Informationstechnik

Tharwat Morsy

Convex Optimization for Detection in Structured Communication Problems

2012

Convex Optimization for Detection in Structured Communication Problems

Von der Fakultät für Elektrotechnik und Informationstechnik
der Technischen Universität Dortmund
genehmigte

Dissertation

zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
eingereicht von

Tharwat Elsayed Hamed Morsy

Tag der mündlichen Prüfung:	08.11.2012
Hauptreferent:	Univ.-Prof. Dr.-Ing. J. Götze
Korreferent:	Univ.-Prof. Dr.rer.nat. C. Wöhler

Arbeitsgebiet Datentechnik, Technische Universität Dortmund

Abstract

The receiver in a wireless communication system has the task of computing good estimates for the data symbols that have been transmitted. The best (optimum) detector is the Maximum Likelihood (ML) detector. However, it requires a high computational complexity. This work aims to efficiently detect the transmitted symbols with a reduced complexity.

In order to produce a near optimum receiver, two methods are presented. These methods are obtained by convex optimization relaxations which yield global optimum solutions. The relaxations are combined with the idea of using the structure of the channel matrix to reduce the computational complexity. The channel matrix exhibits a banded Toeplitz structure.

In each case, the dual problem of the convex optimization relaxation is solved to estimate the noise power. Gradient descent algorithm is used to solve the dual problem in the first relaxation while the bisection method is applied for the second relaxation. In both cases, the result is a Generalized Minimum Mean Squared Error (GMMSE) detector which has a form similar to the Minimum Mean Square Error (MMSE) detector and a performance almost the same as the MMSE detector, but it is not require the knowledge of the noise power. The GMMSE detectors can be used in scenarios where adapted or blind adaptive detection is not suitable, for instance when the channel is rapidly changing. Using a circular approximation of banded Toeplitz matrix the Fast Fourier Transform (FFT) can be applied to reduce the computational complexity of the detectors.

Finally, the local search method is applied to enhance the performance of the proposed GMMSE detector. The proposed detector is a near optimum detector with low computational complexity.

Acknowledgements

First of all, I thank God for giving me the strength and the ability to continue my graduate studies and finish this dissertation.
I want to express my deep thanks and appreciation to my country Egypt for the financial support in the first two years of my PhD study.
I would like to thank Professor Jürgen Götze, the head of the laboratory, for all the interesting discussions that contributed essentially to this thesis and for creating an open and relaxed atmosphere.
Furthermore, I am very pleased to thank Professor Hamed Nassar (Faculty of Computers and Informatics, Suez Canal University-Egypt) just as much, for introducing me to the topic and for his guidance during my study. I would also like to thank Professor Christian Wöhler (Image Analysis Group, TU Dortmund) for his comments on my results and his review for my thesis.
Many thanks to my present and former colleagues especially Jan Geldmacher, it was a pleasure to share so much time with all of you.
Certainly, this work never be done without support that I had from my beloved wife Heba and my kids Mirna, Mohamed and Ahmed. Heba...I appreciate so much your love and support. You have always believed in my ability and my dreams, and given me wings to make them possible. I thank God for giving me such a wonderful wife. I Love You!
To my parents, Elsayed Hamed and Amaal Elshobrawy, my brothers and my sisters....thank you so much for all of the support you have given me throughout the years. I love all of you!
To my father in law, Nagy Elnemr, thank you so much for giving me such a wonderful woman as a wife. I know if you are still alive, you will be so happy.
My brother Ahmed Elsayed Ibrahim...my best friend, I would like to thank you for supporting me throughout my life.

Dortmund, November 2012

Contents

List of Figures

It is better to sit alone than in company with the bad, and it is better still to sit with the good than alone. It is better to speak to a seeker of knowledge than to remain silent, but silence is better than idle words.!

Prophet Mohamed

1 Introduction

1.1 Overview

The problem of designing the receivers in wireless communication systems is the data detection from noisy measurements of the transmitted signals. The receiver makes occasional errors due to distortions and noise. Therefore, designing a receiver that has minimal probability of error is appealing.

Unfortunately, these kinds of designs produce the computational complex receivers. Therefore, they are often abandoned in favor of computationally more efficient, but suboptimal receivers. There is a substantial gap in performance between suboptimal and optimal receivers. Only this problem makes the optimal receivers interesting. Convex optimization is one technique that tries to close the gap.

Throughout history, people have faced optimization problems and made great efforts to solve them. In general, optimization is the process that finds the best way to use available resources, while at the same time it is not violating any of the proposed constraints. Optimization can be a problem of searching the product with minimum cost or a problem of finding the nearest school from your apartment that is suitable for your children. These kinds of problems can be solved by the intuition and they are not hard to find the optimum solution. All previous problems are considered as simple optimization problems. Unfortunately, there are many important optimization problems that can not easily solved by the intuition. Some problems such as, how to find the optimal route for an airplane to minimize the fuel consumption or how to allocate the production of a product to different machines (with different capacities, startup cost and operating cost) to meet the production target at minimum cost. In these cases, it is impossible to solve this problem by intuition. Instead, the problem must be modeled mathematically [41, 44], then the problem is solved based on the model using one of the efficient optimization algorithms [114].

Elements of an Optimization Model

Any mathematical optimization model has three main elements [77]:

- *Decision variable: x.*
- *Objective function:* $f(x)$ to be minimized or maximized over the variable. It can be thought as, what do you want? That is, what is your objective.
- *Constraints*: two types of constraints where both, one or none of them can exist:
    - *Equality constraints: $h_i(x) = 0, i = 1, ..., m$.*
    - *Inequality constraints: $g_i(x) \leq 0, i = 1, ..., p$.*

When the objective function is minimized, the problem is called a minimization optimization problem and a maximization problem is the problem of maximizing the objective function. Equality constraint problem is the problem that only has equality constraints and the inequality constraint problem has only inequality constraints. An unconstrained optimization problem is the problem without constraints.

