr_t}\boldsymbol{\Delta y} = \mathbf{0}$. In this specific case this implies that the newton step $(\boldsymbol{\Delta x}, \boldsymbol{\Delta \lambda}, \boldsymbol{\Delta \nu})$ can be obtained from the system in 38.

$$\begin{bmatrix} \boldsymbol{Hess}(f_0(\mathbf{x})) + \sum_{i=1}^{m} \lambda_i \boldsymbol{Hess}(f_i(\mathbf{x})) & \mathbf{Df}(\mathbf{x})^T & \mathbf{A}^T \\ -\mathbf{diag}(\boldsymbol{\lambda})\mathbf{Df}(\mathbf{x}) & -\mathbf{diag}(\mathbf{f}(\mathbf{x})) & \mathbf{0} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta x} \\ \boldsymbol{\Delta \lambda} \\ \boldsymbol{\Delta \nu} \end{bmatrix} = - \begin{bmatrix} \mathbf{r_{dual}} \\ \mathbf{r_{cent}} \\ \mathbf{r_{pri}} \end{bmatrix} \tag{38}$$

#### 3.3.3.2 *The surrogate duality gap*

Following this strategy implies that the value of $\boldsymbol{\lambda}$ is not set previous to finding the rest. Therefore, the solutions for each of the iterates when increasing the value of $t$ are not guaranteed to be feasible points of the original problem. Hence, it is not possible to evaluate the duality gap. To define a stopping criterion, a *surrogate duality gap* is defined as:

$$\hat{\eta}(\mathbf{x}, \boldsymbol{\lambda}) = -f(\mathbf{x})^T \boldsymbol{\lambda} \tag{39}$$

The surrogate duality gap would be equivalent to the duality gap if the points we are dealing with were the primal and dual feasible. Therefore, using the result obtained in (35), the parameter $t$ that corresponds to the surrogate duality gap is $t = m/\hat{\eta}$

#### 3.3.3.3 *Primal-Dual interior point method*

The primal-dual interior point method follows the steps in algorithm 3 , where the parameter $\mu$ is just a scaling factor. Note that $t$ increases with each iteration since the surrogate duality gap decreases.

Note that the primal-dual interior point method can be used to optimize a non-convex function as well, acting in this case as a local optimization algorithm.

**Algorithm 3** Primal-Dual interior point method

**Require:** $\mathbf{x}$ that satisfies $f_i(\mathbf{x}) < 0 \, i = 1, \ldots, m$, $\boldsymbol{\lambda} \succ 0$, $\mu > 1$, $\epsilon > 0$, $\epsilon_{feas} > 0$

**repeat**

1. Set $t = \mu m / \hat{\eta}$
2. Compute primal-dual step: $\boldsymbol{\Delta y}$
3. Line search and update: $\mathbf{y} = \mathbf{y} + s\boldsymbol{\Delta y}$

**until** $||\mathbf{r_{pri}}|| \leq \epsilon_{feas}$, $||\mathbf{r_{dual}}|| \leq \epsilon_{feas}$, $\hat{\eta} \leq \epsilon$

## 3.4 CONVEX OPTIMIZATION IN NON-CONVEX PROBLEMS

Among others, the solution of a convexified non-convex problem provides a good heuristic initial point for algorithms such as interior point methods applied to the non-convex problem.

# 4

# CHANNEL MODELING AND PROPAGATION

This chapters covers the essentials required to motivate the models used during this master thesis. Therefore, only the theoretical background to finally understand the model at use will be covered. Of course, more complex models that contemplate other effects are used in the field but are not required for the understanding of this thesis. Refer to [16] and [6] for more detailed explanations.

## 4.1 CHANNELS: STOCHASTIC LINEAR TIME VARIANT SYSTEMS

The wireless channel can be described as a linear filter, meaning that the relation between the transmitted signal $x(t)$ and the received signal $y(t)$ can be expressed by the convolution of $x(t)$ and the channel's impulse response, as in equation (40). This impulse response is time varying due to the random behavior of the wireless channel. Therefore, the channel's impulse response is referred to as $h(t, \tau)$, where $t$ expresses the channel's impulse response $h(\tau)$ variation in time. As a matter of fact, it is possible to interpret the variable $t$ as a parametrization of the impulse response, meaning that it is selecting which $h(\tau)$ is valid at that specific instant.

$$y(t) = \int_{-\infty}^{\infty} x(t-\tau)h(t,\tau)\,d\tau + n(t), \tag{40}$$

where $n(t)$ is additive white Gaussian noise.

For the next section, $h(t, \tau)$ is assumed to follow the equation $h(t, \tau) = h(t)\delta(\tau)$ for simplicity. An interpretation of what this represents in the wireless channel will be provided in section 4.1.2. Once this has been assumed, the received signal is now expressed as a function of the channel and the transmitted signal as in equation (41).

$$y(t) = h(t)x(t) + n(t) \tag{41}$$

Note that, in order for the transmission to be possible the variation of $h(t)$ has to be slower than the symbol duration since otherwise the channel would not allow data transmission.

As mentioned, $h(t)$ is a random variable but it is possible to model its behavior when one considers the causes that generate it.

### 4.1.1 *Models for the prediction of power*

The knowledge of the received power is very relevant since it will allow the system to guarantee a specific quality of service. $|h(t)|^2$ represents the ratio between the received power of the signal of interest $P_{RX}$ and the transmitted power $P_{TX}$.

$$|h(t)|^2 = \frac{P_{RX}}{P_{TX}} \tag{42}$$

The value of $|h(t)|^2$ is modeled using three different scales of variation or fading:

1. Path loss: $h_{ploss}(d)$
2. Large scale fading: $h_{large}(d)$
3. Small scale fading: $h_{small}(d)$

#### 4.1.1.1 *Path loss*

Path loss is a *deterministic* effect that mainly depends on the distance between the antennas. Of course different models can be used and the more data from the surrounding one gets and tries to fit into the model the more accurate it will be. As a general idea, due to path-loss, $\frac{P_{RX}}{P_{TX}} \propto \frac{1}{d^4}$ where $d$ represents the distance between the antennas.

Considering the deterministic behavior of this parameter is not the main focus of interest when trying to estimate the channel.

#### 4.1.1.2 *Large scale fading*

Large scale fading represents the shadowing that large elements may create over one of the terminals. Suppose a mobile terminal is moving in a circle that has the base station, with omnidirectional antennas, for center. At the same distance of the base station, the received signal will not have the same power at all points due to large obstacles. Large scale fading can be modeled using a log-normal distribution as in equation (43).

$$f(h_{large_{dB}}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(h_{large_{dB}} - h_{ploss}(d))^2}{2\sigma^2}} \tag{43}$$

where $h_{large_{dB}} = 10\log(|h_{large}|^2)$ and the variance $\sigma^2$ depends on the kind of surrounding[1]. Combining path-loss and large scale fading we have that given a specific distance between antennas $d$ the received power follows a log-normal distribution, due to large scale fading with average $10\log(|h_{ploss}(d)|^2)$, i.e., as a constant $10\log(|h_{ploss}(d)|^2)$ plus a random variable with zero average that follows a log-normal distribution. Note that the dependence of $h_{large}$ with the distance $d$ is due to path loss.

1 Urban, rural, etc.

#### 4.1.1.3 *Small scale fading*

Assume that the receiver is at a specific distance $d$ of the base station. In a specific point, the received power can vary due to multiple reflections, with different amplitudes and phases, the signal may suffer on objects that are close to the receiving terminal. The channel $h_{small}$ is defined in baseband as in (44).

$$h_{small} = h_{re} + jh_{im} \tag{44}$$

Considering the signal arrives with a great number of local reflections, it is possible to consider that it is the sum of many identically distributed random amounts. This justifies the use of the central limit theorem, meaning that is is possible to consider $h_{re}$ and $h_{im}$ equally distributed Gaussian random variables. It is easy to verify from here, that $|h_{small}|$ follows a Rayleigh distribution as in equation (45) and $\phi_{small} = \sphericalangle(h_{small})$ follows a uniform distribution (46).

$$f(h_{small}) = \frac{|h_{small}|}{\sigma^2} e^{-\frac{|h_{small}|^2}{\sigma^2}} \tag{45}$$

Where $\sigma^2 = E\{|h_{small}|^2\} = h_{large}(d)|_{linear}$.

$$f(\phi_{small}) = \frac{1}{2\pi}, -\pi \leq \phi_{small} \leq \pi \tag{46}$$

Additionally, the different objects around the terminal can either move, or the terminal itself can move distances that are not relevant to the other two effects. Therefore, an additional kind of modeling will be required later on to model the variation of the channel's impulse response with time. A brief explanation of this model is provided in section 4.1.3.

### 4.1.2 *Narrowband and wideband channels*

Recall the initial formulation of the channel's impulse response in equation (40) and assume that the channel is now time invariant and therefore the impulse response of the channel can be expressed as $h(\tau)$. This impulse response models the behavior of the channel given that several multipath components are arriving with different delays $\tau$ to the receiver. These multipath components correspond to paths of different lengths, meaning that the beams following a larger path will of course suffer of a larger delay.

The receiver observes a superposition of all the delayed components that may have suffered of different attenuations and phase shifts. This effect can be modeled as illustrated in equation (47).

$$h(\tau) = \sum_{n=0}^{N-1} a_n e^{j\phi_n} \delta(\tau - \tau_n) \tag{47}$$

where N is the number of pats, $a_n$, $\phi_n$ and $\tau_n$ are the amplitudes, phases and delays of each of the paths respectively. Following the same reasoning as previously, given the amount of paths that contribute to each of the N paths listed above, $a_n$ will follow a Gaussian distribution while $\phi$ will follow a uniform one.

Once this model has been defined, two different cases must be treated differently depending on the symbol time.

#### 4.1.2.1 *Wideband channels*

When the symbol time $T_s$ is smaller than the temporal duration of the channel's impulse response $\tau_{N-1}$, the convolutive effect on the symbol results in a sequence of length $T_s + \tau_{N-1}$, which has a dramatic impact on the next symbol or symbols. This effect is referred to ISI.

Using a smaller symbol time has the main advantage that it allows a high data rate if the transmission is successful.

#### 4.1.2.2 *Narrowband channels*

When the symbol time $T_s$ is very large compared to the temporal duration of the channel's impulse response $\tau_{N-1}$ the convolutive effect on the symbol results in a sequence of length $T_s + \tau_{N-1}$. Considering that $T_s >> \tau_{N-1}$ the next symbol will hardly be affected by the overlap with the remains of the current one. Therefore, a narrowband channel can be modeled as in equation (48).

$$h_{NB}(\tau) = Ae^{j\phi}\delta(\tau) \tag{48}$$

Where $Ae^{j\phi} = h_{NB_{RE}} + jh_{NB_{IM}}$. Using again the central limit theorem both real and imaginary parts will follow the same random Gaussian distribution.

The advantage of narrowband channels is mainly that there is no ISI. On the other hand, using a larger symbol time implies that the transmission is also slower. This can be countered by the use of several subcarriers like in OFDM where each of the subcarriers transmits through a narrow band channel, obtaining a high throughput but without suffering from strong ISI.

Narrowband channels are also referred to as frequency flat for obvious reasons. From now on this model will be assumed for the entire master thesis.

### 4.1.3 *Models for the variation in time*

The movement of scattering objects around the receiver can be modeled as the movement of the terminal itself for simplicity. Assuming that the mobile terminal is moving with direction and speed indicated by $\mathbf{v}$ and that it receives a plane wave with carrier frequency $f_C = \frac{\lambda}{c}$ from direction $\mathbf{i}$.

Considering that the receiver initially received a signal proportional to $\cos(2\pi f_c t)$, it will receive after time $\Delta t$ a signal proportional to $\cos(2\pi f_c(t+\Delta t) + \frac{2\pi}{\lambda} v \Delta t \cos(\alpha))$, where $\alpha$ is the angle defined between the movement direction defined in $\mathbf{v}$ and the direction of wave propagation $\mathbf{i}$. Therefore, a mobile terminal perceives a shift in frequency that can be expressed as in (49).

$$f_{perc} = f_c + f_c \frac{v}{c} \cos(\alpha) \tag{49}$$

Where the last element is the Doppler shift. Considering the origin of the small scale fading explained previously, a large amount of paths will be arriving from different angles to the moving receiving terminal at the same time. Therefore, the effects of this multipath components are modeled statistically. For this purpose the Doppler Spectrum $G_s(f_d)$, which models how much power is received from each of the frequency shifts, is used. The Doppler spectrum is expressed as in 50.

$$G_s(f_d) = \frac{1}{\sqrt{f_{d_{max}}^2 - f_d^2}} [p(\alpha)G(\alpha) + p(-\alpha)G(-\alpha)] \tag{50}$$

Where $f_d$ is the Doppler shift, $f_{d_{max}}$ is the maximum possible Doppler shift and happens for $\alpha = 0$, $p(\alpha)$ is the power of the wave arriving to the receiver from angle $\alpha$ and $G(\alpha)$ is the receiving antenna gain in the direction defined by the angle $\alpha$.

The channel's time correlation $R_h(\Delta t) = E\{h(t)h^*(t+\Delta t)\}$ can be calculated from the Doppler Spectrum by using the Inverse Fourier transform (51).

$$R_h(\Delta t) = \int_{-f_{d_{max}}}^{f_{d_{max}}} G_s(f_d) e^{-j2\pi f_d \Delta t} \, df_d \tag{51}$$

This autocorrelation function provides information of how similar the channel $h(t)$ is to itself after a specific time $\Delta t$. As any other autocorrelation function, $R_h(\Delta t)$ has its maximum when $\Delta t = 0$. The *coherence time* is defined as the time $T_\tau$ it takes the correlation function to suffer a 3 dB decay, i.e.

$$R_h(T_\tau) = R_h(0)/2 \tag{52}$$

### 4.1.4 *Modeling a MIMO Channel*

A MIMO channel uses several antennas in transmission and reception. Therefore, it can mathematically be modeled as in equation (53).

$$\mathbf{y}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{n}(t) \tag{53}$$

Where $\mathbf{y}(t)$, $\mathbf{x}(t)$ and $\mathbf{n}(t)$ are the received signal, transmitted signal and interfering signal respectively. The term $\mathbf{n}(t)$ models the impact

of noise and interference from other communication links. It is assumed that it is independent of the transmitted signal $\mathbf{x}(t)$. Finally, $\mathbf{H} \in \mathbb{C}^{n_T \times n_R}$ is the current channel realization and is assumed to be constant during the channel's coherence time and independent from the following coherence time. This is referred to as *block fading*.

Many different approaches can be considered when intending to model a MIMO channel. In this master thesis, the Kronecker model is used. The Kronecker model is a stochastic correlation based model. In this model it is assumed that two points that are relatively close in space are related by a correlation coefficient. Considering a narrowband channel and that each of the coefficients is modeled as a random Gaussian variable, it is possible to represent a MIMO channel by a matrix $\mathbf{H} \in \mathbb{C}^{n_r \times n_t}$ in which each of the coefficients represent the channel between the two antenna pair that correspond to the indexes in the matrix. Then, the channel's correlation matrix can be defined as in equation (54).

$$\mathbf{R} = E\{\text{vec}(\mathbf{H})\text{vec}^H(\mathbf{H})\} \tag{54}$$

The Kronecker model assumes the following on the channel's correlation matrix:

1. The correlation factor of two different antennas, either receiving or transmitting, are the same independently of where they are calculated. This implies that the correlation of two links from two different transmit antennas to *any* receive antenna is the same, i.e. $\rho_{i,j}^{Tx} = E\{h_{i,n}h_{j,n}^*\}\, n = 1, \ldots, n_r$ and $\rho_{i,j}^{Rx} = E\{h_{m,i}h_{m,j}^*\}\, m = 1, \ldots, n_t$, where $\rho_{i,j}^{Tx}$ and $\rho_{i,j}^{Rx}$ stay for the correlation coefficient in transmission and reception respectively. This way it is possible to build a reception correlation matrix $\mathbf{R_R}$ and a transmission correlation matrix $\mathbf{R_T}$, which of course are hermitian and positive semidefinite.

2. The correlation is separable, meaning that it can be separated as the product of two different correlation coefficients. From a physical perspective this means that the behavior due to scatterers in one side has nothing to do with the behavior in the other.

Using this two assumptions, the channel's correlation matrix can be described by the Kronecker product as in equation (55).

$$\mathbf{R} = \mathbf{R_T}^T \otimes \mathbf{R_R} \tag{55}$$

Additionally, considering that each of the link coefficients $h_{m,n}$ is Gaussian distributed, the Kronecker channel can be synthesized as in equation .

$$\mathbf{H}_{kron} = \mathbf{R}_R^{1/2}\mathbf{H}_{iid}\mathbf{R}_T^{1/2} \tag{56}$$

Where $\mathbf{H}_{iid}$ is a matrix with independent identically distributed complex Gaussian random variables with 0 mean and 1 variance.

## 4.2 CHANNEL ESTIMATION

From the previous it is shown that modeling the channel helps to predict the received power as a function of the transmitted power, and even the time variation of the channel. For many applications, some of which will be mentioned later on in the master thesis, it is very useful to know the exact channel coefficients and not just a statistical characterization of them. To obtain this information, a pilot sequence that both transmitter and receptor know is sent and using the impact the channel had on the pilot sequence an estimation of the channel is made. Of course, modeling the channel provides a ground base knowledge that allows to estimate the channel more efficiently.

### 4.2.1 *Pilot sequence*

As mentioned earlier, the goal of this master thesis is to find a pilot sequence that suits the necessities of the system. From now on, the pilot sequence will be referred to as $\mathbf{P} \in \mathbb{C}^{n_T \times B}$ where $B$ is the length of the training sequence. As mentioned earlier, the wireless channel has a coherence time within which it can be considered not to vary. Therefore, this pilot sequence needs to be transmitted, the channel estimate has to be done, and then this channel estimate is used during the *data transmission phase*, all within a coherence time.

### 4.2.2 *Training phase*

During the training phase the pilot sequence is transmitted and the channel estimate is obtained. The received sequence when the pilot signal $\mathbf{P}$ is transmitted can be mathematically expressed as in equation (57).

$$\mathbf{Y} = \mathbf{HP} + \mathbf{N} \tag{57}$$

Where $\mathbf{Y} = [\mathbf{y}(1) \dots \mathbf{y}(B)] \in \mathbb{C}^{n_r \times B}$ and $\mathbf{N} = [\mathbf{n}(1) \dots \mathbf{n}(B)] \in \mathbb{C}^{n_r \times B}$. In order to express the received sequence in vectorial form, it is necessary to define $\tilde{\mathbf{P}} = \mathbf{P}^T \otimes \mathbf{I}$, and then the expression in equation 57 can be expressed as in equation (58).

$$\text{vec}(\mathbf{Y}) = \tilde{\mathbf{P}}\text{vec}(\mathbf{H}) + \text{vec}(\mathbf{N}) \tag{58}$$

It is assumed that $\text{vec}(\mathbf{N}) \in \mathcal{CN}(\mathbf{0}, \mathbf{S})$ where the covariance matrix $\mathbf{S}$ is Kronecker structured, and can be expressed as follows:

$$\mathbf{S} = \mathbf{S}_Q^T \otimes \mathbf{S}_R \tag{59}$$

which following the reasoning in section 4.1.4, assumes that the interference's covariance in time, $\mathbf{S}_Q$, has a behavior that is independent of its behavior in space, $\mathbf{S}_R$.

### 4.2.3 *Optimal estimator*

Assuming all the correlation matrices in the previous section are known, and knowing that both the channel coefficients, due to Rayleigh fading, and the interference follow a Gaussian distribution; the MAP (Maximum a Posteriori) and the MMSE estimate are identical as shown in [17]. The minimum mean square error estimator follows the equation in (60) and has the variance expressed in (61).

$$\text{vec}(\hat{\mathbf{H}}_{MMSE}) = (\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tilde{\mathbf{P}})^{-1} \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tag{60}$$

$$\mathbf{C}_{MMSE} = (\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tilde{\mathbf{P}})^{-1} \tag{61}$$

### 4.2.4 *Data transmission phase*

During the data transmission phase the channel estimate is used. As mentioned, it is necessary that it lies within the channel's coherence time for the estimate to be valid. For simplicity, in this master thesis the transmitted signal during the data transmission phase is assumed to be 0 mean and temporally and spatially white,i.e. with spectrum $\mathbf{\Psi}(\omega) = \lambda_X \mathbf{I}$.

# 5

# STATE OF THE ART: OPTIMIZED PILOT SEQUENCES FOR MIMO CHANNEL ESTIMATION

Given the need of accurate instantaneous Channel State Information (instantaneous CSI) in communications systems, extended research has been carried out in order to obtain it as accurately as possible without severely impacting the system's performance. The estimation of the instantaneous CSI is carried out by sending a pilot sequence that is known both by the transmitter and the receiver. This way the channel's impact on the pilot signal is used to obtain an estimation of the instantaneous CSI. It is important to notice, that the training phase is immediately followed by a data transmission phase in which the obtained information about the channel is applicable. Hence, both the training and the data transmission phase should be within the channel's coherence time.