As we had classified the problem with respect to the kind of the constraints, we also present two classifications of optimization problems that depend on the variable x. When x is a scalar, i.e., $x \in \mathbb{R}$ or $x \in \mathbb{C}$, the problem is said to be a *one-dimensional optimization problem*, but if the variable is a vector $\boldsymbol{x}$, the problem is called a *multi-dimensional optimization problem* (We will consider these kinds of problems).

The general mathematical optimization problems could be very hard to solve, especially when the number n of decision variables, collected in the vector $\boldsymbol{x} = (\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, ...\boldsymbol{x}^{(n)})^T$ is large [88]. The reasons of these difficulties are:

- The problem "terrain" may be riddled with local optima.

- It might be very hard to find a feasible point (i.e., an $\boldsymbol{x}$ which satisfies all the constraints), in fact, the feasible set could be empty.

- Stopping criterion used in general optimization algorithms are often arbitrary.

- Optimization algorithms may have poor convergence rates.

- Numerical problems could cause the optimization algorithm to stop all together or wander.

Convex Optimization Problem

The convex optimization problem [96] is obtained if the objective function $f(x)$ and the inequality constraints $g_i(x)$ are convex. In addition the equality constraints $h_i(x)$ must be affine. In case of the convex optimization problem, the first three problems stated above disappear which means that:

- The solution is a global optimum solution.

- Feasibility of convex optimization problems can be determined unambiguously, at least in principle.

- Very precise stopping criteria are available using *duality.*

There are many various applications of convex optimization techniques in signal processing and communication. Successful examples of this kind include detection and estimation [3, 26, 54, 92], channel equalization [24, 49, 68], circuit design [2, 19, 20, 95, 108], digital beamforming [53, 103] and communication system design [59, 104, 110].

Convex optimization is used in this thesis for detection problems in wireless communication . Because the *Maximum Likelihood* (ML) problem has a high complexity [107, 109], sub-optimum detectors such as Least Squares (LS) and Minimum Mean Square Error (MMSE) are employed to design receivers for wireless communication systems [111]. The problem of these methods is that the bit error rate (BER) performance is poor compared to that of ML detector. Also, Generalized Minimum Mean Squared Error (GMMSE) detector has a performance which is almost the same as that of MMSE, but it does not require the knowledge of noise power σ_n^2 [112]. So it can be used in scenarios where adapted or blind adaptive detection is not suitable, for instance when the channel is changing rapidly, and the ambient noise power is unknown. Although the computational complexity of LS, MMSE and GMMSE is much lower than ML, they are still computationally demanding in virtue of their realization in communication systems. Furthermore, their performance is lower than ML detector.

In this thesis, a near optimum ML receiver is considered by using *convex optimization*. Concerning the efficiency of using convex optimization in wireless communication problems, the reader is referred to the common and well-known references [57, 60] for more details. Two convex relaxations are proposed to solve the ML problem. Both relaxations are solved using the dual optimization problem and the resulting detectors have almost the same performance as MMSE detector, but the computational complexity is still high. To achieve this results, we relax the ML detection problem which is a discrete problem into a continuous (convex) problem. Then, we solve the dual problem (which is a single valued problem) of the relaxation problem. We take this solution as an estimation of the noise power σ_n^2. In the first relaxation, we use the gradient descent algorithm to solve the extracted dual problem while the second relaxation uses the bisection method to solve the dual problem.

Structured Convex Optimization Problem

One of the most efficient ways to reduce the computational complexity is finding a way or a method to make use of the matrix structures [29], which occur in the channel convolution matrix and the respective detection problem. We use the Toeplitz structure [4] of the channel matrix which enables us to use Eigenvalue Decomposition (EVD) to solve the dual problem. The gradient descent algorithm or the bisection method in this case runs over diagonal matrices and the multiplications operations are reduced. This approach has reduced the overall computational complexity of the detection problem, but the complexity of computing EVD still requires considerable efforts [16]. These efforts are further reduced using the approximation of the Toeplitz structure by a circular structure [22]. Then, we apply Fast Fourier Transform (FFT) method which is a well known computational tool [13, 78]. The EVD of a circular matrix can be obtained by using FFT, such that the computational complexity is reduced from $O(n^2)$ to $O(n\, log\, n)$. The performance analysis of the proposed solution is also given [74]. The resulting detector is a reduced complexity form of GMMSE detector [70, 72, 73].

Near Optimum Receiver

The GMMSE detector has a reduced complexity, but it does not provide the performance of the ML detector. We enhance its performance using the *local search algorithm* to find the global solution of the problem. Local search moves in the domain of solution from local optimum to another until it finds the global optimum for the detection problem. Then, the performance of the presented detector is near to that of ML detector [71].

1.2 Outline

This section outlines the chapters of this thesis.

Chapter 2 begins with the basic definitions in the field of convex optimization such as *convex set*, *convex function* and *convex optimization problem.* The purpose of this introduction is to show how we relax some constellation sets such as Binary Phase Shift Keying (BPSK) and Quadrature Phase Shift Keying (QPSK) to convex sets and to relax the detection problem to a convex optimization problem. This chapter also covers Lagrangian duality, which plays a central role in convex optimization. Lagrangian duality converts the constraint primal detection problem into unconstrained dual problem. The classical *Karush-Kuhn-Tucker* optimality conditions are given. Interior point methods are briefly discussed in this chapter which can efficiently solve this dual problem.

Chapter 3 discusses two main problems in the field of signal processing and wireless communications. The first problem is the detection problem or the problem of deciding which signal from multiple possible signals was transmitted. Bayes criterion is presented in both binary hypothesis and M hypotheses. The second problem is the estimation problem. Parameter space, observation space, probabilistic mapping from parameter space to observation space and estimation rule concepts are presented. Random parameter estimation such as Bayes estimation, MMSE estimation and Maximum a Posteriori (MAP) estimation are discussed. Nonrandom Parameter estimation such as ML is presented. Digital modulation schemes such as BPSK and QPSK are discussed in this chapter. The description of the system of transmitted bits through a simple channel or a fading channel with the presence of an Additive White Gaussian Noise (AWGN) is presented. Some sub-optimum receivers are given such as LS, MMSE, GMMSE, and semidefinite relaxation.

Chapter 4 focuses on the convex relaxation of MPSK (especially BPSK and QPSK) constellation sets [23], so the ML problem is relaxed to a constrained convex optimization problem. The resulting detector is GMMSE detector with a performance that is almost the same as MMSE detector, but it does not require the knowledge of noise power σ_n^2. This relaxation is combined with the idea of using the structure of the channel matrix (banded Toeplitz) to reduce the computational complexity. To achieve that, the EVD of the channel matrix is used. Circular approximation of the Toeplitz structure is used to further reduce the complexity by using FFT decomposition of the channel matrix. Gradient descent algorithm as one of the interior point methods is applied to solve the dual problem of the relaxed convex problem. The solution of this problem is taken as an estimation of the noise power, then it is substituted in the MMSE solution.

Chapter 5 presents hidden convexity relaxation to solve ML problem. The bisection method is applied to solve the dual problem for this relaxation. Bisection method is one of the bracketing methods that has two basic properties. It is always convergent and its error can be controlled. The solution of bisection method is again the noise power estimation. As in chapter 4, combined with the banded Toeplitz structured of the channel matrix a GMMSE solution is given. Then, approximating this structure by circular structure, the reduced form of GMMSE detector is again produced. The simulation results for QPSK constellation is presented. In this simulation, BER performance is compared for LS, MMSE, and the presented GMMSE

Chapter 6 presents a detector that has the form of the GMMSE detector with complexity nearly the same as GMMSE detector with bit error rate (BER) performance is enhanced using the local search algorithm. The overall computational complexity of the proposed detectors is presented. This complexity includes the complexity of finding the solution and the complexity of all three used algorithms, gradient descent, bisection method and local search.

Chapter 7 provides a summary of this thesis and gives a discussion on topics that could be relevant for future work.

2 Convex Optimization

Convex optimization problem is a special class of the mathematical optimization problems, which has been developed for about a century. Interior point methods were developed in the 1980s in order to efficiently solve these kinds of problems. These solution methods are reliable enough to be embedded in a computer-aided design or analysis tool. After these developments, researchers in many different fields such that automatic control systems, electronic circuit design, data analysis and modeling, statistics, finance, estimation and signal processing, and communications and network are interested to solve their problems using this powerful mathematical optimization tool. A very important advantage of using convex optimization techniques is that we can efficiently find the global solution for the problem.

The following sections introduce the basic concepts which cover the elements of convex optimization problems. Much of the material in this chapter is heavily based on the book, '*Convex Optimization*' [11].

2.1 Convex Set

In this section, we begin our discussion of convex optimization with some vital notations in this field. The most important notation is *convex set*.

Affine Set: If the line through any two points in a set $S \subseteq \mathbb{R}^n$ lies in S, i.e., for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$ and $\lambda \in \mathbb{R}$, we have

$$\lambda \boldsymbol{x}_1 + (1 - \lambda)\boldsymbol{x}_2 \in S,$$

then the set S is said to be an *affine set*. This definition can be stated as, the linear combination of any two points in S that is contained in

S, provided the coefficients in the linear combination sum to one. In general (more than two points), an affine set contains every affine combination of its points, i.e., if S is an affine set, $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_k \in S$, and $\lambda_1+\lambda_2+...+\lambda_k = 1$, then the point $\lambda_1\boldsymbol{x}_1+\lambda_2\boldsymbol{x}_2+...+\lambda_k\boldsymbol{x}_k$ also belongs to S.

Geometrically, an affine set is simply considered as a subspace that is not necessarily centered at the origin.

Example: The solution set of the system of the linear equations, $S = \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$ where, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{b} \in \mathbb{R}^m$ is an example of affine sets. To prove that suppose $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$, i.e., $\boldsymbol{A}\boldsymbol{x}_1 = \boldsymbol{b}, \boldsymbol{A}\boldsymbol{x}_2 = \boldsymbol{b}$. Then for any $\lambda \in \mathbb{R}$, we have

$$\boldsymbol{A}(\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2) = \lambda\boldsymbol{A}\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{A}\boldsymbol{x}_2 = \lambda\boldsymbol{b} + (1-\lambda)\boldsymbol{b} = \boldsymbol{b},$$

which shows that the affine combination $\lambda\boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2$ is also in S.

Affine Function: A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *affine* if it has the form linear plus constant $f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$. If $\boldsymbol{F}$ is a matrix valued function, i.e., $\boldsymbol{F} : \mathbb{R}^n \to \mathbb{R}^{p\times q}$ then, $\boldsymbol{F}$ is affine if it has the form $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{A}_0 + \boldsymbol{x}^{(1)}\boldsymbol{A}_1 + ... + \boldsymbol{x}^{(n)}\boldsymbol{A}_n, \boldsymbol{A}_i \in \mathbb{R}^{p\times q},\ i = 0, 1, ..., n.$

Convex Set: If the line segment between any two points in S lies completely in S, i.e., if for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$ and any λ with $0 \le \lambda \le 1$, we have

$$\lambda\boldsymbol{x}_1 + (1-\lambda)\,\boldsymbol{x}_2 \in S,$$

then the set S is said to be *convex set.* The square which includes its boundary in Figure 2.1a is convex, the set in Figure 2.1b is not convex

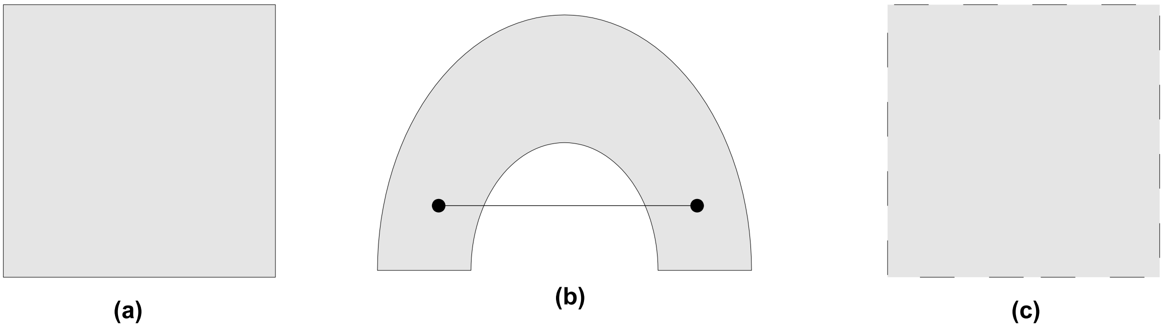


Figure 2.1: Convex and non-convex sets.