In [8] an analysis of the optimal training sequence length considering the channel's coherence time, the training phase SNR and the number of antennas on both sides is considered, concluding that to obtain a meaningful estimate of the channel, the training sequence length should be equal to the number of transmit antennas, hence, obtaining one measurement for each unknown. This analysis is capacity focused and therefore does not consider a specific technique when it comes to obtaining the training sequence.

Two different alternatives are considered when estimating the channel. In blind channel estimation no explicit pilot sequences are used and the information is extracted of large data records, obtained after data transmission, and the channel model. On the other hand, training based techniques do not require large data records. They will require a training phase that will limit the data transmission time. A significant decrease in the training phase time can be obtained when using the the statistical CSI. Techniques that use it will require an estimation of it, and errors in these estimates can severely impact the final performance. In this section and thesis only techniques that require statistical information about the channel will be considered. Additionally, it is assumed that the statistical CSI is already provided and has no errors.

As mentioned in chapter 4, when the channel's fading is frequency selective, OFDM is generally used since then the problems created by

the multi-path channel can be handled efficiently. References on how the optimal training sequence is obtained when a MIMO-OFDM system is at hand can be found in [1], where the training OFDM symbols are required to be equi-powered, equi-spaced and phase-shift orthogonal to minimize the channel's estimate MSE, subject to a fixed power $\mathcal{P}$; and [15], where more restrictions such as the signal's PAR (Peak to Average Ration) of the training sequence are considered.

When dealing with a flat fading channel and using statistical CSI to obtain the current channel estimates two main approaches can be found in the current state of the art. These are minimizing the channel's estimate mean square error and maximizing the mutual information between the channel and the received signal during the training phase. It is important to note that even given the optimal training sequence and channel estimator, the resulting channel estimate will still be a random variable and will therefore possess certain variance. Systems that consider that the channel estimate is not perfect are explored in the state of the art as well but are not covered in this master thesis.

## 5.1 MAXIMIZING THE MUTUAL INFORMATION

In this specific case the metric to maximize is the mutual information between the received signal and the channel estimate as a function of the training sequence $\mathbf{P}$. Considering the model described in chapter 4, the mutual information can be expressed as in (62)

$$\mathcal{I}(\text{vec}(\mathbf{H}); \text{vec}(\mathbf{Y})|\mathbf{P}) = \mathcal{H}(\text{vec}(\mathbf{H})) - \mathcal{H}(\text{vec}(\mathbf{H})|\text{vec}(\mathbf{Y}), \mathbf{P}) \tag{62}$$

which can, at the same time, be expressed as in (63).

$$\text{logdet}(\mathbf{I} + \mathbf{R}\tilde{\mathbf{P}}\mathbf{S}^{-1}\tilde{\mathbf{P}}^{H}) \tag{63}$$

Maximizing (63) is equivalent to minimizing the second term in (62) which can be interpreted as minimizing the uncertainty of $\text{vec}(\mathbf{H})$ given the knowledge of $\text{vec}(\mathbf{Y})$, by adequately designing $\mathbf{P}$.

More detailed derivations of this can be found in [12] and [2].

In [12] the noise is assumed to be white, i.e. $\mathbf{S} = \sigma^2\mathbf{I}$ . In this specific case the problem can be expressed as a function of matrix $\mathbf{P}\mathbf{P}^H$ and can be solved using convex optimization [2] by using one of the methods in [14]. Additionally, in [2] a relaxation on the metric (63) is suggested to make the problem convex.

## 5.2 MINIMIZING THE MEAN SQUARE ERROR

In [12] the authors consider as well minimizing the channel's estimate mean square error. This metric can be expressed as in (64)

$$\text{MMSE} = \mathbb{E}\left\{\text{vec}(\tilde{\mathbf{H}})^H\mathbf{I}\,\text{vec}(\tilde{\mathbf{H}})\right\} \tag{64}$$

or as a function of $\mathbf{P}$ as stated in (65).

$$\text{MMSE} = \text{trace}(\mathbf{R}^{-1} + \tilde{\mathbf{P}}\mathbf{S}^{-1}\tilde{\mathbf{P}}^{H})^{-1} \tag{65}$$

According to [12] when $\mathbf{S} = \sigma^2\mathbf{I}$ both maximizing the mutual information and minimizing the MSE lead to water-filling solutions with different proportions in the active elements during the training phase, which is not as surprising considering the close relation between (63) and the MSE [7]. In both [9] and [13] the training sequence is obtained by minimizing the estimation's mean square error when the transmission is interference-limited. Finally, in [3] an analysis of the structure of $\mathbf{P}$ is done when the transmission suffers of different disturbances.

Part III

# APPLICATION-ORIENTED TRAINING SEQUENCE DESIGN

# 6

# MOTIVATION

The knowledge of statistical CSI provides the system of the capability of tailoring the training sequence **P** to obtain a better estimation of the channel's realization. This is already exploited in the techniques mentioned in the state of the art section which have their main advantage in a reduced training sequence length and/or smaller variance, if the provided statistical CSI is correct. However, these techniques do not consider if some of the efforts made will cause a significant impact on the system.

It is possible that minimizing the mean square error of the entire channel estimate does not significantly affect on the components that later on will have a dramatic impact on the system when the channel estimate is used. Therefore, it seems reasonable to consider for what specific purpose the channel estimate is required and to design a metric that is representative of the performance of the channel estimate in this position.

In this master thesis, as in [10], the metrics considered are the MMSE for the sake of comparison and testing due to its simplicity, a metric designed for the Zero Forcing pre-coder and a metric designed for the MMSE Equalizer.

# 7

# PROBLEM FORMULATIONS

## 7.1 GPP: GUARANTEED PERFORMANCE PROBLEM

As its name indicates the Guaranteed Performance Problem guarantees a performance $\gamma$ with a specific probability $1-\epsilon$ while trying to minimize the training sequence power. The problem can therefore be expressed as in (66)

$$\begin{aligned} &\underset{\mathbf{P}}{\text{minimize}} && \text{tr}(\mathbf{P}\mathbf{P}^H) \\ &\text{s.t} && \text{P}\left\{\text{J}(\tilde{\mathbf{H}},\mathbf{H}) \leq \frac{1}{\gamma}\right\} \geq 1-\epsilon, \end{aligned} \tag{66}$$

where $\text{J}(\tilde{\mathbf{H}},\mathbf{H})$ is a measure of the performance degradation due to the estimation error. The problem can be solved requiring an average performance instead, being then expressed as stated in (67).

$$\begin{aligned} &\underset{\mathbf{P}}{\text{minimize}} && \text{tr}(\mathbf{P}\mathbf{P}^H) \\ &\text{s.t} && \mathbb{E}\left\{\text{J}(\tilde{\mathbf{H}},\mathbf{H})\right\} \leq \frac{1}{\gamma} \end{aligned} \tag{67}$$

Considering that, as mentioned in [10], these training signal design problems are highly non-linear and non-convex, only metrics that can be well approximated by a quadratic term in $\tilde{\mathbf{H}}$, i.e. $\text{J}(\tilde{\mathbf{H}},\mathbf{H}) \approx \text{vec}^H(\tilde{\mathbf{H}})\boldsymbol{\mathcal{I}}_{adm}\text{vec}(\tilde{\mathbf{H}})$ are considered for tractability.

This leads to being able to, in the specific case of the Guaranteed Average Performance Problem, writing $\mathbb{E}\left\{J(\tilde{\mathbf{H}},\mathbf{H})\right\}$ as in (68).

$$\mathbb{E}\left\{J(\tilde{\mathbf{H}},\mathbf{H})\right\} = \text{tr}\{\boldsymbol{\mathcal{I}}_{adm}\mathbf{C}_{MMSE}\} \tag{68}$$

where[1] $\mathbf{C}_{MMSE} = \left(\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}}\right)^{-1}$.

In order to be able to solve the problems with a specific probability further approximations are required and are dealt with in the following sections.

1 Recall optimality of MMSE estimator mentioned in chapter 4.

### 7.1.1 *Confidence Ellipsoids approximation*

The idea of the Confidence Ellipsoid approximation is to define a uncertainty set that will contain specific error vectors $\text{vec}(\tilde{\mathbf{H}})$ with probability $\alpha$, which is designed to be equivalent to $1-\epsilon$. This can be expressed as in (69).

$$\mathcal{D}_{\mathcal{D}} = \left\{ \tilde{\mathbf{H}} : vec^H(\tilde{\mathbf{H}})\, \mathbf{C}_{MMSE}^{-1}\, vec(\tilde{\mathbf{H}}) \leq \frac{\chi_\alpha^2(2n_r n_t)}{2} \right\} \tag{69}$$

Intuitively this can be interpreted as a restriction over the square of the Mahalanobis distance [2] from $\text{vec}(\tilde{\mathbf{H}})$ to $\mathbf{0}$. Considering the fact that the distribution of points around the point $\mathbf{0}$ is ellipsoidal and that the error on a direction that has higher variance might have a larger magnitude, it is reasonable that the distance from $\text{vec}(\tilde{\mathbf{H}})$ to $\mathbf{0}$ is weighted by $\mathbf{C}_{MMSE}^{-1}$. This way, by limiting the square of the Mahalanobis distance, the set of errors that may lead to the required performance is smaller, but those that have a higher projection in a direction in which the variance is high are not disregarded. Once applied this idea to (66) the problem can be written as in (70).

$$\begin{aligned} \underset{\mathbf{P}}{\text{minimize}} \quad & \text{tr}(\mathbf{P}\mathbf{P}^H) \\ \text{s.t.} \quad & \text{J}(\tilde{\mathbf{H}}, \mathbf{H}) \leq \frac{1}{\gamma},\ \forall\, \tilde{\mathbf{H}} \in \mathcal{D}_{\mathcal{D}} \end{aligned} \tag{70}$$

At this point the feasible set contains only those estimation errors $\tilde{\mathbf{H}}$ that go that have a Mahalanobis distance equal or below $\frac{\chi_\alpha^2(2n_r n_t)}{2}$, which occurs with probability $\alpha$, and at the same time fulfill the performance restriction.

Considering only metrics that can be written as a quadratic function of the error, the set of $\tilde{\mathbf{H}}$ that fulfill $\text{J}(\tilde{\mathbf{H}}, \mathbf{H}) \leq \frac{1}{\gamma}$ can be described, as mentioned in [10], by the ellipsoid in (71).

$$\mathcal{D}_{adm} = \left\{ \tilde{\mathbf{H}} : \text{vec}(\tilde{\mathbf{H}})^H \gamma \boldsymbol{\mathcal{I}}_{\boldsymbol{adm}} \text{vec}(\tilde{\mathbf{H}}) \leq 1 \right\} \tag{71}$$

Using all the mentioned approximations the problem can be expressed as in equation (72).

$$\begin{aligned} \underset{\mathbf{P}}{\text{minimize}} \quad & \text{tr}(\mathbf{P}\mathbf{P}^H) \\ \text{s.t.} \quad & \mathcal{D}_D \subseteq \mathcal{D}_{adm} \end{aligned} \tag{72}$$

The inclusion of the confidence ellipsoid in the metric ellipsoid guarantees that the resulting outage probability $\epsilon_{real}$ will be bounded $\alpha \geq 1 - \epsilon_{real}$, implying that the desired outage probability $\epsilon \geq \epsilon_{real}$.

This can be seen if it is considered that, first of all, belonging to the ellipsoid $\mathcal{D}_D$ is an event that occurs with probability $\alpha$. Secondly, that

[2] $\underset{Mah}{d}(\mathbf{x}, \mathbf{y})^2 = (\mathbf{x}-\mathbf{y})^H \mathbf{C}^{-1} (\mathbf{x}-\mathbf{y})$

this set of events that happen with probability $\alpha$ is included in a ellipsoid that guarantees the metric, implies that the metric is guaranteed at least with probability $\alpha$.

Following the steps in [10] the problem can be written as in (73).

$$\begin{aligned} \underset{\mathbf{P}}{\text{minimize}} \quad & \text{tr}(\mathbf{P}\mathbf{P}^H) \\ \text{s.t.} \quad & \mathbf{R}^{-1} + \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tilde{\mathbf{P}} \succeq \frac{\gamma \chi^2_\alpha(2n_r n_t)}{2} \boldsymbol{\mathcal{I}}_{\boldsymbol{adm}} \end{aligned} \tag{73}$$

In this last formulation, in the same way as in (68), the restriction is written as a function that is quadratic in $\tilde{\mathbf{P}}$. Once the problem is formulated as in (73) it is noticeable that the final restriction in performance $\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tilde{\mathbf{P}} \succeq \frac{\gamma \chi^2_\alpha(2n_r n_t)}{2} \boldsymbol{\mathcal{I}}_{\boldsymbol{adm}}$ represents the search for a training sequence $\mathbf{P}$ that forces a specific shape on the estimator's covariance matrix that will cause the estimation error to have a small Mahalanobis distance in the directions that are relevant to the metric defined by $\boldsymbol{\mathcal{I}}_{\boldsymbol{adm}}$. As a matter of fact, for the restriction to be tight it is required that the inverse covariance matrix is a scaled version of the metric. Therefore, the quality of the feasible set defined in the Confidence ellipsoid case depends on how hard it is to adapt the shape of the inverse covariance matrix to the metric, i.e. how difficult it is to jointly find the two ellipsoids.

#### 7.1.1.1 *Analytical Solutions*

Under specific circumstances an analytical solution can be found for the problem stated above, as described in [10]. Specifically in this case, it is required, additional to all the restrictions described in the preliminaries, that $\boldsymbol{\mathcal{I}}_{\boldsymbol{adm}} = \boldsymbol{\mathcal{I}}_T^T \otimes \boldsymbol{\mathcal{I}}_R$. A summary of the solutions in different cases will be provided here, refer to [10] for the proofs. For the sake of simplicity $c$ is defined as $c = \frac{\gamma \chi^2_\alpha(2n_r n_t)}{2}$. In all the cases described in table 1 the optimal training sequence can be written as in (74).

$$\mathbf{P}_{opt} = \mathbf{U}_B \mathbf{D}_P \mathbf{U}_Q^H \tag{74}$$

where $\mathbf{U}_Q$ and $\mathbf{D}_Q$ are $\mathbf{S}_Q$'s modal matrix and eigenvalues respectively, $\mathbf{U}_B$ is the modal matrix[3] of a matrix $\mathbf{B}$, that will vary from case to case, and $\mathbf{D}_P(i,i) = \sqrt{\mathbf{D}_B(i,i)\mathbf{D}_Q(i,i)}$ with $i = 1, \ldots, \min(n_T, B)$. The values in $\mathbf{D}_Q$ and $\mathbf{D}_B$ are placed in *decreasing* and *increasing* order respectively. Note that according to theorem 1 in [10], solutions in which the columns of $\mathbf{U}_B$ and $\mathbf{S}_Q$ are multiplied by complex unit-norm scalars are equivalent if the eigenvalues of $\mathbf{B}$ and $\mathbf{S_Q}$ are distinct and strictly positive.

3 $\mathbf{S}_Q = \mathbf{U}_Q \mathbf{D}_Q \mathbf{U}_Q^H$ and $\mathbf{B} = \mathbf{U}_B \mathbf{D}_Q \mathbf{U}_B^H$

1. $\mathbf{R}_R = \mathbf{S}_R$ and $B \geq \text{rank}([c\lambda_{max}(\mathbf{S}_R\boldsymbol{\mathcal{I}}_R)\boldsymbol{\mathcal{I}}_T - \mathbf{R}_T^{-1}]_+)$: the training sequence $\mathbf{P}_{opt}$ can be expressed as (74) with $\mathbf{B} = [c\lambda_{max}(\mathbf{S}_R\boldsymbol{\mathcal{I}}_R)\boldsymbol{\mathcal{I}}_T - \mathbf{R}_T^{-1}]_+$.
2. $\mathbf{R}_R = \boldsymbol{\mathcal{I}}_R$ and $B \geq \text{rank}([c\boldsymbol{\mathcal{I}}_T - \mathbf{R}_T^{-1}]_+)$: the training sequence $\mathbf{P}_{opt}$ can be written as (74) with $\mathbf{B} = \lambda_{max}(\mathbf{S}_R\boldsymbol{\mathcal{I}}_R)[c\boldsymbol{\mathcal{I}}_T - \mathbf{R}_T^{-1}]_+$.
3. Finally if $\mathbf{R}_T^{-1} = \boldsymbol{\mathcal{I}}_T$ and $B \geq \text{rank}(\boldsymbol{\mathcal{I}}_T)$, the optimal training sequence can be expressed as (74) with $\mathbf{B} = \lambda_{max}(\mathbf{S}_R[c\boldsymbol{\mathcal{I}}_R - \mathbf{R}_R]_+)\boldsymbol{\mathcal{I}}_T$.

Table 1.: Available analytical solutions for GPP using confidence ellipsoids.

### 7.1.2 *Markov Bound approximation*

The Markov inequality states that if a random variable $x$ is nonnegative and $x_0 > 0$, then $\text{P}(x \geq x_0) \leq \frac{\mathbb{E}(x)}{x_0}$.

Applying this inequality to (66), as illustrated in [11], the problem can be approximated as in (75).

$$\begin{aligned} &\underset{\mathbf{P}\in\mathbf{C}^{n_T\times B}}{\text{minimize}} && \text{tr}(\mathbf{P}\mathbf{P}^H) \\ &\text{s.t.} v && \text{tr}(\boldsymbol{\mathcal{I}}_{\boldsymbol{adm}}(\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}})^{-1}) \leq \frac{\epsilon}{\gamma} \end{aligned} \tag{75}$$

It is important to note, that in this case, a GPP using Markov Bound approximation with a performance requirement of $\gamma_0$ and a probability requirement of $\epsilon_0$ is equivalent to a problem with 1 for probability requirement and performance requirement of $\gamma' = \frac{\gamma_0}{\epsilon_0}$, and therefore the parameter $\epsilon$ which should directly represent the outage probability does not due to the use of Markov's inequality. Additionally, note that the Guaranteed Average Performance Problem can be seen as a GPP using Markov Bound approximation with $\epsilon = 1$.

The restriction in (75) can be rewritten using (131) as

$$\text{vec}^H(\boldsymbol{\mathcal{I}}_{\boldsymbol{adm}}{}^H)\text{vec}((\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}})^{-1}) \leq \frac{\epsilon}{\gamma} \tag{76}$$

meaning that the aim is to reduce the projection of the estimator's covariance matrix over the metric. Keeping this fact in mind, it is possible to suggest that the set of feasible $\mathbf{P}$ when using the Markov Bound approximation has a less restrictive shape than when using Confidence Ellipsoids. This is due to the fact that in the Confidence Ellipsoid case sequences $\mathbf{P}$ that allow the inverse of the covariance matrix to have the same shape as $\boldsymbol{\mathcal{I}}_{\boldsymbol{adm}}$ are in the feasible set. On the other hand, when using the Markov Bound approximation, sequences $\mathbf{P}$ that allow the covariance matrix to have a relatively small projection on $\boldsymbol{\mathcal{I}}_{\boldsymbol{adm}}$ are considered.

On the other hand, it is also necessary to consider that the probability restriction in the Confidence Ellipsoids case actually considers the shape of the distribution of $\text{vec}(\tilde{\mathbf{H}})$, while in the Markov Bound case it does not. All facts considered, the experiment in chapter 10 has been carried out to verify this theoretical conclusions.

## 7.2 MPP: MAXIMIZED PERFORMANCE PROBLEM

In a Maximized Performance Problem, the restriction is set on the training signal power instead of the performance. Meaning that the aim is to maximize the performance given specific training power budget. An MPP can be expressed as in (77).

$$\begin{aligned} \underset{\mathbf{P}\in\mathbb{C}^{n_T\times B}}{\text{maximize}} \quad & \gamma \\ \text{s.t.} \quad & \mathrm{P}\left\{\mathrm{J}(\tilde{\mathbf{H}},\mathbf{H}) \leq \frac{1}{\gamma}\right\} \geq 1-\epsilon \\ & \text{tr}(\mathbf{P}\mathbf{P}^H) \leq \mathcal{P} \end{aligned} \tag{77}$$

Analogously as in the GPP case the metrics considered will be those that can be expressed or properly approximated by a form that is quadratic with $\text{vec}(\tilde{\mathbf{H}})$.

All procedures and argumentations are analogous to the GPP case, so in the two following sections only the result is provided for the MPP case.