since the line segment between the two black points in the set is not contained in the set and in Figure 2.1c, the square that contains some boundary points, but not others is not convex. Any point of the form $\lambda_1\boldsymbol{x}_1+\lambda_2\boldsymbol{x}_2+...+\lambda_k\boldsymbol{x}_k$, where $\lambda_1+\lambda_2+...\lambda_k = 1$ and $\lambda_i \geq 0, i = 1, 2, ..., k$ is a convex combination of the points $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_k$. Generally, it can be shown that a set is convex if and only if it contains every convex combination of its points. Some important convex sets are, Euclidean ball, norm ball, polyhedral, and the positive semi-definite cone.
We note that the intersection of two convex sets is a convex set.

Convex Cone: If a set $S \subseteq \mathbb{R}^n$ contains all rays passing through its points which emanate from the origin, as well as all line segments joining any points on those rays, i.e.,

$$\boldsymbol{x}, \boldsymbol{y} \in S, \lambda, \mu \geq 0, \Rightarrow \lambda\boldsymbol{x} + \mu\boldsymbol{y} \in S,$$

where, $\mu \in \mathbb{R}$ then this set is said to be *convex cone.*

Geometrically, $\boldsymbol{x}, \boldsymbol{y} \in S$ means that S contains the entire pie slice between $\boldsymbol{x}$ and $\boldsymbol{y}$ as shown in Figure 2.2. The set $\mathbb{S}_+^n = \{\boldsymbol{X} \in \mathbb{S}^n | \boldsymbol{X} \succeq 0\}$ of symmetric positive semidefinite (PSD) matrices is an example of a convex cone [39].

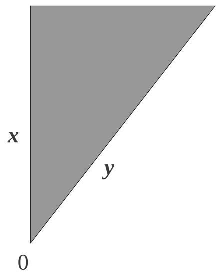


Figure 2.2: Convex cone.

2.2 Convex Function

In this section, the definition of the convex function and some important examples and techniques are introduced to the reader for verifying convexity.
In mathematics, a function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ is *convex* if it is defined on a convex domain (convex set) and if for all $\boldsymbol{x}_1$,$\boldsymbol{x}_2 \in \ dom f$ (where $dom f$ is the domain of the function f), and λ with $0 \leq \lambda \leq 1$, we have

$$f\left(\lambda \boldsymbol{x}_1 + (1-\lambda)\,\boldsymbol{x}_2\right) \leq \lambda f\left(\boldsymbol{x}_1\right) + (1-\lambda)\, f\left(\boldsymbol{x}_2\right).$$

Geometrically, this definition can described as the line segment between $(\boldsymbol{x}_1, f(\boldsymbol{x}_1))$ and $(\boldsymbol{x}_2, f(\boldsymbol{x}_2))$ lies above the graph of f in Figure 2.3.
f is *concave* if $-f$ is convex where, the line segment between the mentioned two points lies below the graph of f as shown in Figure 2.4.

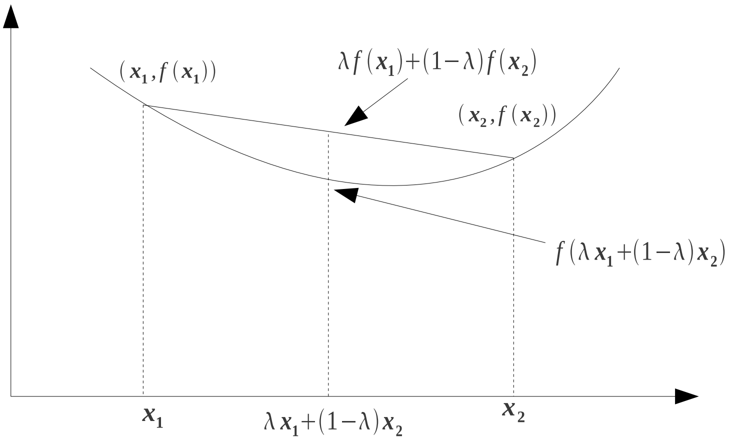

$(x_1, f(x_1))$
$\lambda f(x_1)+(1-\lambda)f(x_2)$
$(x_2, f(x_2))$
$f(\lambda x_1+(1-\lambda)x_2)$
x_1
$\lambda x_1+(1-\lambda)x_2$
x_2

Figure 2.3: Graph of the convex function.

2.2.1 First-Order Conditions

Let f be differentiable (i.e., its gradient $\boldsymbol{\nabla} f$ exists at each point in the domain of f, which is open). Then f is convex if and only if its domain is convex and

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{\nabla} f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x})\,, \tag{2.1}$$

holds for all $\boldsymbol{x}, \boldsymbol{y} \in dom f$. Figure 2.5 illustrates the inequality (2.1). $f(\boldsymbol{x}) + \boldsymbol{\nabla} f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x})$ is the first-order Taylor approximation of f

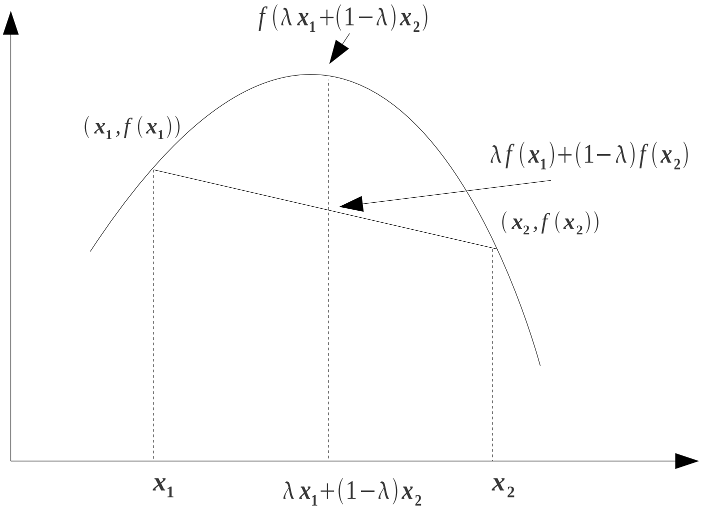

$f(\lambda x_1+(1-\lambda)x_2)$
$(x_1,f(x_1))$
$\lambda f(x_1)+(1-\lambda)f(x_2)$
$(x_2,f(x_2))$
x_1
$\lambda x_1+(1-\lambda)x_2$
x_2