### 7.2.1 *Confidence Ellipsoids approximation*

$$\begin{aligned} \underset{\mathbf{P}\in\mathbb{C}^{n_T\times B}}{\text{maximize}} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}} \succeq \frac{\gamma\chi^2_\alpha(2n_rn_t)}{2}\boldsymbol{\mathcal{I}}_{adm} \\ & \text{tr}(\mathbf{P}\mathbf{P}^H) \leq \mathcal{P} \end{aligned} \tag{78}$$

### 7.2.2 *Markov Bound approximation*

$$\begin{aligned} \underset{\mathbf{P}\in\mathbb{C}^{n_T\times B}}{\text{maximize}} \quad & \gamma \\ \text{s.t.} \quad & \text{tr}(\boldsymbol{\mathcal{I}}_{adm}(\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}})^{-1}) \leq \frac{\epsilon}{\gamma} \\ & \text{tr}(\mathbf{P}\mathbf{P}^H) \leq \mathcal{P} \end{aligned} \tag{79}$$

#### 7.2.2.1 *Analytical Solutions*

In the specific case of dealing with maximizing the average performance, or equivalently, using the Markov Bound approximation with $\epsilon = 1$, the analytical solutions mentioned in [10] require that $\mathbf{S}_R = \mathbf{R}_R$.

In the first of the cases listed, *Proposition 2* in [9] has been adapted[4] as mentioned in [10], while in the second case the result provided in [10] is included. These analytical solutions can be found in table 2.

As in the GPP case a general structure is followed by the different solutions for $\mathbf{P}_{opt}$. In this case, all solutions included in table 2 refer to the matrices that appear in (80).

$$\mathbf{P}_{opt} = \mathbf{U}_{T'}\mathbf{D}_{P_{opt}}\mathbf{U}_{Q'} \tag{80}$$

where $\mathbf{R}_T^T$ and $\mathbf{S}_Q^T$ can be expressed as $\mathbf{U}_{T'}\mathbf{\Lambda_{T'}}\mathbf{U}_{T'}^H$ and $\mathbf{U}_{Q'}\mathbf{\Lambda_{Q'}}\mathbf{U}_{Q'}^H$ respectively. The order in which these eigenvalues are ordered is discussed in each case specifically.

1. When $\mathcal{I}_T = \mathbf{I}$: The optimal training sequence can be expressed as in (80) with the eigenvalues of $\mathbf{R}_T^T$ and $\mathbf{S}_Q^T$ placed in *descending* and *ascending* order respectively. The values of $(\mathbf{D}_{P_{opt}})_{i,i}$ for $i = 1, \ldots, m^*$ are expressed in (81) and are 0 for $i > m^*$.

$$\mathbf{D}_{P_{opt}}(i,i) = \frac{\mathcal{P} + \sum_{j=1}^{m^*} \frac{(\mathbf{\Lambda}_{Q'})_{(j,j)}}{(\mathbf{\Lambda}_{T'})_{(j,j)}}}{\sqrt{(\mathbf{\Lambda}_{Q'})_{(i,i)}} \sum_{j=1}^{m^*} \sqrt{(\mathbf{\Lambda}_{Q'})_{(j,j)}}} - \frac{1}{(\mathbf{\Lambda}_{T'})_{(i,i)}} \tag{81}$$

   Where $m^*$ is expressed as:

$$\begin{aligned} m^* = &\max\{m \in 1, \ldots, n_T : \frac{\sqrt{(\mathbf{\Lambda}_{Q'})_{(m,m)}}}{(\mathbf{\Lambda}_{T'})_{(m,m)}} \sum_{i=1}^{m} \sqrt{(\mathbf{\Lambda_{Q'}})_{(\mathbf{i},\mathbf{i})}} \\ &- \sum_{i=1}^{m} \frac{(\mathbf{\Lambda}_{Q'})_{(i,i)}}{(\mathbf{\Lambda}_{T'})_{(i,i)}} < \mathcal{P}\} \end{aligned} \tag{82}$$

2. When $\mathbf{R}_T^{-1} = \mathcal{I}_T$ can be written as in (80) with a specific eigenvalue ordering determined in [10]. When this optimal ordering is given, the optimal $(\mathbf{D}_{P_{opt}})_{i,i}$ can be expressed as (83).

$$(\mathbf{D}_{P_{opt}}(i,i))^2 = \frac{\mathcal{P} + \sum_{j=1}^{m_*} \frac{(\Lambda_Q)_{j,j}}{(\Lambda'_T)_{j,j}}}{\sum_{j=1}^{m_*} \sqrt{\frac{(\Lambda_Q)_{j,j}}{(\Lambda'_T)_{j,j}}}} \sqrt{\frac{(\Lambda_Q)_{i,i}}{(\Lambda'_T)_{i,i}}} - \frac{(\Lambda_Q)_{i,i}}{(\Lambda'_T)_{i,i}} \tag{83}$$

   and the expression of $m^*$ can be found in Appendix 5 in [10]

Table 2.: Available analytical solutions for Averaged Performance MPP.

4 Proof can be found in the appendix A.

# 8

## SEMIDEFINITE RELAXATION

In order to be able to write the problems as SDPs most of the steps described in [11] have been followed. These steps will be included in appendix B for completeness and a higher level of detail will be provided as well.

The GPP and the MPP when using the Markov Bound approximation can be expressed as (84) and (85) respectively.

$$\begin{aligned} \underset{\mathbf{P}\in\mathbb{C}^{n_T\times B}}{\text{minimize}} \quad & \operatorname{tr}(\mathbf{P}\mathbf{P}^H) \\ \text{s.t.} \quad & \operatorname{tr}(\mathbf{M}) \leq \frac{\epsilon}{\gamma} \\ & \begin{bmatrix} \mathbf{M} & \boldsymbol{\mathcal{I}}_{adm}{}^{\frac{1}{2}} \\ \boldsymbol{\mathcal{I}}_{adm}{}^{\frac{1}{2}} & \mathbf{C}_{MMSE}^{-1} \end{bmatrix} \succeq 0 \\ & \mathbf{M} = \mathbf{M}^H \end{aligned} \tag{84}$$

$$\begin{aligned} \underset{\mathbf{P}\in\mathbb{C}^{n_T\times B}}{\text{maximize}} \quad & \gamma \\ \text{s.t.} \quad & \operatorname{tr}(\mathbf{M}) \leq \frac{\epsilon}{\gamma} \\ & \begin{bmatrix} \mathbf{M} & \boldsymbol{\mathcal{I}}_{adm}{}^{\frac{1}{2}} \\ \boldsymbol{\mathcal{I}}_{adm}{}^{\frac{1}{2}} & \mathbf{C}_{MMSE}^{-1} \end{bmatrix} \succeq 0 \\ & \mathbf{M} = \mathbf{M}^H \\ & \operatorname{tr}(\mathbf{P}\mathbf{P}^H) \leq \mathcal{P} \end{aligned} \tag{85}$$

Where $\mathbf{M}$ is an auxiliary matrix and $C_{MMSE}^{-1} = \mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}}$, which when recalling that $\mathbf{S} = \mathbf{S}_Q^T \otimes \mathbf{S}_R$, can be expressed as (86) using (87)-(89).

$$\mathbf{C}_{MMSE}^{-1} = \mathbf{R}^{-1} + \mathbf{P}^*\mathbf{S}_{\mathbf{Q}}^{-\mathbf{T}}\mathbf{P}^T \otimes \mathbf{S}_R^{-1} \tag{86}$$

$$\mathbf{AB} \otimes \mathbf{CD} = (\mathbf{A}\otimes\mathbf{C})(\mathbf{B}\otimes\mathbf{D}) \tag{87}$$

$$(\mathbf{A}\otimes\mathbf{B})^{-1} = \mathbf{A}^{-1}\otimes\mathbf{B}^{-1} \tag{88}$$

$$(\mathbf{A}\otimes\mathbf{B})^{H} = \mathbf{A}^{H}\otimes\mathbf{B}^{H} \tag{89}$$

In order to be able to write the whole problem as an SDP, it is necessary that the matrix inequality restrictions can be written as a

LMI with the optimization variables. Therefore, a change of variable is required. As proposed in [11] $\mathbf{P}^{*}\mathbf{S}_{\mathbf{Q}}^{-\mathbf{T/2}}$ is set to be $\mathbf{P}_{\mathbf{Q}}$ so that the restriction in performance can be expressed as (90) and the expression of training sequence power is rewritten as (91).

$$\begin{bmatrix} \mathbf{M} & \mathcal{I}_{adm}{}^{\frac{1}{2}} \\ \mathcal{I}_{adm}{}^{\frac{1}{2}} & \mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H} \otimes \mathbf{S}_{R}^{-1} \end{bmatrix} \succeq 0 \qquad \mathbf{M} = \mathbf{M}^{H} \tag{90}$$

$$\mathrm{tr}(\mathbf{P}_{\mathbf{Q}}\mathbf{S}_{\mathbf{Q}}{}^{H}\mathbf{P}_{Q}^{H}) \leq \mathcal{P} \tag{91}$$

(90) is not convex with $\mathbf{P}_{Q}$ so another change of variable is required. $\mathbf{Z} = \mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H}$, as mentioned in [11], is a very convenient change of variable since it makes it possible to express (90) as LMI with $\mathbf{Z}$, which will be a new optimization variable. At the same time, this change has to be applied to (91). This change of variable is expanded with detail in [11] and is included in (92) for completeness.

$$\begin{aligned} \mathrm{tr}(\mathbf{P}_{\mathbf{Q}}\mathbf{S}_{\mathbf{Q}}{}^{T}\mathbf{P}_{\mathbf{Q}}{}^{H}) &= \mathrm{vec}^{T}(\mathbf{S}_{\mathbf{Q}})\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}}) \\ &= \mathrm{vec}^{H}(\mathbf{S}_{\mathbf{Q}}{}^{*})\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}}) \\ &= |\mathrm{vec}^{H}(\mathbf{S}_{\mathbf{Q}}{}^{*})\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}})| \\ &\leq ||\mathrm{vec}(\mathbf{S}_{\mathbf{Q}}{}^{*})||\,||\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}})|| \\ &= ||\mathrm{vec}(\mathbf{S}_{\mathbf{Q}}{}^{*})||\,||\mathrm{vec}(\mathbf{Z})|| \end{aligned} \tag{92}$$

where the inequality follows from the Cauchy inequality, and is satisfied with equality when $\mathrm{vec}(\mathbf{S}_{\mathbf{Q}}{}^{*})$ and $\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}})$ are linearly dependent, i.e. parallel. Note that the last equality follows from the fact in equation (93).

$$\begin{aligned} ||\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}})|| = ||\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}}||_{F} = \sqrt{\mathrm{trace}((\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}})^{H}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}}))} = \\ = \sqrt{\mathrm{trace}(\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}})} = \sqrt{\mathrm{trace}(\mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H}\mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H})} = \\ = \sqrt{\mathrm{trace}((\mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H})^{H}(\mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H}))} = ||\mathrm{vec}(\mathbf{P}_{\mathbf{Q}}\mathbf{P}_{\mathbf{Q}}{}^{H})|| = ||\mathrm{vec}(\mathbf{Z})|| \end{aligned} \tag{93}$$

The restriction on the power, as mentioned in [11] can be expressed as an LMI with $\mathbf{Z}$ using again the Schur-complement[1], meaning that the problem is finally expressed as an SDP. Note that, if the restriction were expressed as a second order convex cone constraint, i.e. $||\mathrm{vec}(Z)|| \leq \frac{\mathcal{P}}{||\mathrm{vec}(\mathbf{S}_{\mathbf{Q}}{}^{*})||}$ the problem would have been successfully convexified as well and the result would be identical.

Rewriting the restriction in power in terms of the training sequence $\mathbf{P}$, it is possible to express it as in (94).

$$||\mathrm{vec}(\mathbf{P}^{*}\mathbf{S}_{Q}^{-*}\mathbf{P}^{H})|| \leq \frac{\mathcal{P}}{||\mathrm{vec}(\mathbf{S}_{Q}^{*})||} \tag{94}$$

1 Proof found in Appendix

Which implies a feasible set contained in the feasible set described by $\text{tr}(\mathbf{P}\mathbf{P}^H)$ due to the additional restriction that the involvement of matrix $\mathbf{S}_Q$ entails. For experimental results and additional conclusions derived from them regarding this approximation please refer to the first section in chapter 10.

These changes of variables can be applied to the confidence ellipsoid cases as well as to the Markov Bound ones, resulting in the expressions (95) - (98).
GPP Confidence Ellipsoid:

$$\begin{aligned} \underset{\mathbf{Z}\in\mathbb{C}^{n_T\times B},\beta\in\mathbb{R}^+}{\text{minimize}} \quad & \beta \\ \text{s.t.} \quad & \mathbf{R}^{-1}+\mathbf{Z}\otimes{\mathbf{S}_\mathbf{R}}^{-1}\succeq\frac{\gamma\chi_\alpha^2(2n_rn_t)}{2}\boldsymbol{\mathcal{I}_{adm}} \\ & \begin{bmatrix}\frac{\beta}{||\text{vec}(\mathbf{S}_Q^*)||} & \text{vec}(\mathbf{Z})^H \\ \text{vec}(\mathbf{Z}) & \frac{\beta}{||\text{vec}(\mathbf{S}_Q^*)||}\mathbf{I}\end{bmatrix}\succeq 0 \\ & \mathbf{Z}=\mathbf{Z}^H\succeq 0 \end{aligned} \tag{95}$$

GPP Markov Bound:

$$\begin{aligned} \underset{\mathbf{Z},\mathbf{M}\in\mathbb{C}^{n_T\times B},\beta\in\mathbb{R}^+}{\text{minimize}} \quad & \beta \\ \text{s.t.} \quad & \text{tr}(\mathbf{M})\leq\frac{\epsilon}{\gamma} \\ & \begin{bmatrix}\mathbf{M} & \boldsymbol{\mathcal{I}_{adm}}^{\frac{1}{2}} \\ \boldsymbol{\mathcal{I}_{adm}}^{\frac{1}{2}} & \mathbf{Z}\otimes\mathbf{S}_R^{-1}\end{bmatrix}\succeq 0 \\ & \begin{bmatrix}\frac{\beta}{||\text{vec}(\mathbf{S}_Q^*)||} & \text{vec}(\mathbf{Z})^H \\ \text{vec}(\mathbf{Z}) & \frac{\beta}{||\text{vec}(\mathbf{S}_Q^*)||}\mathbf{I}\end{bmatrix}\succeq 0 \\ & \mathbf{Z}=\mathbf{Z}^H\succeq 0 \end{aligned} \tag{96}$$

MPP Confidence Ellipsoid:

$$\begin{aligned} \underset{\mathbf{Z}\in\mathbb{C}^{n_T\times B},\gamma\in\mathbb{R}^+}{\text{maximize}} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{R}^{-1}+\mathbf{Z}\otimes{\mathbf{S}_\mathbf{R}}^{-1}\succeq\frac{\gamma\chi_\alpha^2(2n_rn_t)}{2}\boldsymbol{\mathcal{I}_{adm}} \\ & \begin{bmatrix}\frac{\mathcal{P}}{||\text{vec}(\mathbf{S}_Q^*)||} & \text{vec}(\mathbf{Z})^H \\ \text{vec}(\mathbf{Z}) & \frac{\mathcal{P}}{||\text{vec}(\mathbf{S}_Q^*)||}\mathbf{I}\end{bmatrix}\succeq 0 \\ & \mathbf{Z}=\mathbf{Z}^H\succeq 0 \end{aligned} \tag{97}$$

MPP Markov Bound:

$$\begin{aligned} \underset{\mathbf{Z},\mathbf{M}\in\mathbb{C}^{n_T\times B},\beta\in\mathbb{R}^+}{\text{maximize}} \quad & \gamma \\ \text{s.t.} \quad & \operatorname{tr}(\mathbf{M}) \leq \frac{\epsilon}{\gamma} \\ & \begin{bmatrix} \mathbf{M} & \boldsymbol{\mathcal{I}_{adm}}^{\frac{1}{2}} \\ \boldsymbol{\mathcal{I}_{adm}}^{\frac{1}{2}} & \mathbf{Z}\otimes\mathbf{S}_R^{-1} \end{bmatrix} \succeq 0 \\ & \begin{bmatrix} \frac{\mathcal{P}}{||\operatorname{vec}(\mathbf{S}_Q^*)||} & \operatorname{vec}(\mathbf{Z})^H \\ \operatorname{vec}(\mathbf{Z}) & \frac{\mathcal{P}}{||\operatorname{vec}(\mathbf{S}_Q^*)||}\mathbf{I} \end{bmatrix} \succeq 0 \\ & \mathbf{Z} = \mathbf{Z}^H \succeq 0 \end{aligned} \tag{98}$$

## 8.1 OBTAINING THE TRAINING SEQUENCE

Once these problems are solved and a $\mathbf{Z}_{opt}$ is found a new problem arises, since a way to obtain the training sequence $\mathbf{P}$ is required. The same exact procedure as described in [11] is used to recover $\mathbf{P}$. A synthesized description is included below.

1. Assume $\mathbf{Z}_{opt}$ has the following eigenvalue decomposition: $\mathbf{Z}_{opt} = \mathbf{U}_{Z_{opt}}\mathbf{D}_{Z_{opt}}\mathbf{U}_{Z_{opt}}^H$, where $\mathbf{U}_{Z_{opt}} \in \mathbb{C}^{n_T\times n_T}$ contains the eigenvectors and $\mathbf{D}_{Z_{opt}} \in \mathbb{R}^{n_T\times n_T}$ contains the eigenvalues in *decreasing order*.

2. Assume that $\mathbf{U}_Q\mathbf{D}_Q\mathbf{U}_Q^H$ is the corresponding eigenvalue decomposition of ${\mathbf{S_Q}}^T$, where no assumption of the order of the eigenvalues in $\mathbf{D}_Q$ is made.

3. Recall that $\mathbf{Z} = \mathbf{P_Q}{\mathbf{P_Q}}^H = \mathbf{P}^*{\mathbf{S_Q}}^{-T}\mathbf{P}^H$.

4. An intuitive but *adhoc* way of selecting a training sequence $\mathbf{P}$, with generalized eigenvalue decomposition $\mathbf{P}^* = \mathbf{U}_{P^*}\mathbf{D}_{P^*}\mathbf{V}_{P^*}$, that can generate $\mathbf{Z}_{opt}$, is to pick $\mathbf{U}_{P^*} = \mathbf{U}_{Z_{opt}}$, $\mathbf{V}_{P^*} = \mathbf{U}_Q$, and finally $(\mathbf{D}_{\mathbf{P}^*}(i,i))^2 = \mathbf{D}_{Z_{opt}}(i,i)\mathbf{D}_Q(i,i)$, $i = 1,\ldots,\min(n_T, B)$.

5. The eigenvalues in $\mathbf{D}_Q$ should be arranged to minimize the training sequence power $\operatorname{tr}(\mathbf{P}^*\mathbf{P}^T) = \sum_{i=1}^{n_T}\mathbf{D}_P(i,i)^2$. Which according to [11] should be placed in *increasing order*.

As a summary the training sequence found using this methodology will be built as in (99).

$$\mathbf{P} = \mathbf{U}_{Z_{opt}}^*\mathbf{D}_P\mathbf{U}_Q^T \tag{99}$$

Where $\mathbf{D}_P = \sqrt{\mathbf{D}_Z(i,i)\mathbf{D}_Q(i,i)}$ with $i = 1,\ldots,\min(n_T, B)$.

## 8.2 ACHIEVING THE ANALYTICAL SOLUTION

Considering the structure forced upon $\mathbf{P}$ when obtaining it through the matrix $\mathbf{Z}_{opt}$, a theoretical analysis of the possibility of meeting the analytical solutions is done for some of the cases.

### 8.2.1 *GPP: Confidence Ellipsoids*

Recall the structure of the analytical solution in equation (74) and the analytical solutions provided in figure 1. In all cases the analytical solution for $\mathbf{P}_{opt}$ is built with the eigenvectors of $\mathbf{S}_Q$ to the right, as well as when the solution is obtained from $\mathbf{Z}_{opt}$. Additionally, matrix $\mathbf{B}$ never requires information that is not available in the SDP problem formulation. This implies that if the optimal training sequence is not disregarded by the reduced feasible set, it is actually possible that the analytical solution is met. Specifically, if $\mathbf{Z}$ were equivalent to $\mathbf{B}$'s except for a unit-norm scaling factor in each column of their modal matrices, the solution obtained with the SDP would be the analytical one. The experiments and results carried out to test if the analytical solution is met or not are included in chapter 10.

### 8.2.2 *Averaged performance MPP*

The same type of analysis can be carried out in the MPP case. In this case recall the final formulation of the MPP, (98), using Markov Bound with $\epsilon = 1$. Recall as well that the structure of the optimal training sequence follows the one in equation (80) with the solutions listed in figure 2.