Figure 2.4: Graph of the concave function.

near $\boldsymbol{x}$. The inequality (2.1) states that for a convex function, the first-order Taylor approximation is a *global under-estimator* of the function. At the other hand, if the first-order Taylor approximation of a function is a global under-estimator of the function, then the function is convex. The most important property of convex functions is that we can derive *global information* from *local information* about a convex function. As an example, the inequality (2.1) shows that if $\boldsymbol{\nabla} f(\boldsymbol{x}) = 0$, then for all $\boldsymbol{y} \in dom f$, $f(\boldsymbol{y}) \geq f(\boldsymbol{x})$, i.e., $\boldsymbol{x}$ is a *global minimizer* of the function f. The proof of first-order conditions can be found in [6].

2.2.2 Second-Order Conditions

In case of a twice differentiable function f, the Hessian $\boldsymbol{\nabla}^2 f$ of f exists at each point in the domain of f (which is open). Then, f is convex if and only if $dom f$ is convex and its Hessian is positive semi-definite, i.e., for all $\boldsymbol{x} \in dom f$,

$$\boldsymbol{\nabla}^2 f(\boldsymbol{x}) \geq 0. \tag{2.2}$$

For a defined function f on $\mathbb{R}$, this condition becomes very simple, i.e., $f''(x) \geq 0$ and $dom f$ is convex. The following is an example of the second-order conditions.

Quadratic Function: The function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, with $dom f = \mathbb{R}^n$, is

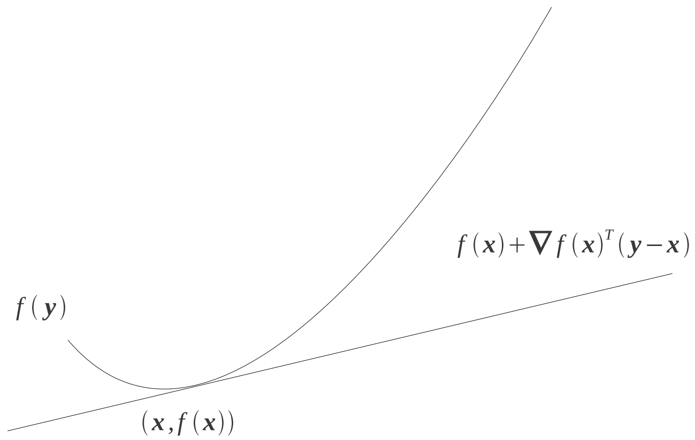

$f(\boldsymbol{x})+\nabla f(\boldsymbol{x})^T(\boldsymbol{y}-\boldsymbol{x})$
$f(\boldsymbol{y})$
$(\boldsymbol{x},f(\boldsymbol{x}))$

Figure 2.5: First-order conditions for convexity.

given by

$$f(\boldsymbol{x}) = (1/2)\,\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} + r,$$

with $\boldsymbol{P} \in \mathbb{S}^n, \boldsymbol{q} \in \mathbb{R}^n$, and $r \in \mathbb{R}$. Applying the condition (2.2) to this function, we find that $\nabla^2 f(x) = \boldsymbol{P}$ for all $\boldsymbol{x}$, f is convex if and only if $\boldsymbol{P} \succeq 0$ which is satisfied from the restriction $\boldsymbol{P} \in \mathbb{S}^n$.

2.3 Convex Optimization Problems

2.3.1 Mathematical Optimization

Consider the problem

$$\begin{array}{ll} \textit{minimize} & f_0(\boldsymbol{x}) \\ \textit{subject to} & f_i(\boldsymbol{x}) \leq 0, \quad i = 1, ..., m \\ & h_i(\boldsymbol{x}) = 0, \quad i = 1, ..., p. \end{array} \tag{2.3}$$

Problem (2.3) is the general form of the mathematical optimization problem. The description of the problem is that, how to find an $\boldsymbol{x}$ that minimizes $f_0(\boldsymbol{x})$ among all vectors that satisfy the conditions

$f_i(\boldsymbol{x}) \leq 0, i = 1, ..., m$, and $h_i(\boldsymbol{x}) = 0, i = 1, ..., p$. Here we have basic elements of the mathematical optimization problem. The first element is called the *optimization variable*, $\boldsymbol{x} \in \mathbb{R}^n$, while the second element is called the *objective function* or the *cost function*, $f_0 : \mathbb{R}^n \longrightarrow \mathbb{R}$. The third element is called *inequality constraints*, $f_i(\boldsymbol{x}) \leq 0$. The last element of a mathematical optimization problem is called *equality constraints* $h_i(\boldsymbol{x}) = 0$. If there are no constraints (inequality and equality constraints) the problem (2.3) is called *unconstrained problem.* Now we present some important definitions in the field of mathematical optimization problems.

Optimization Problem's Domain: The domain of any optimization problem is the set of points for which the objective and all constraint functions are defined,

$$D = \bigcap_{i=0}^{m} dom f_i \cap \bigcap_{i=1}^{p} dom h_i.$$

Feasible Point: A point $\boldsymbol{x} \in D$ is *feasible* if it satisfies the constraints $f_i(\boldsymbol{x}) \leq 0, i = 1, ..., m$, and $h_i(\boldsymbol{x}) = 0, i = 1, ..., p$. The set of all feasible points is called the *feasible set.*

Optimal Value: The *optimal value* of the problem (2.3) is defined as

$$p^* = inf \left\{ f_0(\boldsymbol{x}) \,|\, f_i(\boldsymbol{x}) \leq 0, i = 1, ..., m, h_i(\boldsymbol{x}) = 0, i = 1, ..., p \right\}.$$

Optimal Point: $\boldsymbol{x}^*$ is said to be an *optimal point*, or solves the problem (2.3), if it is feasible and $f_0(\boldsymbol{x}^*) = p^*$.