It is important to note that in the SDP solution, information about $\mathbf{S}_Q$'s structure is not included within $\mathbf{Z}$, but when building the training sequence $\mathbf{P}$. Therefore, considering that the diagonal entries of $(\mathbf{D}_P)^2_{(i,i)}$ when obtaining it from $\mathbf{Z}$, are built as the product of the eigenvalues of $\mathbf{S}_Q$ and $\mathbf{Z}$, which does not contain structural information of $\mathbf{S}_Q$; a solution as the ones listed in figure 2 can not be obtained from the SDP.

# 9

# METRICS

As mentioned at the beginning of the chapter 7, metrics of the shape $J(\tilde{\mathbf{H}}, \mathbf{H}) = \text{vec}^H(\tilde{\mathbf{H}})\boldsymbol{\mathcal{I}}_{adm}\text{vec}(\tilde{\mathbf{H}})$ are considered. Since it is possible that these type of metrics have their origins in an approximation, an analysis of the impact of this approximation is carried out in chapter 10, and the expressions required for the tests are provided in this chapter. In the following sections, the metrics proposed in [10] are reviewed. Additionally, for the Zero Forcing pre-coder case a new expression is provided to evaluate the real performance of the metric.

## 9.1 MSE: MEAN SQUARE ERROR

The Mean Square error can be, of course, represented as a quadratic form using $\boldsymbol{\mathcal{I}}_{adm} = \mathbf{I}$. This metric is not system specific, and its aim is to minimize the channel estimation mean square error. From the point of view of obtaining the most accurate description of the channel, this is the best option, but as stated in chapter 6, the best channel estimation in terms of Mean Square error might not be the channel estimate that will provide the best throughput of data when this channel estimate is used in a specific application. This metric has been implemented as well but mainly for comparative purposes.

## 9.2 SYSTEM SPECIFICS: USING A ZERO FORCING PRECODER

In this specific case a Zero Forcing pre-coder is considered. Following the notation in [10] the data transmission can be described as (100)

$$\mathbf{y}(t) = \mathbf{H}\boldsymbol{\Psi}\mathbf{x}(t) + \mathbf{n}(t) \tag{100}$$

where $\mathbf{y}(t) \in \mathbb{C}^{n_R \times 1}$ and $\mathbf{x}(t) \in \mathbb{C}^{n_R \times 1}$ are, respectively, the received signal and transmited signal during the data transmission phase and $\boldsymbol{\Psi} = \mathbf{H}^{\dagger} \in \mathbb{C}^{n_T \times n_R}$ is the ideal Zero Forcing pre-coder. This element attempts to pre-process the signal $\mathbf{x}(t)$ before the transmission so that no interference is perceived from antenna to antenna in reception. Consider a situation in which different symbols are transmitted from each of the $n_T$ transmitting antennas. The channel $\mathbf{H}$ will mix the signals originating from the different transmitted antennas and therefore they will, if nothing is done, interfere each other in reception.

By using the channel's pseudoinverse, which is actually the inverse when $\mathbf{H}$ is square and full-rank, the channel appears to be the identity from the receiver's point of view, and therefore, no interference is perceived. Additionally, since the channel inversion is done in transmission and not reception, no noise enhancement effect is produced.

### 9.2.1 *Optimization Metric*

The usage of this technique clearly requires a channel estimate if the channel is unknown. A metric that is representative of the function the channel estimate will fulfill needs to be selected to obtain the optimal training sequence for this specific case. Keeping in mind that the metric should be reasonably possible to be approximated by a quadratic form with $\text{vec}(\tilde{\mathbf{H}})$. For that purpose a very convenient structure to follow is the one of the Mean Square Error, in other words, computing, and trying to minimize the mean square error of a magnitude. Therefore, the same strategy as in [10] is followed, leading to the metric in (101).

$$\mathrm{J}_{ZF}(\hat{\mathbf{H}}, \mathbf{H}) = \mathbb{E}\{[\mathbf{y}(t;\hat{\mathbf{H}}) - \mathbf{y}(t;\mathbf{H})]^H[\mathbf{y}(t;\hat{\mathbf{H}}) - \mathbf{y}(t;\mathbf{H})]\} \tag{101}$$

where $\mathbf{y}(t;\hat{\mathbf{H}})$ is the received signal at instant $t$ when the channel estimate is being used in the ZF pre-coder, and $\mathbf{y}(t;\mathbf{H})$ is the received signal at instant $t$ when perfect knowledge of the channel is provided. The result obtained after some approximations is expressed in (102).

$$\text{vec}^H(\tilde{\mathbf{H}})((\lambda_x \mathbf{H}^\dagger(\mathbf{H}^\dagger)^H)^T \otimes \mathbf{I})\text{vec}(\tilde{\mathbf{H}}) \tag{102}$$

If the problem were written using this metric, the optimal training sequence would depend on the true channel, therefore, the model in (103) is assumed.

$$\mathbf{H}_i = \mathbf{H}_{i-1} + \mu\mathbf{E}_i \tag{103}$$

where $\mathbf{E}_i$ has the same Kronecker structure as $\mathbf{H}$ and is independent of the channel realizations. This model of change from block to block can be interpreted as the movement of some of the elements in the system, where $\mu$ indicates how fast this element moves and therefore, regulates the contribution of the random part in the next block. Under this assumption, the realizations of the channel are not uncorrelated to the previous block. Meaning that it is possible to use the previous channel estimate in the metric. Resulting in (104).

$$\text{vec}^H(\tilde{\mathbf{H}}_i)((\lambda_x \hat{\mathbf{H}}_{i-1}^\dagger(\hat{\mathbf{H}}_{i-1}^\dagger)^H)^T \otimes \mathbf{I})\text{vec}(\tilde{\mathbf{H}}_i) \tag{104}$$

The inconvenience of this approach is that guaranteeing a certain performance based upon this metric will not imply that it will be the perceived performance. That is why it is necessary to evaluate the

actual obtained performance when using a non-approximated metric. For that purpose an expression of the real metric has been derived.

### 9.2.2 *Real Metric*

When using a metric with evaluative purposes it is not necessary to consider that certain information, such as the current channel realization or estimate, is not available until after transmission. With this in mind, an alternative formulation for (101), with no approximations required, is provided in (105).

$$J(\hat{\mathbf{H}}_i, \tilde{\mathbf{H}}_i) = \text{vec}^H(\tilde{\mathbf{H}}_i)(\hat{\mathbf{H}}_i^{\dagger *}\lambda_x \mathbf{I}\hat{\mathbf{H}}_i^{\dagger T} \otimes \mathbf{I})\text{vec}(\tilde{\mathbf{H}}_i) \tag{105}$$

A proof of this expression is provided in appendix C.

## 9.3 SYSTEM SPECIFICS: USING AN MMSE EQUALIZER

The Zero Forcing Pre-coder pre-processes the signal so that the channel appears to be white in reception, but does nothing when it comes to noise or interferences that do not have their origin in the same system. On the other hand, the MMSE Equalizer uses the channel estimate to combat the effect of the channel and outer-system effects as well, i.e., the MMSE Equalizer $\mathbf{F}_{eq}(q, \mathbf{H})$, tries to provide an estimation of the transmitted sequence by minimizing the mean square error between the transmitted sequence $\mathbf{x}(t)$ and its estimate $\hat{\mathbf{x}}(t)$, by linearly filtering the received sequence $\mathbf{y}(t)$.

$$\hat{\mathbf{x}}(t) = \mathbf{F}_{eq}(q, \mathbf{H})\mathbf{y}(t) \tag{106}$$

Where $q$ indicates the unit time-shift. As in [10] the non-casual Wiener filter will be used as an approximation of the MMSE Equalizer. Therefore, as provided in [10] the filter follows the expression in (107).

$$\mathbf{F}(e^{j\omega}; \mathbf{H}) = \mathbf{H}^H(\mathbf{H}\mathbf{H}^H + \Phi_n(\omega)/\lambda_x)^{-1} \tag{107}$$

### 9.3.1 *Real metric*

Just as with the Zero Forcing case, for convenience and simplicity the MSE of the difference will be considered: $\hat{\mathbf{x}}(t; \hat{\mathbf{H}}) - \mathbf{x}(t; \mathbf{H}) = (\mathbf{F}(q; \hat{\mathbf{H}}) - \mathbf{F}(q; \mathbf{H}))\mathbf{y}(t)$, and so the metric can be expressed as in (108).

$$J_{EQ}(\tilde{\mathbf{H}}, \mathbf{H}) = \mathbb{E}\{[(\mathbf{F}(q; \hat{\mathbf{H}}) - \mathbf{F}(q; \mathbf{H}))\mathbf{y}(t)]^H[(\mathbf{F}(q; \hat{\mathbf{H}}) - \mathbf{F}(q; \mathbf{H}))\mathbf{y}(t)]\} \tag{108}$$

which can at the same time be expressed as:

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} \text{tr}\left((\mathbf{F}(e^{j\omega}; \hat{\mathbf{H}}) - \mathbf{F}(e^{j\omega}; \mathbf{H}))\mathbf{\Phi}_y(\omega)(\mathbf{F}(e^{j\omega}; \hat{\mathbf{H}}) - \mathbf{F}(e^{j\omega}; \mathbf{H}))^H\right) d\omega \tag{109}$$

### 9.3.2 *Optimization metric*

The expression provided does not follow a quadratic form with the estimation error, and leads to a non-convex problem when fed in the GPP or MPP formulations. Therefore several approximations are made in [10] in order to obtain a metric that is both convex and quadratic with vec($\tilde{\mathbf{H}}$). This proofs can be found in Appendix 1 in [10]. The results of these approximations are provided in table 3 for convenience.

1. High SNR and rank($\mathbf{H}$) $= n_R \leq n_T$: $\boldsymbol{\mathcal{I}_{adm}} = \lambda_x \mathbf{I} \otimes (\mathbf{H}\mathbf{H}^H)^{-1}$
2. High SNR and rank($\mathbf{H}$) $= n_T < n_R$: $\boldsymbol{\mathcal{I}_{adm}} = \lambda_x \mathbf{I} \otimes \left( \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathbf{\Phi}_n^{-1/2} [\mathbf{\Phi}_n^{-1/2} \mathbf{H}\mathbf{H}^H \mathbf{\Phi}_n^{-1/2}]^{\dagger} \mathbf{\Phi}_n^{-1/2} \, d\omega \right)$
3. Low SNR: $\boldsymbol{\mathcal{I}_{adm}} = \mathbf{I} \otimes \left( \frac{\lambda_x^2}{2\pi} \int_{-\pi}^{\pi} \mathbf{\Phi}_n^{-1} \, d\omega \right)$

Table 3.: Approximated metrics for a system with a MMSE Equalizer.

Note that, the Markov 1 assumption is not yet applied to the approximations. It is sufficient to replace $\mathbf{H}$ by $\hat{\mathbf{H}}_{\mathbf{i-1}}$ for the metric to be implementable.

# 10

# EXPERIMENTS

In this chapter the experiments that have been carried out will be described and the conclusions drawn from them will be explained. The experiments are designed to answer the question formulated in each specific section. In the experiments, the covariance matrices that characterize both the channel and the noise, $\mathbf{R_R}$, $\mathbf{R_T}$, $\mathbf{S_R}$ and $\mathbf{S_Q}$, follow an exponential model as in [10]; in which the entries of the covariance matrices are built according to (110)

$$(\mathbf{R})_{i,j} = r^{j-i},\ j \geq i, \tag{110}$$

where $r$ is the normalized correlation coefficient with $\rho = |r| < 1$. In all experiments, a random parameter $r$ is generated for each covariance matrix considering a fix and provided $\rho$. After that, the covariance matrices are synthesized and fed as input parameters to the optimization and estimation routines. In order to obtain the performances and outage probabilities plotted in the following sections, the steps in table 4 are followed.

## 10.1 FULFILLING THE REQUIREMENTS

Once the different formulations of the problems have been implemented, it is important to see if the solutions provided by the optimization routine fulfill the restrictions tightly. Therefore, an experiment using the MMSE metric, due to its simplicity, has been carried out for each of the different SDP formulations. The results for this experiments are included in figures 1, 2, 3 and 4. Where 1 and 3 correspond to the SDP formulation of the GPP using the Markov Bound and the Confidence Ellipsoids approximations, respectively, and 2 and 4 correspond to the SDP formulation of the MPP using the Markov Bound and the Confidence Ellipsoids approximation.

In figure 1 it is shown that in the GPP case, when the value $\epsilon$ is set to be 1, the average metric corresponds to the metric restriction. Consequently, the average metric decreases as the value of $\epsilon$ decreases. On the other hand, as seen in figure 2, even when $\epsilon = 1$, the restriction of power in the MPP is not active since the solution never reaches the restriction $\mathcal{P}$. An analogous analysis can be carried out with figures 3 and 4.

1. Generation of covariance matrices or modification if required.
2. In case it is required: generation of initial channel.
3. Generation of metric matrix.
4. Covariance and metric matrices are fed to the optimization routine and a training sequence is obtained.
5. The training sequence is transmitted through several realizations of the channel *number of averages* times, following a suitable statistical description.
6. After each transmission, the estimation error is computed and used to estimate the obtained performance.
7. The performance in each transmission is compared to the performance restriction. If the requirement is not met, the estimated outage probability is increased.
8. In case it is required: the last channel estimate is used to generate the next metric matrix.

Table 4.: General experiment guidelines. Specific guidelines are provided in each section mentioning the particulars of each case. Additionally cases that required the mentioned special treatments will be explained in detail in the specific sections.

The restriction is not active due to the use of the Cauchy inequality applied in (92). The non fulfillment of the restriction implies that the Chauchy inequality is not fulfilled with equality, meaning that the vectors $\text{vec}(\mathbf{S}_Q^*)$ and $\text{vec}(\mathbf{Z})$ are not parallel.

As a matter of fact, given the current formulation of the problems, $\text{vec}(\mathbf{Z})$ can not be parallel to $\text{vec}(\mathbf{S}_Q^*)$ since no information about the shape of this last matrix is provided in the problem. It is important to note, that even though this strategy does not allow to tightly fulfill the restriction for the MPP case, it does for the GPP case. With the GPP SDP formulation the parameter to minimize is $\beta$ which is equivalent to the restriction $\mathcal{P}$ in the MPP. This does not imply that the power of $\mathbf{Z}$ is $\frac{\beta}{||\text{vec}(\mathbf{S}_Q)||}$, but acts as a bound, that is not reached, of the power of $\mathbf{Z}$. Minimizing this bound implies that an effort is done to minimize the actual power of $\mathbf{Z}$, as long as it is not orthogonal to $\mathbf{S}_Q$.

The use of the Cauchy inequality is potentially detrimental to the GPP case as well, since the reduction of the feasible set might disregard the otherwise optimal solution. As a matter of fact, recalling the formulation in (92) where $\text{tr}(\mathbf{P_Q}\mathbf{S_Q}^T\mathbf{P_Q}^H)$ can be expressed as

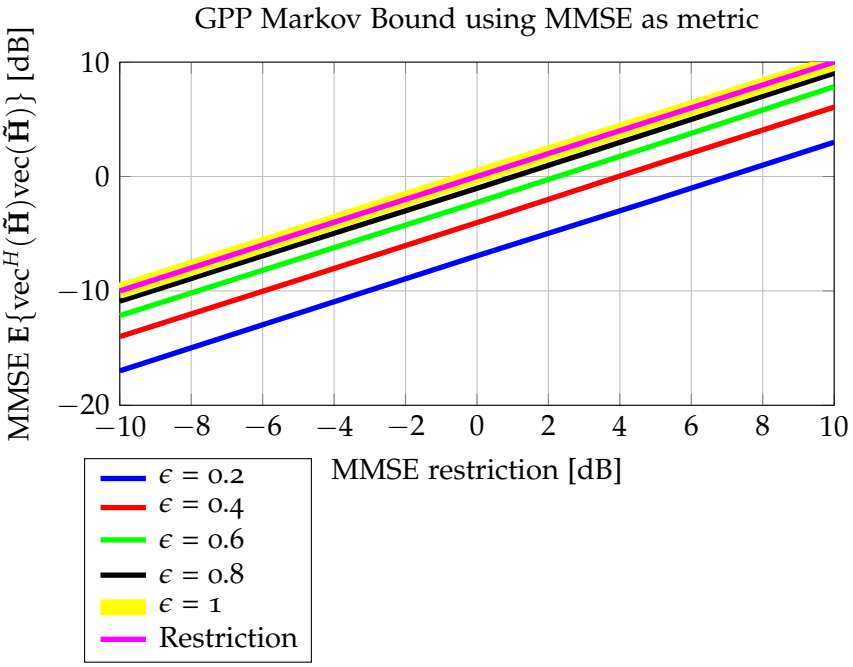


Figure 1.: Restriction fulfillment when using the Markov Bound approximation for GPP using MMSE as metric, with parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$ and 2000 averages.

$\text{vec}^H(\mathbf{S}_\mathbf{Q}^*)\text{vec}(\mathbf{Z})$, it is possible to rewrite the Cauchy inequality as in (111)

$$||\text{vec}(\mathbf{S_Q}^*)||||\text{vec}(\mathbf{Z})||\cos(\phi) \leq ||\text{vec}(\mathbf{S_Q}^*)||||\text{vec}(\mathbf{Z})||, \qquad (111)$$

where $\phi$ denotes the angle between the two vectors. Considering that the restriction is forced upon the second term in equation (111), the parameter $\phi$ has no relevance during optimization, which may lead to not selecting a $\mathbf{Z}$ that might have high power but a very small projection on $\mathbf{S_Q}^*$. This does not necessarily imply that the true optimal solution is not found but implies a risk of it happening.

With this result at hand it is possible to scale the result in order to obtain a training sequence that will already outperform the one obtained by solving the SDP formulation of the MPP, since, given the same shape, an increase in power will positively affect the sequence's performance. On the other hand, if the power restriction were increased, the power allocation in the training sequence is likely to change obtaining a better performance than just by scaling the result. In order to fairly compare the SDP-MPP solution with the SDP-GPP solution from now on, power equalization is applied to all MPP formulations. This implies that the power restriction is increased/decreased until the obtained training sequence meets the desired power.

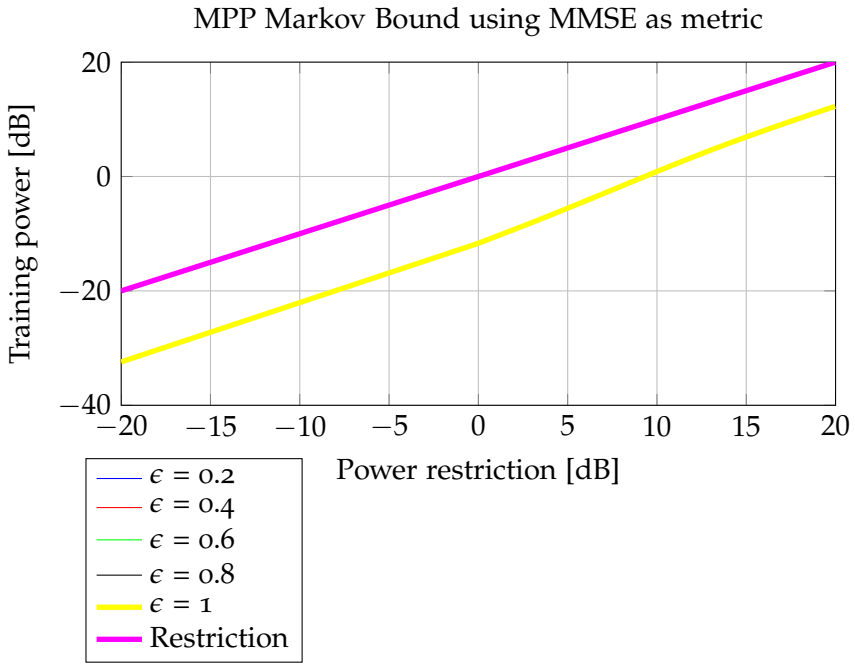


Figure 2.: Restriction fulfillment when using the Markov Bound approximation for MPP using MMSE as metric, with parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$ and 2000 averages.

## 10.2 ACHIEVING THE ANALYTICAL SOLUTION

As mentioned in chapter 8 it could be possible that the SDP formulation matches the analytical solution in some cases. Recall from chapter 8 that an analytical solution only exists in the GPP case when using the Confidence Ellipsoids approximation and under specific conditions. The SDP-MPP case has not been tested since it is not possible to obtain the analytical solution solving the SDP since some of the information required to obtain it is not present in the problem formulation.

### 10.2.1 *GPP Confidence Ellipsoids*

For testing purposes the first case of the analytical solutions listed in table 1 has been simulated. The analytical solution has been calculated and compared to the solution obtained by solving (95).

When examining the matrix $\mathbf{Z}$ obtained through solving the SDP formulation and the matrix $\mathbf{B}$ in figure 1, the eigenvalues coincide and the eigenvectors do as well except for a complex scaling factor of magnitude one in each column, meaning that it is actually equivalent to $\mathbf{B}$. Additionally, $\mathbf{P}$ is obtained from $\mathbf{Z}$ in the same way as $\mathbf{P}_{opt}$ is formulated in (74), i.e. its eigenvalues and $\mathbf{S}_Q$'s are ordered in the

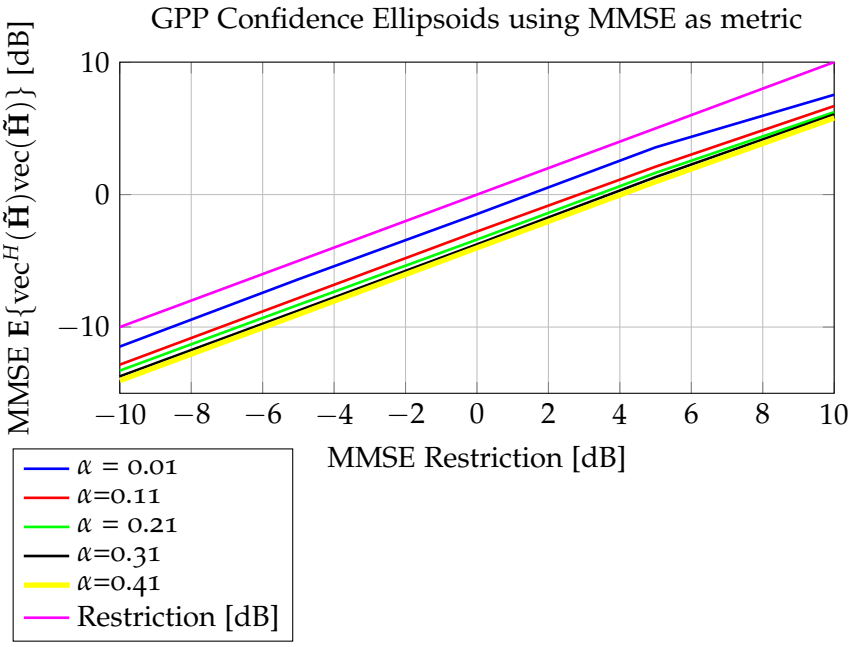


Figure 3.: Restriction fulfillment when using the Confidence Ellipsoid approximation for GPP using MMSE as metric, with parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$ and 2000 averages.

same way in order the build the diagonal matrix in $\mathbf{P}$'s SVD[1], and the eigenvectors of $\mathbf{S}_Q$ are used in the same exact way, meaning that the solving SDP formulation using Confidence Ellipsoids actually provides the analytical solution, implying that in this specific case the reduced feasible set does not disregard the optimal point.

## 10.3 REAL AND APPROXIMATED METRICS

As mentioned in Chapter 9, the optimized metric and the real metric are not generally the same, so in order to evaluate the real obtained performance when using the approximated metrics in the optimization problem some experiments have been carried out.

When the optimized metric depends on the previous channel estimate a block per block transmission has to be modeled. Moreover, it is important to note that the statistical information from block to block is therefore influenced. Specifically, the system functions as indicated in figure 5.

In order to obtain a significant evaluation of the performance, the experiment has been designed to obtain averaged performances, when transmitting several blocks, and considering several initial channels. Note there is no reason to assume that the use of one of the problems or approximations should provide a better performance than

1 Singular Value Decomposition

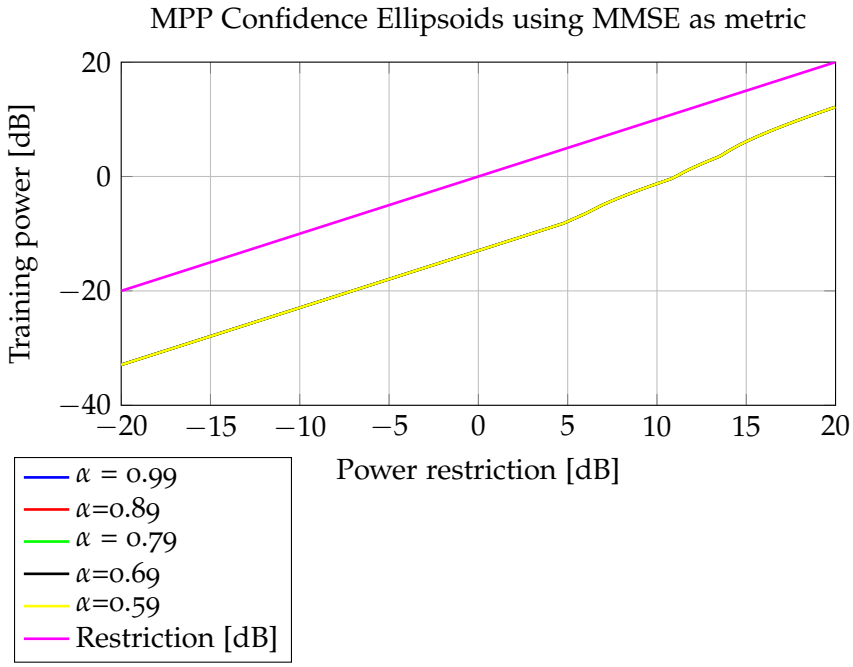


Figure 4.: Restriction fulfillment when using the Confidence Ellipsoid approximation for MPP using MMSE as metric, with parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$ and 2000 averages.

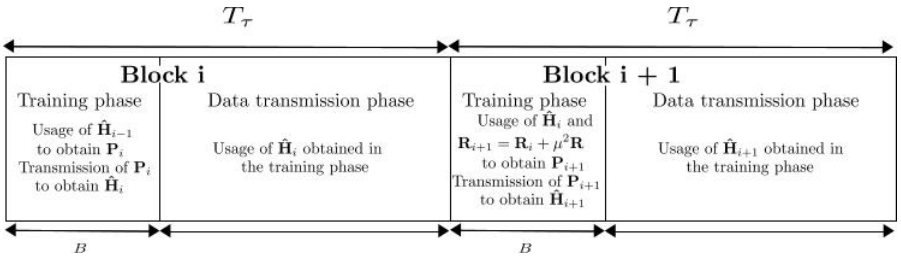


Figure 5.: Diagram of block per block usage of the channel estimate

the other. Therefore, for simplicity the GPP averaged case is considered.

### 10.3.1 *Zero Forcing Precoder*

As already mentioned in the previous sections, the optimization metric for the Zero Forcing Pre-coder (104) is an approximation of (105). In order to evaluate how tight this approximation is, an experiment with several transmission blocks and several initial channels has been carried out. In other words, one initial channel $\mathbf{H}_0$ is drawn from a complex Gaussian distribution with $\mathbf{0}$ mean and $\mathbf{R}$ covariance matrix, is used to generate the metric matrix by being fed in equation (104)

which is then used, together with the statistical characterization of the channel in the optimization problem.

Once the training sequence resulting of the optimization is obtained, it is transmitted through a channel that is generated following the first order Markov model, i.e. $\mathbf{H}_1 = \mathbf{H}_0 + \mu\mathbf{E}_1$. Once $\mathbf{P}$ is transmitted using the realization of $\mathbf{H}_1$, the corresponding channel estimate and error are obtained and evaluated in (105), obtaining the real value of the expression in (101).

The outage probabilities, which are the probability of performing worse than the required performance, are estimated as well by comparing the obtained performance corresponding to each channel realization and comparing it to the performance restriction. 2000 realizations of channel $\mathbf{H}_1$ are considered and the $2000^{\text{th}}$ channel estimate is then used to generate the next metric matrix. This procedure is repeated 20 times, meaning that 20 different blocks have been simulated. Additionally, 20 blocks are transmitted for each of the 10 different initial channels. All the average performances and outage probabilities plotted are obtained through averaging all the blocks and initializations for each performance restriction.

Note that at each transmitted block, the statistics of the channel change according to the expression in figure 5.

Following these directions the results in figures 6 and 7 were obtained. Moreover, for the sake of comparison, the training sequence that minimizes the mean square error of the channel estimation is computed and transmitted obtaining its corresponding channel estimate and error. These are fed in both (104) and (105) to illustrate the significance of minimizing a system specific metric.

Figures 6 and 7 illustrate two important facts:

1. If the metric in (105) is actually representative of the system's BER when using a ZF pre-coder, the optimized $\mathbf{P}$ performs better than the one obtained by minimizing the channel's mean square error.

2. The approximation is very tight on average, and does not seem to have a very severe impact on the instantaneous performance.

In order to provide a quantitative reference related to this last point, the outage probabilities are computed in both the approximated and the real case with the analytical restriction as reference. The outage probability is estimated using Monte Carlo and each of the obtained outage probabilities plotted have been estimated using 2000 experiments.

These outage probabilities are shown in figure 8. Which illustrates that the impact is tolerable considering $\mu = 0.01$. It is left for future work to evaluate what increase in $\mu$ would lead to a very poor performance of this approximation, and if it would entail that it is preferable to use the MMSE since the cost function does not depend

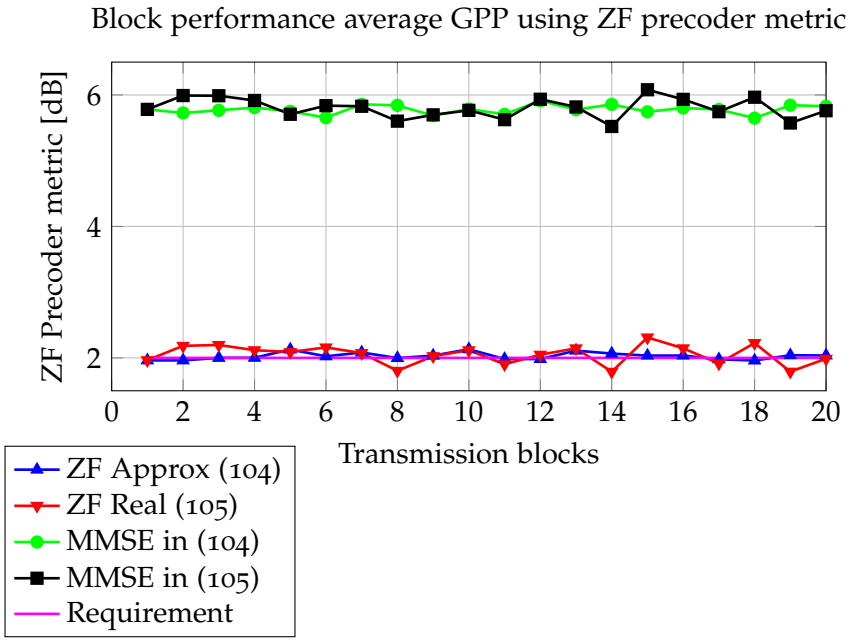


Figure 6.: Evaluation of approximated block to block ZF metric using Averaged Performance GPP. Results obtained with $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, 10 initial channels, 20 blocks, 2000 averages, SNR $= 20$ and $\mu = 0.01$.

on the previous channel estimate, but on the covariance matrices exclusively.

### 10.3.2 *MMSE Equalizer*

The exact same strategy as in the ZF precoder applies to the MMSE Equalizer case, although an analysis needs to be carried out for the 3 different approximations that are provided in table 3. The training sequence is obtained through each of the 3 approximations and then the obtained estimates and errors are fed in equation (109). The results of these experiments have been carried out for one initial channel, due to the computational cost the numerical integrations entail, that has been used to generate a sequence of 10 blocks. Each training sequence obtained for each block has been transmitted using 2000 different channel realizations that follow the first order Markov model.

In order to pick a representative realization of the channel, 100 channel realizations have been generated and, the one which, evaluated in the PDF, provided the highest value, was selected. The results of these experiments are shown in figures 9, 10. The parameter $\Delta$metric is defined as in (112).

$$\Delta\text{metric} = 10\log(\text{Optimized metric}) - 10\log(\text{Real metric}) \qquad (112)$$

Where the *Optimized metric* is obtained by averaging estimation errors obtained through 2000 realizations and transmissions of 10 different

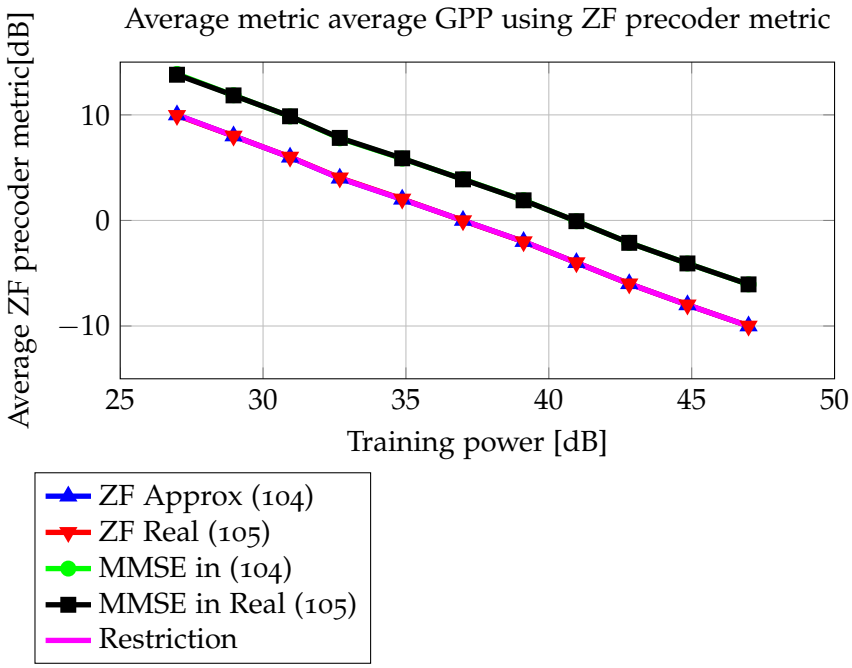


Figure 7.: Evaluation of approximated average ZF metric using Averaged Performance GPP. Results obtained with $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, 10 initial channels, 20 blocks, 2000 averages, SNR $= 20$ and $\mu = 0.01$.

blocks, i.e. 2000 realizations of $\mathbf{H}_i$ following $\mathbf{H}_i = \mathbf{H}_{i-1} + \mu\mathbf{E}_i$, $i = 1, \ldots, 10$, for all the equations that appear in table 3. This has been done separately for each of the cases depending on the rank of the channel matrix $\mathbf{H}$ and the data transmission phase SNR. During the transmissions, once the channel estimate and the estimation error are obtained, equation (109) is evaluated with these. This way, information about the real obtained performance during the transmission of data during the specific block is obtained. This parameter is then averaged through the 2000 realizations and is represented in equation (112) as *Real metric*.

From the results plotted in figures 9 and 10, it is possible to conclude that the first two approximations in table 3, i.e. the ones with high data transmission SNR, provide a good reference both instantaneously and in average, while the one used for low SNR cases diverges more. As a matter of fact, both approximations used when having high SNR during the transmission phase provide higher real performances while the one in low SNR does not.

As mentioned in [10] the training sequence obtained by minimizing the approximated metric and the training sequence obtained by minimizing the channel estimation mean square error perform very much alike. Therefore, from now on the analysis carried out will be done with the Zero Forcing precoder metric.

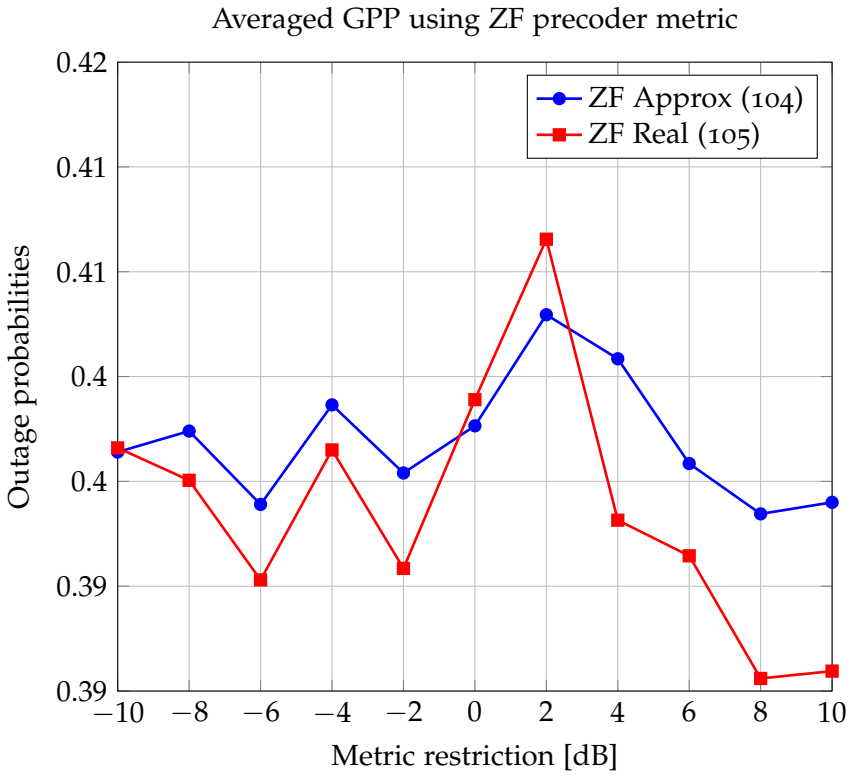


Figure 8.: Outage probabilities corresponding to the experiment in figure 6

## 10.4 CONFIDENCE ELLIPSOIDS AND MARKOV BOUND

In this section the description of the experiments carried out to evaluate which of the two approximations performs better is included. In both cases, a probability related to the outage probability is specified. As already illustrated in figures 1 and 3 the metric restriction is active in the SDP-GPP cases while the power restriction is not in the SDP-MPP cases. Therefore, for simplicity i.e. in order to avoid power equalization, this specific analysis will be done using the GPP formulation as basis. In the experiments, the restriction in performance is the same for both approximations and only $\alpha$ or $\epsilon$ will vary. Since the parameter $\epsilon$ should actually represent the outage probability, $\alpha$ is picked to be fix while $\epsilon$ is equalized in order to obtain the same outage probability. Experimentally, this means that the parameter $\epsilon$ has been increased or decreased until the estimated outage probability of both Confidence Ellipsoids and Markov Bound is the same. Therefore, considering the outage probability to be fixed and specifying a certain performance that both methods will try to achieve, the only degree of freedom left is the compromise average performance - training power that will illustrate which of the two methods provides the best solution.

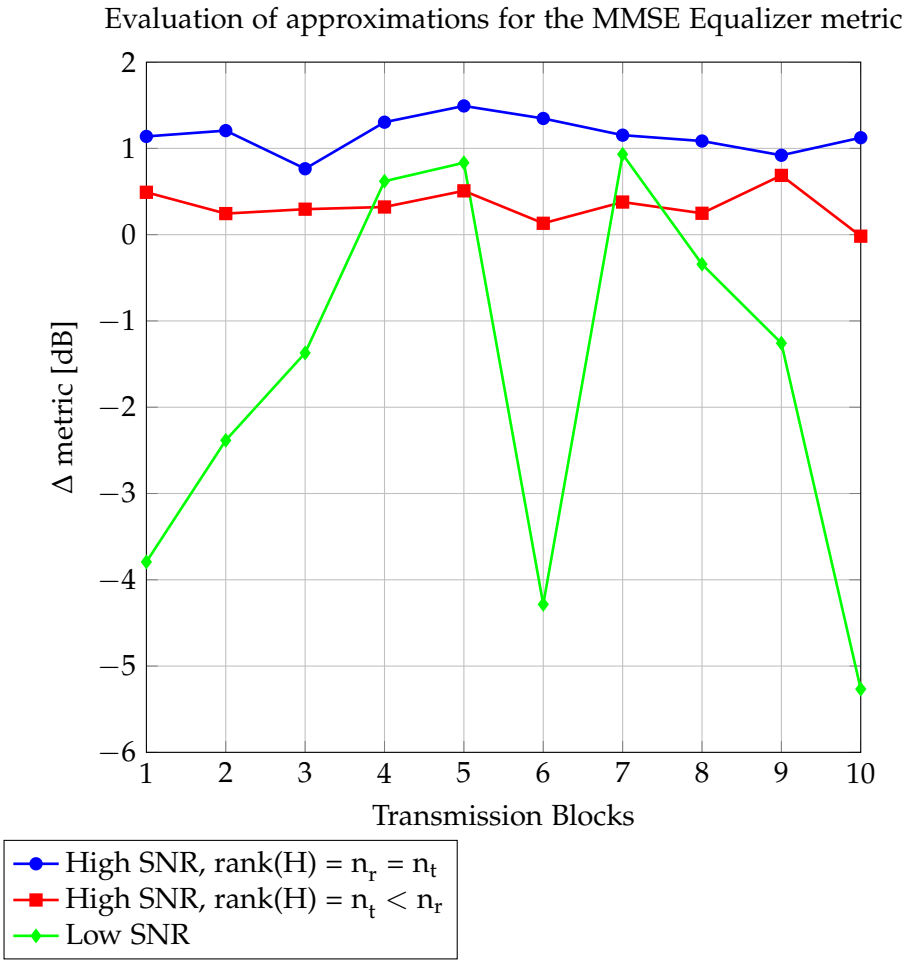


Figure 9.: Evaluation of approximated block to block MMSE Equalizer metrics in table 3 using Averaged Performance GPP, with parameters $n_T = n_R = 4$, $SNR = 20\,\text{dB}$ in the first case, $n_t = 2$, $n_r = 4$ and $SNR = 20\,\text{dB}$, in the second case and $n_t = n_r = 4$ and $SNR = -5\,dB$ in the third case, $\rho = 0.7$, 10 blocks and 2000 averages.