Feasibility Problem: It is a problem of the form,

$$\begin{array}{rll} find & \boldsymbol{x} & \\ subject\ \ to & f_i(\boldsymbol{x}) \leq 0, & i = 1, ..., m \\ & h_i(\boldsymbol{x}) = 0, & i = 1, ..., p. \end{array}$$

The *feasibility problem* is thus to determine whether the constraints are consistent, and if so, find a point that satisfies them.

Local Optimum and Global Optimum: A local optimum of an optimization problem is the optimum solution within a neighboring set of solution, while a global optimum is the optimum solution among all possible solutions. Figure 2.6 describes these two terms

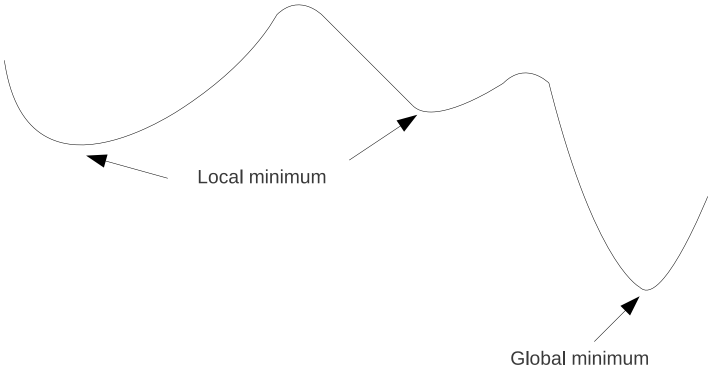


Figure 2.6: Local optima and global optimum.

Maximization Problem: The Maximization problem

$$\begin{array}{ll} \textit{maximize} & f_0\left(\boldsymbol{x}\right) \\ \textit{subject to} & f_i\left(\boldsymbol{x}\right) \leq 0, \quad i = 1, ..., m \\ & h_i\left(\boldsymbol{x}\right) = 0, \quad i = 1, ..., p, \end{array}$$

can be easily solved by minimizing the function $-f_0$.

Problem (2.3) is in general very hard to solve, especially when the number of decision variables in $\boldsymbol{x}$ is large (this is the case in most applications). There are several reasons for this difficulty:

1. The problem can be riddled with local optimum.

2. In many cases, it is very hard to find a feasible point, in fact the feasible set, which needs not even be fully connected, could be empty.

3. Stopping criteria that used in general optimization algorithms are often arbitrary.

4. Optimization algorithms often have very poor convergence rates.

5. Numerical problems may cause the minimization algorithm to stop all together or wander.

2.3.2 Convex Optimization Problems

Convex optimization is the technique that is used to avoid the difficulties that are presented at the end of the Subsection 2.3.1.
Convex optimization problems have three important properties that make these kinds of problems fundamentally more tractable than non-convex optimization problems:

1. Each local optimum is necessarily a global optimum.
2. Exact in-feasibility detection that is achieved by using duality theory (we will present it later in this chapter), hence algorithms are easy to initialize.
3. Efficient numerical solution methods that can handle very large problems.

A *convex optimization problem* is a mathematical problem that takes the form,

$$\begin{array}{ll} \textit{minimize} & f_0\left(\boldsymbol{x}\right) \\ \textit{subject to} & f_i\left(\boldsymbol{x}\right) \leq 0, \quad i = 1, ..., m \\ & \boldsymbol{a}_i^T \boldsymbol{x} = \boldsymbol{b}^{(i)}, \quad i = 1, ..., p, \end{array} \tag{2.4}$$

where $f_0, ..., f_m$ are convex functions. Looking at problem (2.4) and comparing it with problem (2.3), we find that, a convex optimization problem has three additional requirements:

- The convexity of the objective function.
- The convexity of the inequality constraint functions.
- The affine property of equality constraint functions

$$h_i(\boldsymbol{x}) = \boldsymbol{a}_i^T \boldsymbol{x} - \boldsymbol{b}^{(i)} = 0.$$

It is well known that if f_i are convex, and h_i are affine, then the first three problems of the mathematical optimization problem which are mentioned in the Subsection 2.3.1 disappear. In the following, we present the proof of each local minimum in a convex optimization problem is global (which is the most important property of convex optimization as shown in Figure 2.7).

Theorem 2.1 *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function and $S \subseteq \mathbb{R}^n$ be a convex set. Given a point $\boldsymbol{x}^* \in S$, suppose that there is a ball $B(\boldsymbol{x}^*, \epsilon) \subset S$ such that for all $\boldsymbol{x} \in B(\boldsymbol{x}^*, \epsilon)$ we have $f(\boldsymbol{x}) \geqslant f(\boldsymbol{x}^*)$. Then $f(\boldsymbol{x}^*) \leqslant f(\boldsymbol{x})$ for all $\boldsymbol{x} \in S$.*

Proof: Let $\boldsymbol{x} \in S$. Since f is convex over S, for all $\lambda \in [0, 1]$ we have $f(\lambda \boldsymbol{x}^* + (1 - \lambda)\boldsymbol{x}) \leq \lambda f(\boldsymbol{x}^*) + (1 - \lambda) f(\boldsymbol{x})$. Notice that there exists $\overline{\lambda} \in (0, 1)$ such that $\overline{\lambda}\boldsymbol{x}^* + (1 - \overline{\lambda})\boldsymbol{x} = \overline{\boldsymbol{x}} \in B(\boldsymbol{x}^*, \epsilon)$. By the convexity of f we have $f(\overline{\boldsymbol{x}}) \leqslant \overline{\lambda} f(\boldsymbol{x}^*) + (1 - \overline{\lambda}) f(\boldsymbol{x})$. After rearrangement we get,

$$f(\boldsymbol{x}) \geqslant \frac{f(\overline{\boldsymbol{x}}) - \overline{\lambda} f(\boldsymbol{x}^*)}{1 - \overline{\lambda}}.$$

Since $\overline{\boldsymbol{x}} \in B(\boldsymbol{x}^*, \epsilon)$, we have $f(\overline{\boldsymbol{x}}) \geqslant f(\boldsymbol{x}^*)$, thus

$$f(\boldsymbol{x}) \geqslant \frac{f(\boldsymbol{x}^*) - \overline{\lambda} f(\boldsymbol{x}^*)}{1 - \lambda} = f(\boldsymbol{x}^*),$$

as required.