### 10.4.1 *MMSE*

The above described experiment has been carried out for the channel's estimate mean square error. The results are portrayed in figures 11 and 12.

Both the Confidence Ellipsoid and the Markov Bound based approximations lead to good performances in case of using the channel estimate mean square error. Specifically, the training sequence obtained by using the Confidence Ellipsoid approximation outperforms the one obtained by using the Markov Bound approximation. As mentioned in chapter 7, the Confidence Ellipsoid approximation fits an ellipsoid that represents events, estimation errors, that occur with

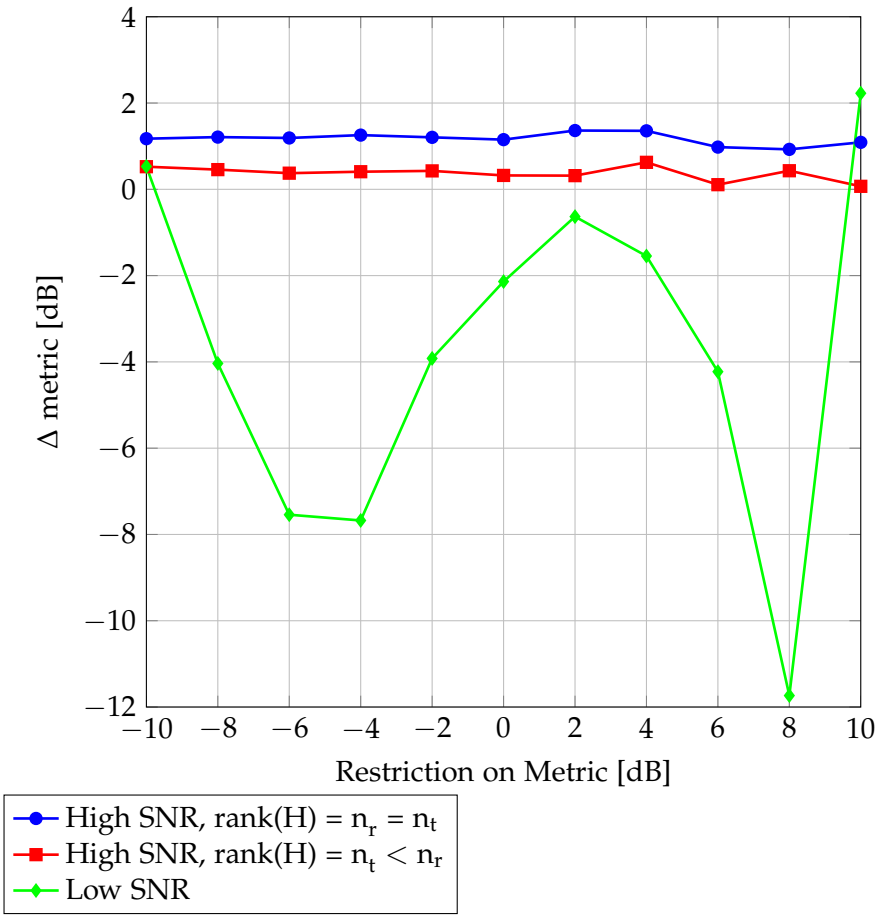


Figure 10.: Evaluation of average approximated MMSE Equalizer metrics in table 3 using Averaged Performance GPP, with parameters $n_T = n_R = 4$, $SNR = 20\,\text{dB}$ in the first case, $n_t = 2$, $n_r = 4$ and $SNR = 20\,\text{dB}$, in the second case and $n_t = n_r = 4$ and $SNR = -5\,dB$ in the third case, $\rho = 0.7$, 10 blocks and 2000 averages.

probability $\alpha$ in a ellipsoid that, using the metric matrix, represents the required performance.

An increase in the parameter $\alpha$ represents a larger ellipsoid to be included in the second one, and therefore an added difficulty. Hence, in the case illustrated in figure 11 using the Confidende Ellipsoid approximation does not outperform the use of the Markov Bound approximation as much as in the case in figure 12.

Additionally, the outperforming of the Confidence Ellipsoid approximation might be due to the use of the MMSE as metric. The metric matrix in the case of MMSE describes a hypersphere, which is not skew in any of the dimensions and therefore, might be easier to adapt to.

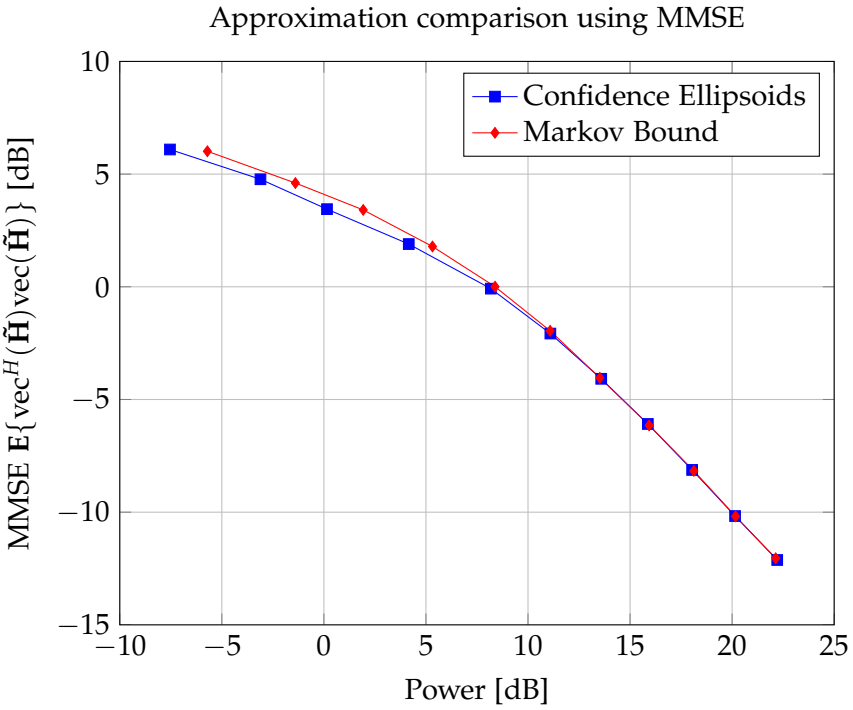


Figure 11.: Evaluation of approximations using GPP with MMSE metric with parameters: $n_T = 4, n_R = 2$, $B = 6$, $\rho = 0.7$, $\alpha = 0.5$ and 2000 averages.

### 10.4.2 *Zero Forcing Precoder*

The same experiment has been carried out with the Zero Forcing Precoder. Only one block has been simulated in each case, since two different training sequences generate two different channel estimates which lead to a different metric in the following block. This contradicts the aim of this experiment which is to evaluate the performance of the two different approximations given the same metric. Results from this experiment are illustrated in figures 13 and 14.

In this case it is easier to find a training sequence $\mathbf{P}$ that provides a small projection of $\mathbf{C}_{MMSE}$ over $\mathcal{I}_{adm}$, than a training sequence that allows $\mathbf{C}_{MMSE}^{-1}$ to have the same shape as $\mathcal{I}_{adm}$.

Considering that the Markov Bound approximation provides a better relation performance - training power according to figures 13 and 14, from now on only the Averaged Performance cases will be considered due to simplicity and its equivalence to using the Markov bound approximation.

## 10.5 AVERAGED PERFORMANCE

Due to simplicity the average performance case will be considered from now on. Additionally, it has already been showed that the Markov Bound approximation for the ZF equalizer case, which is

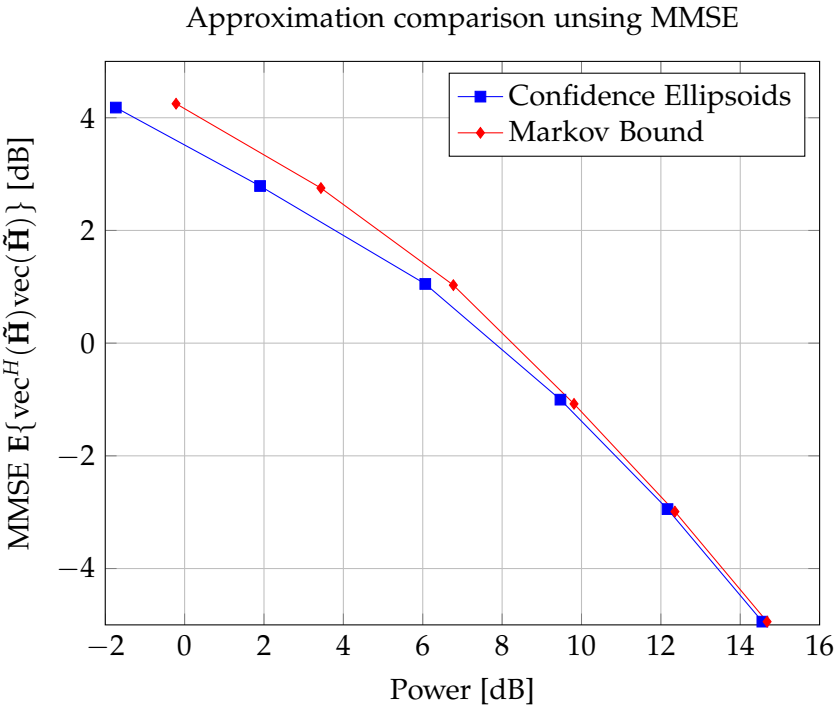


Figure 12.: Evaluation of approximations using GPP with MMSE metric with parameters: $n_T = 4, n_R = 2$, $B = 6$, $\rho = 0.7$, $\alpha = 0.1$ and 2000 averages.

equivalent to the averaged performance for $\epsilon = 1$, performs better than the confidence Ellipsoid approximation. In this experiment section the two different problem formulations are considered for the Zero Forcing pre-coder metric. The MMSE Equalizer is disregarded due to time limitations. The Zero Forcing pre-coder has been prioritized since the optimization of the training sequence provides an improvement in the resulting metric while in the case of the MMSE Equalizer the performance is very similar to the one obtained when using the MMSE as metric.

In order to be able to compare the performance of the training sequence resulting of both problems, it is required that both use the same training power. The intuitive way of forcing this to happen, is to find the solution to the GPP, calculate its power and use it as restriction for the MPP.

As shown in figure 1 this can not be done since in the SDP-MPP the power restriction is not active.

### 10.5.1 *MPP: The need of power equalization*

Considering the facts stated above, it is necessary to force the MPP to reach the required power in order to be able to compare the solution provided by each problem. For this purpose, first of all a power restriction that forces the MPP to reach a higher power than required is

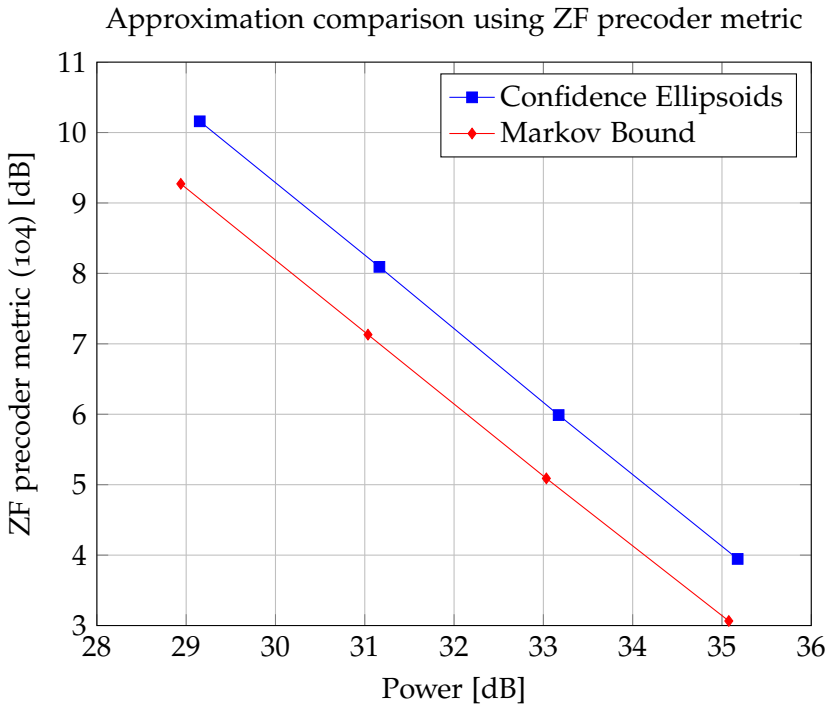


Figure 13.: Evaluation of approximations using GPP with ZF pre-coder metric with parameters: $n_T = 4, n_R = 2$, $B = 6$, $\rho = 0.7$, $\alpha = 0.5$, SNR = 20 dB and 2000 averages. 10 Channel Realizations.

found. Once this is done, the power constraint that forces the desired training signal power is found by bisection.

### 10.5.2 *GPP and MPP: a performance comparison*

The results of the experiment using the metric for the ZF pre-coder are shown in figures 15 and 16.

The conclusion one can get of these results is that when using power equalization the Averaged MPP and GPP provide very similar solutions, both in performance and outage probabilities. As a matter of fact, the training sequences are practically identical and it is possible that the differences are due to numerical errors.

Considering the results at hand, there is no inherent advantage in any of the two problems with respect to the other. Using GPP or MPP will only depend on if the system is forced to guarantee a specific performance independently of the power, or use a specific power independently of the performance. With the last option, it is necessary to keep in mind that power equalization is required.

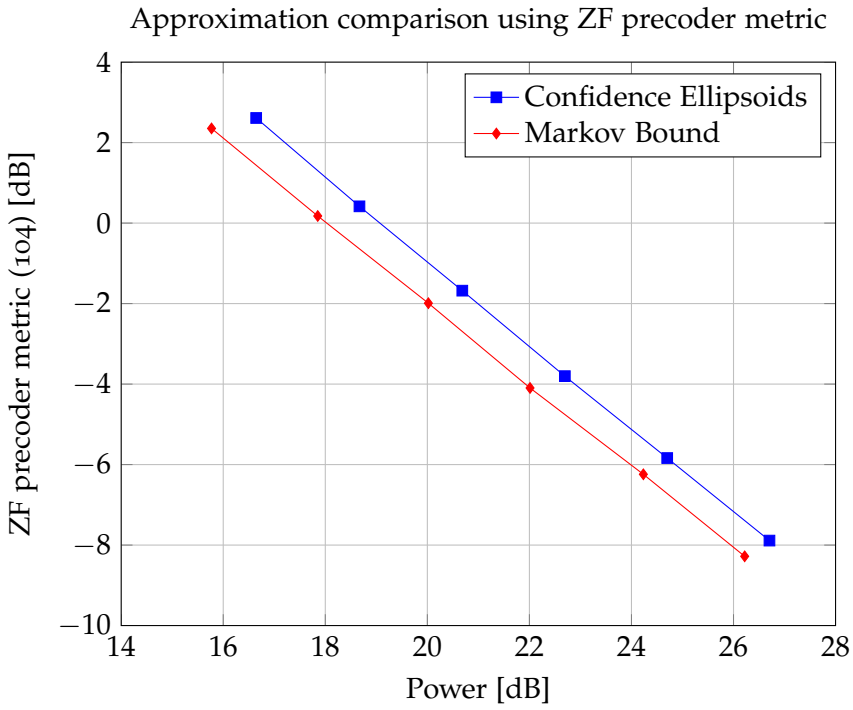


Figure 14.: Evaluation of approximations using GPP with ZF pre-coder metric with parameters: $n_T = 4, n_R = 2$, $B = 6$, $\rho = 0.7$, $\alpha = 0.1$, SNR = 20 dB and 2000 averages. 10 Channel Realizations.

### 10.5.3 *How far from optimal are the solutions?*

In order to provide an idea of how close from optimal the training sequence obtained from the SDP - Markov Bound formulation is, its the performance has been compared to the performance of the training sequences obtained using the built-in optimization function in *MATLAB*, *fmincon*, which is capable of guaranteedly converging to a local minimum of the averaged performance GPP and MPP with no approximations, provided an initial point.

In the experiments, the solution obtained from the SDP formulation is provided to *fmincon* as initial point, as well as 100 random white Gaussian sequences that are guaranteed to fulfill the constraints before being set as initial points. Specifically, the function has been set to use interior-point methods to converge to a solution.

In the figures that illustrate the solutions the performance or power and the outage probabilities of:

1. the training sequence obtained from the SDP formulation,
2. the training sequence obtained after the optimization using the SDP solution as initial point,
3. the best solution obtained with the white initialization,
4. the average solution obtained with the white initialization and

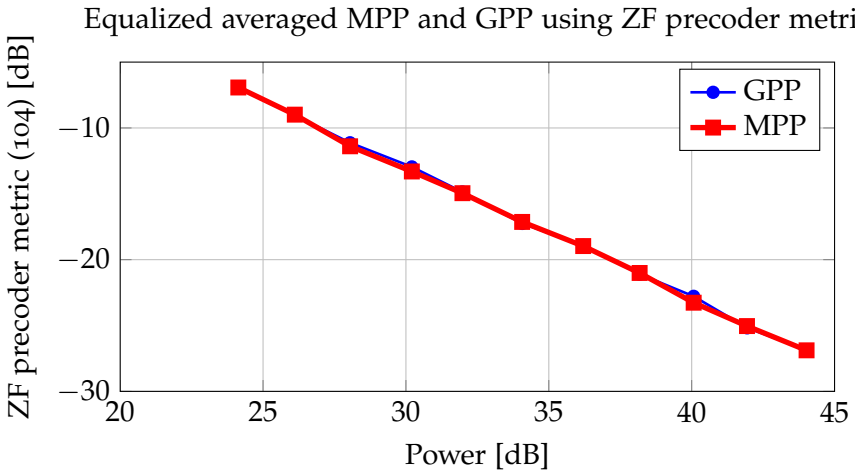


Figure 15.: GPP with ZF pre-coder metric using power equalization to compare averaged MPP and GPP solutions, with parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, SNR = 20 dB, $\mu = 0.01$, 10 initial channels, 20 blocks and 2000 averages.

5. the worst solution obtained with the white initialization,

are represented.

#### 10.5.3.1 *GPP: MMSE*

The first of all experiments of this kind has been carried out with a GPP formulation using the MMSE as metric. The results are provided in figures 17 and 18.

As shown in figure 17, the solution provided by *fmincon* when using the SDP solution as initial point performs better than the rest. In some cases solutions that are derived from a random initial sequence perform better than the SDP solution although it is not a general tendency.

#### 10.5.3.2 *GPP: Zero Forcing Pre-coder*

The same experiment has been carried out with the zero forcing pre-coder metric. This experiment has the particularity that, the number of transmitted blocks is set to 1. In this case, since the performance of the SDP is being evaluated, independently of how the metric performs, the block to block evolution of the metric is irrelevant. Again, a relevant channel is picked as initial for each of the points. The results of this experiment are illustrated in figures 19 and 20.

In this case, it is very much obvious that the SDP solution performs better than any of the solutions obtained using white initial sequences. Following the tendency of the previous experiment, the solution provided by *fmincon* with the SDP solution as initial point improves slightly, but more than in the MMSE case.

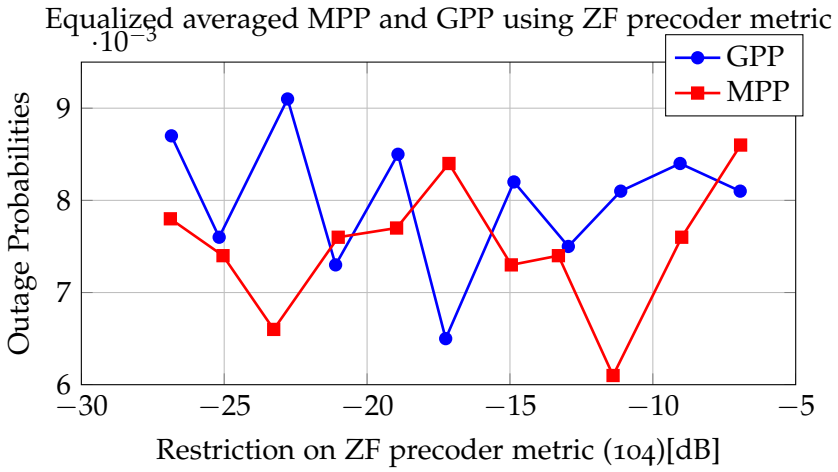


Figure 16.: GPP with ZF pre-coder metric using power equalization to compare averaged MPP and GPP solutions, with parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, SNR = 20 dB, $\mu = 0.01$, 10 initial channels, 20 blocks and 2000 averages.

#### 10.5.3.3 *MPP: MMSE*

In this case the experiments have been carried out with the MPP problem. The methodology is analogous to the GPP with the difference that a equalization is required to obtain a training sequence that reaches the power constraint. The solution of the SDP after equalization is fed to the *fmincon* function. The results of this experiment are shown in figures 21 and 22.

Again, the training sequence obtained by using the SDP solution as initial point performs better than the rest, and slightly better than the SDP solution itself.

#### 10.5.3.4 *MPP: Zero Forcing Precoder*

In this case the same exact methodology as in the GPP zero forcing pre-coder case as been used, with the additional fact of equalizing the power of the training sequence obtained from the SDP solution. The results of this experiment are illustrated in figures 23 and 24.

As in the previous cases the performance of the SDP solution is better than the ones resulting by using the white sequences. In the same way as in the GPP case, the solution provided by *fmincon* function when using the SDP solution as initial point performs best, and with a greater margin than in the MMSE case.

#### 10.5.3.5 *GPP and MPP: Equivalent problems*

According to the results obtained in figure 15 the solutions of the SDP formulations of the GPP and MPP problems result in very similar training sequences. Additionally, a similar result in performance

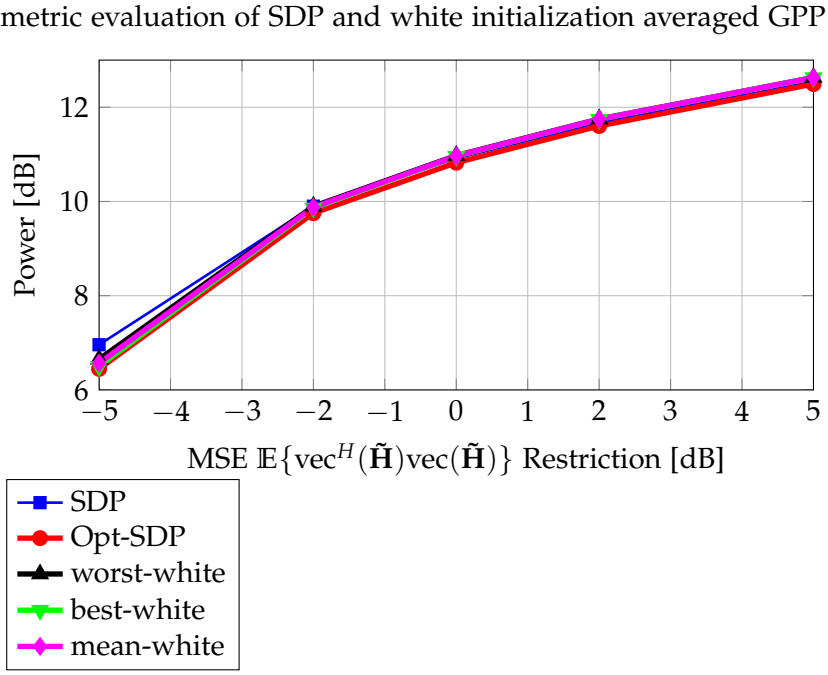


Figure 17.: Power of different solutions for the Averaged GPP using MMSE as metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, and 2000 averages.

is obtained both in figures 19 and 23 when these solutions are provided to the *fmincon* function. In order to verify if this behavior still holds after the optimization by the use of the function *fmincon* an experiment has been designed. In this experiment, an initial point is obtained through solving the averaged GPP in SDP form. After that, this initial point is fed to *fmincon*. The power of the resulting solution is computed and fed as restriction, with later power equalization, to the averaged MPP in SDP form. This point is fed then again to *fmincon* to solve the MPP. In this case, the ZF pre-coder metric has been used for only one channel realization and one block. The obtained result is illustrated in figure 25.

Figure 25 reveals that the performance - training sequence power ratio is practically identical when using the two methodologies. Considering that the SDP formulation of the MPP and GPP provide almost identical solutions, and lead to same solution when the non-relaxed version of the problem is at hand, it is possible to conclude that the two problems are equivalent. A mathematical proof of this is included below.

Recall that the problems solved by *fmincon* are the ones in equations (113) and (114).

$$\begin{aligned} \text{minimize} \quad & \text{tr}((\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tilde{\mathbf{P}})^{-1} \mathcal{I}_{adm}) \\ \text{s.t.} \quad & \text{tr}(\mathbf{P}\mathbf{P}^H) \leq \mathcal{P} \end{aligned} \tag{113}$$

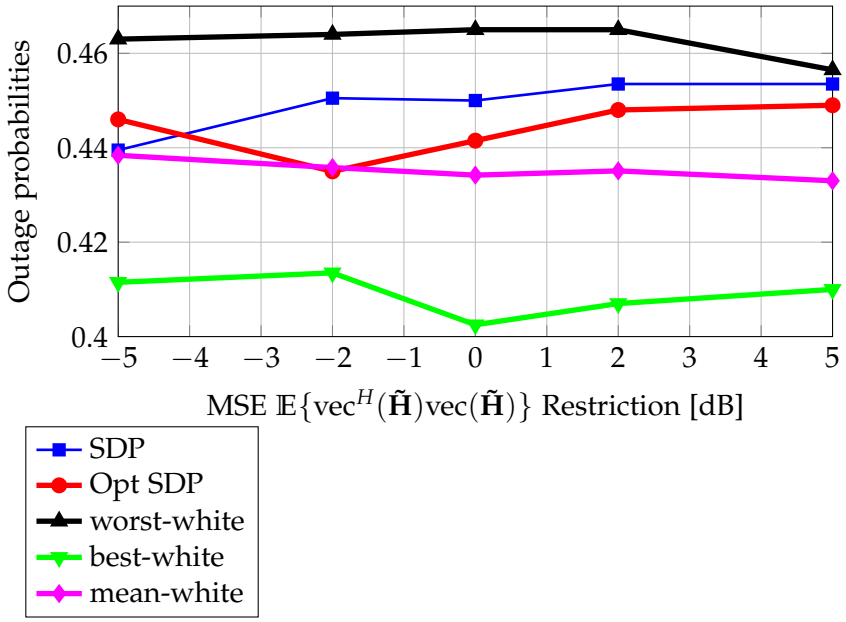


Figure 18.: Outage probabilities of different solutions for the Averaged GPP using MMSE as metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, and 2000 averages.

$$\begin{aligned} &\text{minimize} && \text{tr}(\mathbf{P}\mathbf{P}^H) \\ &\text{s.t.} && \text{tr}((\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}})^{-1}\mathcal{I}_{adm}) \leq \frac{1}{\gamma} \end{aligned} \tag{114}$$

Consider now that the function $\text{tr}((\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H\mathbf{S}^{-1}\tilde{\mathbf{P}})^{-1}\mathcal{I}_{adm})$ can be written as in (115)

$$\frac{1}{\mathcal{A}^2}\text{tr}((\mathbf{R}^{-1}/\mathcal{A}^2 + \tilde{\mathbf{P}}_n^H\mathbf{S}^{-1}\tilde{\mathbf{P}}_n)^{-1}\mathcal{I}_{adm}), \tag{115}$$

where $\tilde{\mathbf{P}}_n$ is equivalent to matrix $\tilde{\mathbf{P}}$ but normalized to have unitary power, and the constant $\mathcal{A}$ is the scaling $\tilde{\mathbf{P}}_n$ requires to recover $\tilde{\mathbf{P}}$. This implies that, the metric will decrease when the power of the training sequence $\mathcal{A}^2$ is increased since the term dominates the entire expression. Analogously, function $\text{tr}(\mathbf{P}\mathbf{P}^H)$ can be expressed, using the same notation, as in equation (116).

$$\mathcal{A}^2\text{tr}(\mathbf{P}_n\mathbf{P}_n^H) \tag{116}$$

In this case, the function is increased when the constant $\mathcal{A}$ increases, which is the exact opposite behavior, with respect to $\mathcal{A}$, that is found in (115).

Being this so, in order to minimize the metric function, i.e. first equation in (113), a training sequence $\mathbf{P}$ will be selected so that is

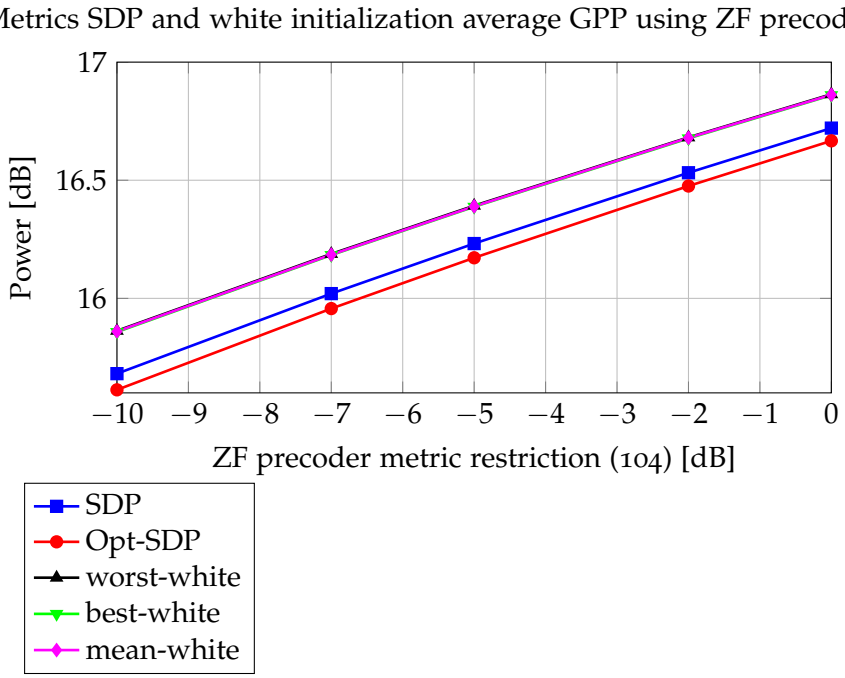


Figure 19.: Metric of different solutions for the Averaged GPP using ZF precoder metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, SNR = 20, and 2000 averages.

tightly fulfills the power restriction since a minimization of the metric is intended. In other words, minimizing the training sequence power as much as possible will lead the problem to achieve the high bound set on the metric since a decrease in power leads to an increase in metric and vice versa.

In the experiments, the value of the cost function in the optimal point found is fed as restriction to the MPP, meaning that, in this case the MPP formulation can be written as (117)

$$\begin{aligned} \text{minimize} \quad & \operatorname{tr}((\mathbf{R}^{-1} + \tilde{\mathbf{P}}^H \mathbf{S}^{-1} \tilde{\mathbf{P}})^{-1} \boldsymbol{\mathcal{I}}_{adm}) \\ \text{s.t.} \quad & \operatorname{tr}(\mathbf{P}\mathbf{P}^H) \leq \operatorname{tr}(\mathbf{P}_{Gopt} \mathbf{P}_{Gopt}^H), \end{aligned} \tag{117}$$

where $\mathbf{P}_{Gopt}$ is the solution found for the GPP problem in (114). Finding this point implies that no point that provides a metric $\frac{1}{\gamma}$ uses less power than $\operatorname{tr}(\mathbf{P}_{Gopt} \mathbf{P}_{Gopt}^H)$.

In the case in (117), the power restriction will be fulfilled tightly as well, since in order to minimize the metric as much as possible the higher bound on the power will be reached.

A point that uses this amount of power and provides a metric over $\frac{1}{\gamma}$ would contradict the problem in (117), since from (114) it is known that a training signal that provides a smaller metric with the same power exists.

On the other hand, if the metric obtained were below $\frac{1}{\gamma}$ this would contradict the problem in (114), since then a scaled version of the

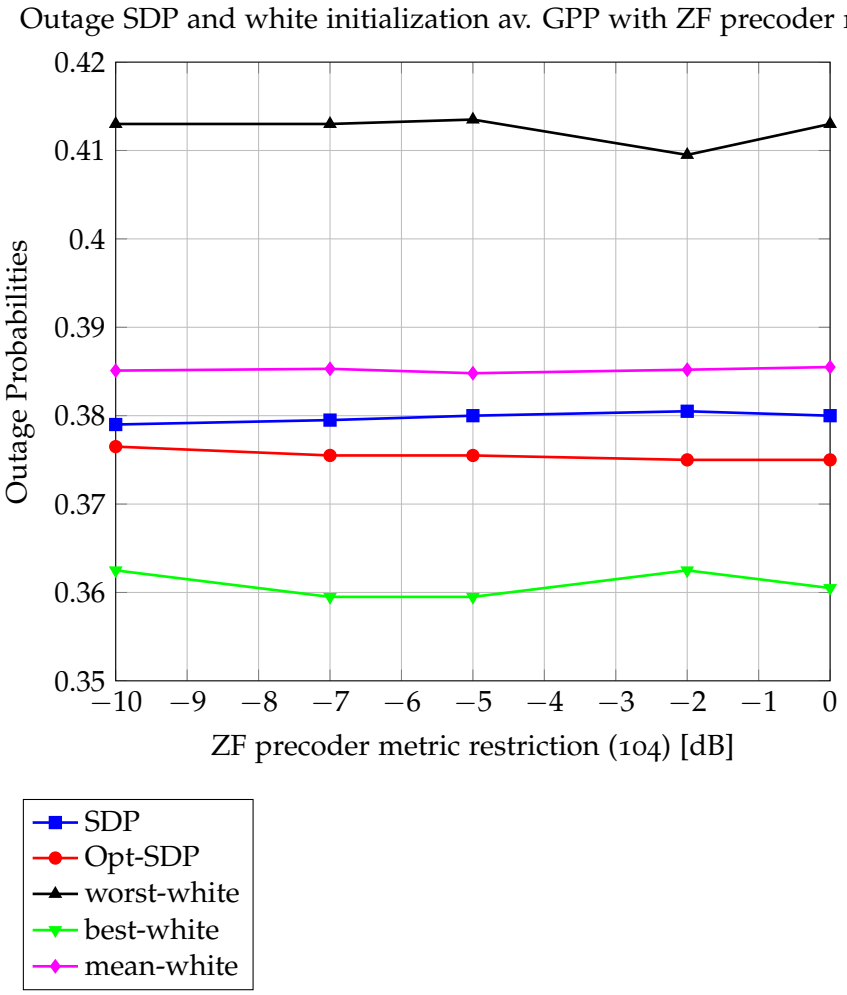


Figure 20.: Outage probabilities of different solutions for the Averaged GPP using ZF precoder metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, SNR = 20, and 2000 averages.

optimal training sequence found with (117), in order to fit the performance requirements, would provide a lower training power requirement. Hence, the two problems are mathematically equivalent which is consistent to the results in figure 25.

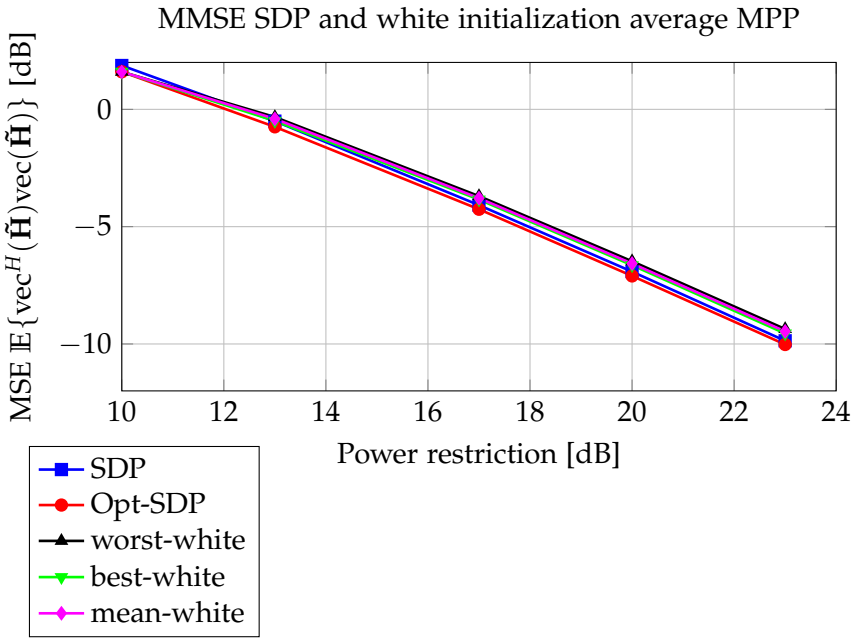


Figure 21.: Metric of different solutions for the Averaged MPP using MMSE as metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, and 2000 averages.

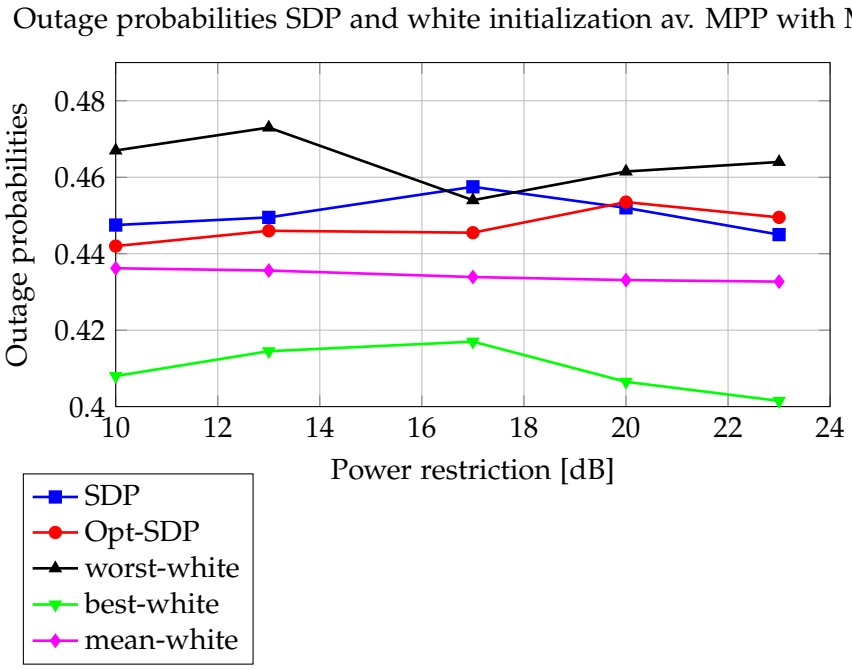


Figure 22.: Outage probabilities of different solutions for the Averaged MPP using MMSE as metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, and 2000 averages.

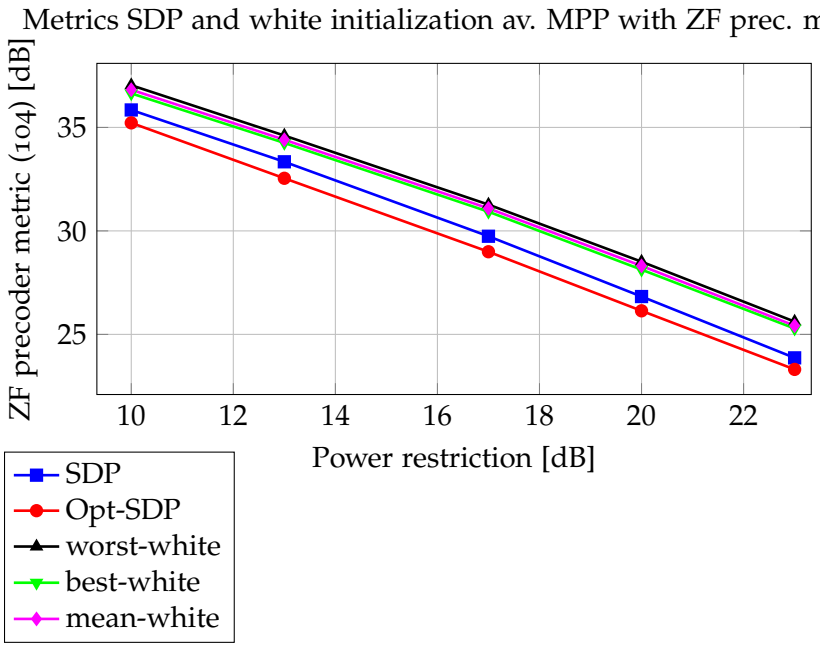


Figure 23.: Metric of different solutions for the Averaged MPP using ZF precoder metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, SNR = 20, and 2000 averages.