From our remark at the end of Section 2.1, the intersection of convex sets is a convex set, we can say that the feasible set of a convex optimization problem is a convex set.

2.4 Duality Problem

Assume that the domain D of the optimization problem (2.3) is nonempty. The variable $\boldsymbol{x} \in \mathbb{R}^n$ in this problem has n components, so when n is large we have a large optimization problem. To solve this problem, we must find n decisions. Duality problem is an easy technique that reduces the number of decisions. We denote the optimal value of the problem (2.3) by p^* and we do not assume that the problem is convex.

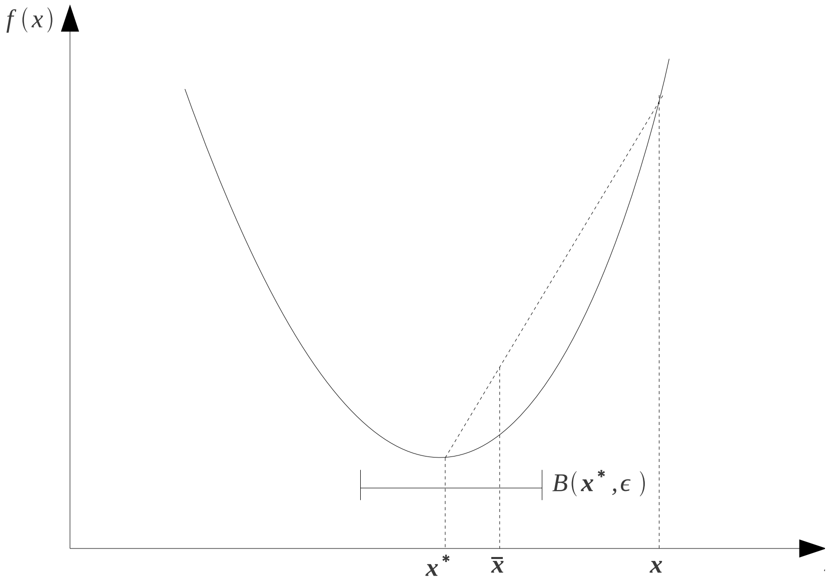


Figure 2.7: The fact that $f(\boldsymbol{x}^*)$ is the minimum in $B(\boldsymbol{x}^*, \epsilon)$ is enough to show that for any point $\boldsymbol{x} \in S, f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$.

2.4.1 The Lagrange Dual Function

Using Lagrangian duality converts the constrained problem such as (2.3) into unconstrained problem by augmenting the objective function with a weighted sum of the constraint functions.
Lagrangian: The *Lagrangian* $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ associated with the problem (2.3) is stated as,

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \boldsymbol{\lambda}^{(i)} f_i(\boldsymbol{x}) + \sum_{i=1}^{p} \boldsymbol{\nu}^{(i)} h_i(\boldsymbol{x}),$$

with *dom* $\mathcal{L} = D \times \mathbb{R}^m \times \mathbb{R}^p$. The parameter $\boldsymbol{\lambda^{(i)}}$ is known as the *Lagrange multiplier* associated with the ith inequality constraint $f_i(\boldsymbol{x}) \leqslant 0$ and $\boldsymbol{\nu}^{(i)}$ as the *Lagrange multiplier* associated with the ith equality constraint $h_i(\boldsymbol{x}) = 0$.
Dual Variable: The *dual variables* or *Lagrange multiplier vectors* associated with the problem (2.3) are the vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$.
Lagrange Dual Function: The *Lagrange dual function*

$g : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ is defined as the minimum value of the Lagrangian over $\boldsymbol{x}$ for $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^p$,

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\boldsymbol{x} \in D} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\boldsymbol{x} \in D} \left(f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \boldsymbol{\lambda}^{(i)} f_i(\boldsymbol{x}) + \sum_{i=1}^{p} \boldsymbol{\nu}^{(i)} h_i(\boldsymbol{x}) \right).$$

If the Lagrangian is unbounded below in $\boldsymbol{x}$, then the dual function tends to $-\infty$. The most important property of the dual function is that it is concave even when the primal problem (2.3) is not convex. That is because the dual function is the point-wise infimum of affine functions of $(\boldsymbol{\lambda}, \boldsymbol{\nu})$.

2.4.2 Lower Bounds on Optimal Value

We can look at the dual function as the lower bounds on the optimal value p^* of the problem (2.3), so for any $\boldsymbol{\lambda} \geqslant 0$ and $\boldsymbol{\nu}$ we get

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leqslant p^*. \tag{2.5}$$

We can easily verify this property. Let the problem (2.3) has $\tilde{\boldsymbol{x}}$ as a feasible point, i.e., $f_i(\tilde{\boldsymbol{x}}) \leqslant 0$ and $h_i(\tilde{\boldsymbol{x}}) = 0$, with $\boldsymbol{\lambda} \geqslant 0$. Then we have

$$\sum_{i=1}^{m} \boldsymbol{\lambda}^{(i)} f_i(\tilde{\boldsymbol{x}}) + \sum_{i=1}^{p} \boldsymbol{\nu}^{(i)} h_i(\tilde{\boldsymbol{x}}) \leqslant 0,$$

since each term in the first sum is non-positive, and each term in the second sum is zero,

$$\mathcal{L}(\tilde{\boldsymbol{x}}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\tilde{\boldsymbol{x}}) + \sum_{i=1}^{m} \boldsymbol{\lambda}^{(i)} f_i(\tilde{\boldsymbol{x}}) + \sum_{i=1}^{p} \boldsymbol{\nu}^{(i)} h_i(\tilde{\boldsymbol{x}}) \leqslant f_0(\tilde{\boldsymbol{x}}).$$

Then,

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\boldsymbol{x} \in D} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) \leqslant \mathcal{L}(\tilde{\boldsymbol{x}}, \boldsymbol{\lambda}, \boldsymbol{\nu}) \leqslant f_0(\tilde{\boldsymbol{x}}).$$

The inequality (2.5) is satisfied because of, $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leqslant f_0(\tilde{\boldsymbol{x}})$ holds for every feasible point $\tilde{\boldsymbol{x}}$.