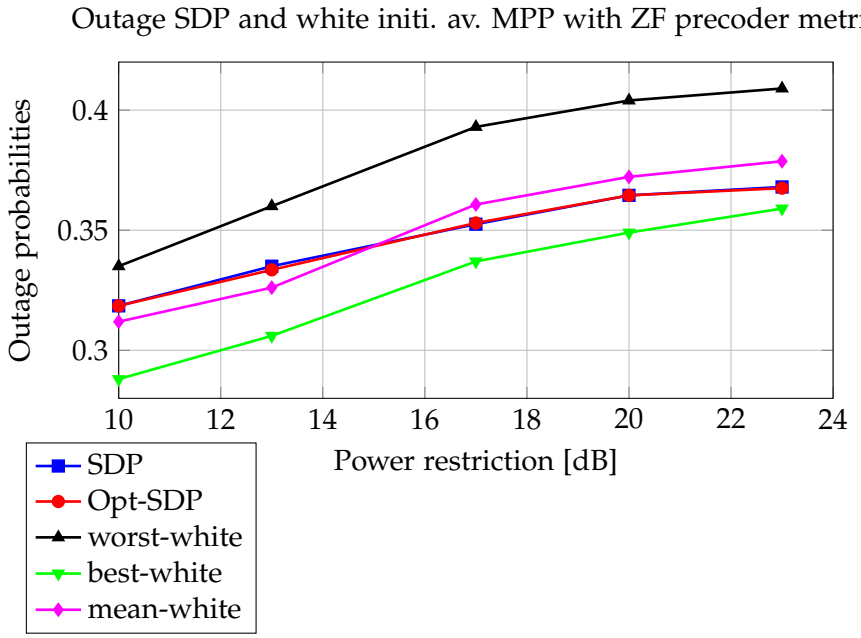


Figure 24.: Outage probabilities of different solutions for the Averaged MPP using ZF precoder metric with system parameters $n_T = n_R = 4$, $B = 6$, $\rho = 0.7$, SNR = 20, and 2000 averages.

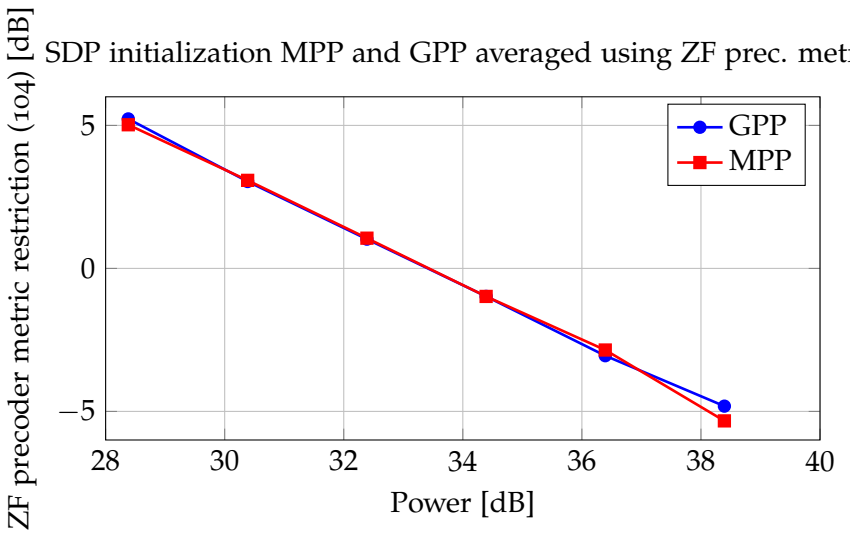


Figure 25.: Result obtained by using the SDP solution as initial point to *fmincon* for both averaged GPP and MPP, with parameters $n_T = n_R = 4$, $B = 6$, SNR = 20 dB, $\rho = 0.7$ and 2000 averages.

# Part IV

# CONCLUSIONS

# 11

## SUMMARY

In this thesis different alternatives to find an optimal, application oriented, training sequence to estimate a MIMO channel have been analyzed. In order to be able to motivate the steps followed during this master thesis some preliminary knowledge about convex optimization has been provided. Additionally, MIMO channels and their modeling is introduced in order to ease the comprehension of the thesis.

Two different problem formulations have been considered using the two major relevant design parameters, which are performance and power. The first approach intends to minimize the training sequence power while guaranteeing a specific performance (GPP[1]), while the second approach maximizes the performance given a specific training power budget (MPP[2]). Once these two different problems are formulated, two different approximations based on confidence ellipsoids and the Markov Bound are analyzed and applied to each of the problems. These approximations, together with some other, allow to convexify the problem.

Later on, additional reformulations and approximations, based on the Schur-complement and the Cauchy inequality respectively, are forced upon the relaxed problems to be able to express them as a Semi Definite Programs. Once this has been done, metrics designed for specific systems are considered. Specifically, the Zero Forcing precoder and the MMSE Equalizer have been considered since they act linearly on either the transmitted or received signal. The system specific metric for the Zero Forcing precoder requires the knowledge of the current channel, which of course is unknown. Therefore, a first order Markov model is forced upon the channel and the previous channel estimate is used instead. In the case of the MMSE Equalizer the system specific metric is highly non-convex, and some approximations are needed to obtain a convex expression.

The performance, outage probabilities and restriction fulfillment of both the MPP and GPP formulation are analyzed during the master thesis, together with an analysis of the impact of the mentioned approximations. Additional experiments have been carried out in order to measure the impact on performance the approximated metrics

1 Guaranteed Performance Problem
2 Maximized Performance Problem

have on the system, and a comparison to using a non-system specific is provided.

Finally, in order to try to evaluate how far from optimal the solutions to the SDPs are, MATLAB optimization routines initialized in the SDP solution and white sequences are used to solve the original problems.

# 12

# CONCLUSIONS

The experiments carried out in the thesis suggest that approximation based on the Markov Bound performs better than the one based on Confidence Ellipsoids for the Zero Forcing precoder metric.

The Confidence Ellipsoids approximation requires the inverse of the estimation error covariance matrix to have the same shape as the metric matrix for the restriction to be tight. If this shape is not easy to achieve, more power will be used in reaching the required performance than the strictly necessary.

This can as well be interpreted as the difficulty of modifying the shape of the confidence ellipsoid to the one defined by the metric. If the ellipsoid defined by the metric is difficult to achieve, a higher power will be required by the training sequence to reduce the confidence ellipsoid and still be representing the same probability. On the other hand, the Confidence Ellipsoid approximation allows to find better solutions when using the channel estimate's MSE. This implies that when the shape limitation is not specially skew in any of the directions, and therefore shape considerations do not represent the main difficulty, the use of the Markov Bound approximation is outperformed by the use of the approximation based on Confidence Ellipsoids. This may have its origin in the statistical considerations forced upon the confidence ellipsoid during its design, in which the shape of the distribution is taken into account. It is considered that further analysis is required to completely explain the behavior of both approximations, which is left for further work.

The approximation used, to be able to express the problems as SDPs, also have a significant impact on the solution. The Cauchy inequality negatively impacts on the MPP formulation since the restriction on power is never tight. This has its origin in the fact that the Cauchy inequality is only met with equality when the two vectors at hand are parallel. In this specific case, one of the two is an optimization variable and no information about the second, which represents the noise covariance in time, is provided in the optimization problem. This implies that it is rather unlikely that the optimal vector and the vectorized covariance matrix are parallel. The use of the Cauchy inequality also reduces the problems' feasible set which can leave the optimal solution out.

Additionally, the approximated restriction favours some solutions over others. Although this last fact impacts all the SDP formulations considered in the thesis, by solving the SDP-GPP formulation using Confidence Ellipsoids, for specific cases in which an analytical solution can also be found, the optimal point is found. This means, that for these cases the approximation has no detrimental impact on the feasible set, i.e. the optimal point still belongs to it.

The system specific metrics when assuming a first order Markov model on the channel, do not severely impact the system's performance if the channel varies slowly. All the metrics and approximations considered perform well excepting for the MMSE Equalizer metric for low SNR. Additionally, when compared to the training sequence obtained by minimizing the estimation's mean square error, the resulting performance is very similar to the result when using the system specific metric for the MMSE Equalizer.

The training sequences obtained by solving both the averaged MPP and GPP using the ZF precoder metric and MMSE are compared to the solution provided by optimizing the original problem with MATLAB built-in function *fmincon* using white sequences and the solution to the SDPs as initial points. The result of this last optimization outperforms all the rest, and generally the SDP solutions perform better than the local minimum found by *fmincon* when using white sequences as initial points. Finally, both problems are experimentally shown to be equivalent in original form, and in convexified form when power equalization is applied to the MPP formulation.

# 13

## CONTRIBUTIONS

The contributions provided by this thesis are:

- An application of the experimental method proposed in [11] to solve the problems formulated in [10] and its corresponding analysis.
- An analysis of the impact of the approximations used for the application specific metrics proposed in [10] on the system's performance.
- A theoretical and experimental assessment of the Confidence Ellipsoid approximation and the Markov Bound approximation.
- An evaluation of how far from optimal the solution of the SDPs are compared to the solutions obtained when using white sequences as initial points for local minimizing methods.

# 14

## FURTHER WORK

Possible directions for further work could be:

1. Analyze the impact on the increase of the parameter $\mu$ in the cases in which the metric relies on the previous channel estimate and compare it to the performance obtained by minimizing the channel's mean square error.
2. Use an alternative relaxation in the MPP problem to obtain a SDP formulation that does not rely on the Cauchy inequality, using, for example, the strategy suggested in [14].
3. Consider other applications and metrics that could arise from these.
4. Analyze the impact on the perceived BER when using training sequences that are tailored to the application
5. Analyze the cases in which the Confidence Ellipsoid based approximation outperforms the Markov Bound approximation.

Part V

# APPENDIX

# A

## MPP: ANALYTICAL SOLUTION: PROOF

The cost function, i.e. the average metric can be expressed as in (118).

$$\mathrm{tr}\left\{(\boldsymbol{\mathcal{I}}_T^T \otimes \boldsymbol{\mathcal{I}}_R)(\mathbf{R}_T^{-T} \otimes \mathbf{R}_R + (\mathbf{P}^H \otimes \mathbf{I})(\mathbf{S}_Q^{-T} \otimes \mathbf{S}_R^{-1})(\mathbf{P} \otimes \mathbf{I}))\right\} \quad (118)$$

where using that $\boldsymbol{\mathcal{I}}_T = \mathbf{I}$, $\mathbf{S}_R = \mathbf{R}_R$ and the properties in (87)-(89), can be expressed as in (119).

$$\mathrm{tr}\left\{(\mathbf{R}_T^{-T} + \mathbf{P}^T\mathbf{S}_Q^{-T}\mathbf{P}^*) \otimes \boldsymbol{\mathcal{I}}_R\mathbf{R}_R^{-1}\right\} \quad (119)$$

Using that $\mathrm{tr}(\mathbf{A} \otimes \mathbf{B}) = \mathrm{tr}(\mathbf{A})\mathrm{tr}(\mathbf{B})$, the trace of the Kronecker product in (119) can be expressed as product of traces, where the second factor does not depend on $\mathbf{P}$ and is a constant, i.e. has no relevance in the cost function. This entails that the optimal training problem can be solved by using the cost function (120).

$$\mathrm{tr}(\mathbf{R}_T^{-T} + \mathbf{P}^T\mathbf{S}_Q^{-T}\mathbf{P}^*) \quad (120)$$

Which is exactly the same problem that is solved in Theorem 2 in [9].

# B

## SDP FORMULATION: PROOF

If $\mathbf{Q}$ and $\mathbf{R}$ are hermitian, the condition:

$$\begin{bmatrix} \mathbf{Q} & \mathbf{S} \\ \mathbf{S}^H & \mathbf{R} \end{bmatrix} \geq 0 \tag{121}$$

is equivalent to the conditions:

$$\begin{aligned} \mathbf{R} &\geq 0 \\ \mathbf{Q} - \mathbf{S}\mathbf{R}^{\dagger}\mathbf{S}^H &\geq 0 \\ \mathbf{S}(\mathbf{I} - \mathbf{R}\mathbf{R}^{\dagger}) &= 0 \end{aligned} \tag{122}$$

as mentioned in [4].

The restriction in performance can be expressed as (123)

$$\mathrm{tr}(\boldsymbol{\mathcal{I}}_{adm}{}^{1/2}\mathbf{C}_{MMSE}\boldsymbol{\mathcal{I}}_{adm}{}^{H/2}) \leq \frac{\epsilon}{\gamma}, \tag{123}$$

which considering that the equation in the trace is a quadratic form and $\mathbf{C}_{MMSE}$ is positive semidefinite it is possible to say that $\boldsymbol{\mathcal{I}}_{adm}{}^{1/2}\mathbf{C}_{MMSE}\boldsymbol{\mathcal{I}}_{adm}{}^{H/2} \geq 0$. By adding an additional positive semidefinite hermitian matrix $\mathbf{M}$ it is possible to write with additional restrictions that will be added later, that

$$\mathbf{M} - \boldsymbol{\mathcal{I}}_{adm}{}^{1/2}\mathbf{C}_{MMSE}\boldsymbol{\mathcal{I}}_{adm}{}^{1/2} \geq 0 \tag{124}$$

which using the equivalence (121)-(122), forcing $\mathbf{M} \geq 0$ and considering that $\mathbf{C}_{MMSE}$ is invertible and semidefinite positive, can be expressed as in (125).

$$\begin{bmatrix} \mathbf{M} & \boldsymbol{\mathcal{I}}_{adm}{}^{1/2} \\ \boldsymbol{\mathcal{I}}_{adm}{}^{H/2} & \mathbf{C}_{MMSE}^{-1} \end{bmatrix} \geq 0 \tag{125}$$

It is necessary now to add a constraint that forces that the restriction (123) is fulfilled.

By obtaining the trace of the expression in equation (124) the expression in (126) is obtained.

$$\mathrm{tr}(\mathbf{M}) \geq \mathrm{tr}(\boldsymbol{\mathcal{I}}_{adm}\mathbf{C}_{MMSE}) \tag{126}$$

To assure that $\text{tr}(\boldsymbol{\mathcal{I}_{adm}}\mathbf{C}_{MMSE}) \leq \frac{\epsilon}{\gamma}$, the restriction $\text{tr}(\mathbf{M}) \leq \frac{\epsilon}{\gamma}$ is required. Grouping all the restrictions up to the point (127) is obtained.

$$\begin{aligned} \text{tr}(\mathbf{M}) &\leq \frac{\epsilon}{\gamma} \\ \begin{bmatrix} \mathbf{M} & \boldsymbol{\mathcal{I}_{adm}}^{1/2} \\ \boldsymbol{\mathcal{I}_{adm}}^{H/2} & \mathbf{C}_{MMSE}^{-1} \end{bmatrix} &\geq 0 \\ \mathbf{M} &\geq 0 \end{aligned} \tag{127}$$

Assuming that the covariance matrix $\mathbf{C_{MMSE}}$ is semidefinite-positive and invertible, the LMI in (127) is equivalent to (128).

$$\begin{aligned} \mathbf{M} &\succeq \boldsymbol{\mathcal{I}_{adm}}^{1/2}\mathbf{C_{MMSE}}\boldsymbol{\mathcal{I}_{adm}}^{H/2} \\ \text{tr}(\mathbf{M}) &\leq \frac{\epsilon}{\gamma} \\ \mathbf{M} &\succeq 0 \end{aligned} \tag{128}$$

Recall that $\mathbf{M}$ is an optimization variable and that $\mathbf{C_{MMSE}}$ directly depends on a second optimization variable. From (128) it follows that, any pair $\mathbf{M}$, $\mathbf{C_{MMSE}}$ that fulfill the restrictions fulfill (123). In the same way, for any $\mathbf{C_{MMSE}}$ that fulfills the restriction (123), it is possible to find an auxiliary matrix $\mathbf{M}$ that fulfills the restrictions in (128). As a matter of fact, a trivial way of expressing this matrix $\mathbf{M}$ would be $\mathbf{M} = \boldsymbol{\mathcal{I}_{adm}}^{1/2}\mathbf{C_{MMSE}}\boldsymbol{\mathcal{I}_{adm}}^{H/2}$.

# C

## ZF PRECODER, REAL METRIC: PROOF

Recall from equation (101) and (100), after just applying the product's distributive property, the metric can be expressed as (129).

$$J_{ZF}(\tilde{\mathbf{H}}, \mathbf{H}) = \mathbb{E}\left\{\mathbf{x}(t)^H[(\mathbf{H}\hat{\mathbf{H}}^\dagger)^H(\mathbf{H}\hat{\mathbf{H}}^\dagger) - (\mathbf{H}\hat{\mathbf{H}}^\dagger)^H - \mathbf{H}\hat{\mathbf{H}}^\dagger + \mathbf{I}]\mathbf{x}(t)\right\} \tag{129}$$

by using $\mathbf{H} = \hat{\mathbf{H}} - \tilde{\mathbf{H}}$ and simplifying the expression, the metric can be written as (130).

$$J_{ZF}(\hat{\mathbf{H}}, \tilde{\mathbf{H}}) = \mathbb{E}\left\{\mathbf{x}^H(t)(\tilde{\mathbf{H}}\hat{\mathbf{H}}^\dagger)^H(\tilde{\mathbf{H}}\hat{\mathbf{H}}^\dagger)\mathbf{x}(t)\right\} \tag{130}$$

It is important to notice that this expectation is computed within a block, and therefore, the only random variable that is affected by it is $\mathbf{x}(t)$. Additionally, the trace of the metric will be computed in order to take advantage of its rotatory property. Using this combined with the Kronecker product properties (131) and(132), mentioned in Appendix 1 in [10], the metric can be finally expressed as in (105).

$$\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA}) = \mathrm{vec}^H(\mathbf{A}^H)\mathrm{vec}(\mathbf{B}) \tag{131}$$

$$\mathrm{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\mathrm{vec}(\mathbf{B}) \tag{132}$$

# BIBLIOGRAPHY


[1] I. Barhumi, G. Leus, and M. Moonen. Optimal training design for MIMO OFDM systems in mobile wireless channels. *Signal Processing, IEEE Transactions on*, 51(6):1615–1624, 2003.

[2] M. Biguesh, S. Gazor, and M.H. Shariat. Optimal training sequence for MIMO wireless systems in colored environments. *Signal Processing, IEEE Transactions on*, 57(8):3144–3153, 2009.

[3] E. Björnson and B. Ottersten. A framework for training-based estimation in arbitrarily correlated rician MIMO channels with rician disturbance. *Signal Processing, IEEE Transactions on*, 58(3):1807–1820, 2010.

[4] Stephen P Boyd. *Linear matrix inequalities in system and control theory*, volume 15. Siam, 1994.

[5] Stephen Poythress Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[6] Tim Brown, Persefoni Kyritsi, and Elizabeth De Carvalho. *Practical Guide to MIMO Radio Channel: with MATLAB Examples*. Wiley. com, 2012.

[7] Dongning Guo, Shlomo Shamai, and Sergio Verdú. Mutual information and minimum mean-square error in gaussian channels. *Information Theory, IEEE Transactions on*, 51(4):1261–1282, 2005.

[8] B. Hassibi and B.M. Hochwald. How much training is needed in multiple-antenna wireless links? *Information Theory, IEEE Transactions on*, 49(4):951–963, 2003.

[9] Dimitrios Katselis, Eleftherios Kofidis, and Sergios Theodoridis. Training-based estimation of correlated MIMO fading channels in the presence of colored interference. *Signal Processing*, 87(9):2177–2187, 2007.

[10] Dimitrios Katselis, Cristian Rojas, Mats Bengtsson, Emil Björnson, Xavier Bombois, Nafiseh Shariati, Magnus Jansson, and Håkan Hjalmarsson. Training sequence design for mimo channels: an application-oriented approach. *EURASIP Journal on Wireless Communications and Networking*, 2013(1):245, 2013.

[11] Dimitrios Katselis, Cristian Rojas, Håkan Hjalmarsson, Mats Bengtsson, et al. Application-oriented finite sample experiment design: A semidefinite relaxation approach. In *System Identification*, volume 16, pages 1635–1640, 2012.

[12] Jayesh H. Kotecha and A.M. Sayeed. Transmit signal design for optimal estimation of correlated MIMO channels. *Signal Processing, IEEE Transactions on*, 52(2):546–557, 2004.

[13] Yong Liu, T.F. Wong, and William W. Hager. Training signal design for estimation of correlated MIMO channels with colored interference. *Signal Processing, IEEE Transactions on*, 55(4):1486–1497, 2007.

[14] Zhi-Quan Luo, Wing-Kin Ma, A.M.-C. So, Yinyu Ye, and Shuzhong Zhang. Semidefinite relaxation of quadratic optimization problems. *Signal Processing Magazine, IEEE*, 27(3):20–34, 2010.

[15] H. Minn and N. Al-Dhahir. Optimal training signals for MIMO OFDM channel estimation. *Wireless Communications, IEEE Transactions on*, 5(5):1158–1168, 2006.

[16] Andreas F Molisch. *Wireless communications*, volume 15. Wiley. com, 2010.

[17] Sailes K Sengijpta. *Fundamentals of Statistical Signal Processing: Estimation Theory*, volume 37. Taylor & Francis, 1995.