When $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = -\infty$, the inequality (2.5) holds, but is vacuous. If $\boldsymbol{\lambda} \geqslant 0$ and $(\boldsymbol{\lambda}, \boldsymbol{\nu}) \in$ *dom* g, then we have a nontrivial lower bound on

p^* i.e., $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) > -\infty$.
A pair $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ with $\boldsymbol{\lambda} \geqslant 0$ and $(\boldsymbol{\lambda}, \boldsymbol{\nu}) \in$ *dom* g is referred as the *dual feasible.*
Example: *Linear Program (LP) in standard form*
Consider a *LP* in standard form

$$\begin{array}{ll} \textit{minimize} & -\boldsymbol{c}^T\boldsymbol{x} \\ \textit{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geqslant 0. \end{array} \tag{2.6}$$

This problem has the inequality constraint functions

$$f_i(\boldsymbol{x}) = -\boldsymbol{x}^{(i)}, i = 1, ..., n.$$

We introduce the multipliers $\boldsymbol{\lambda}^{(i)}$ for the n inequality constraints and the multipliers $\boldsymbol{\nu}^{(i)}$ for the equality constraints. The Lagrangian is given by

$$\begin{aligned} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) &= \boldsymbol{c}^T\boldsymbol{x} - \sum_{i=1}^{n} \boldsymbol{\lambda}^{(i)}\boldsymbol{x}^{(i)} + \boldsymbol{\nu}^T(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}) \\ &= -\boldsymbol{b}^T\boldsymbol{\nu} + (\boldsymbol{c} + \boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda})^T\boldsymbol{x}. \end{aligned}$$

The dual function in this case is,

$$g\left(\boldsymbol{\lambda}, \boldsymbol{\nu}\right) = \inf_{\boldsymbol{x}} \mathcal{L}\left(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}\right) = -\boldsymbol{b}^T\boldsymbol{\nu} + \inf_{\boldsymbol{x}}(\boldsymbol{c} + \boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda})^T\boldsymbol{x}.$$

This function can analytically determined. When we have a linear function such that it is identically zero, it must be bounded from below, i.e., $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = -\infty$ except in case of $\boldsymbol{c} + \boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda} = 0$, in this case, it is $-\boldsymbol{b}^T\boldsymbol{\nu}$:

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \{ \begin{array}{ll} -\boldsymbol{b}^T\boldsymbol{\nu}, & \boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda} + \boldsymbol{c} = 0 \\ -\infty, & \textit{otherwise}. \end{array}$$

The dual function g is finite only on a proper affine subset of $\mathbb{R}^m \times \mathbb{R}^p$. The lower bound property (2.5) is nontrivial only when $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ satisfy $\boldsymbol{\lambda} \geqslant 0$ and $\boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda} + \boldsymbol{c} = 0$. Then, the lower bound on the optimal value of the problem (2.6) is $-\boldsymbol{b}^T\boldsymbol{\nu}$.

2.4.3 The Lagrange Dual Problem

The lower bound on the optimal value p^* of the problem (2.3) for each $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ such that $\boldsymbol{\lambda} \geqslant 0$ is the Lagrange dual function. From this point, the parameters $\boldsymbol{\lambda}, \boldsymbol{\nu}$ have the most important role to determine the lower bound. One can ask, how can we find the best lower bound that can be obtained from the Lagrange dual function? This question leads us to the following optimization problem.

$$\begin{array}{ll} maximize & g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \\ subject\ \ to & \boldsymbol{\lambda} \geqslant 0. \end{array} \tag{2.7}$$

Problem (2.7) is called *the Lagrange dual problem* associated with problem (2.3). The original problem (2.3) is called the *primal problem.* If $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is feasible for the dual problem (2.7) then we refer to $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ as *dual optimal* for the problem (2.7). The advantage of the Lagrange problem is that it is a convex optimization problem, since the objective function is concave and the constraint is convex. This is the case whether the primal problem (2.3) is convex or not.

Explicitly of the Dual Constraints

It may happen that the domain of the dual function,

$$dom\ \ g = \{(\boldsymbol{\lambda}, \boldsymbol{\nu}) : g(\boldsymbol{\lambda}, \boldsymbol{\nu}) > -\infty\},$$

has a dimension that is smaller than $m + p$. Let us discuss the case (which often occurs) when we identify the affine hull of *dom g*, and represent it as a set of linear equality constraints. This means that the equality constraints which are implicit can be identified in the objective function g of the dual problem (2.7). The result is an equivalent problem such that the equality constraints are given explicitly as constraints. This idea can be demonstrated using the following example.
Example: *Lagrange dual problem of the LP in standard form*

$$\begin{array}{ll} maximize & g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \{ \begin{array}{ll} -\boldsymbol{b}^T\boldsymbol{\nu}, & \boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda} + \boldsymbol{c} = 0 \\ -\infty, & otherewise \end{array} \\ subject\ \ to & \qquad\qquad \boldsymbol{\lambda} \geqslant 0. \end{array} \tag{2.8}$$

Here, g is finite only when $\boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda} + \boldsymbol{c} = 0$. The equivalent problem is formed by making these equality constraints explicit:

$$\begin{array}{ll} maximize & -\boldsymbol{b}^T\boldsymbol{\nu} \\ subject\ \ to & \boldsymbol{A}^T\boldsymbol{\nu} - \boldsymbol{\lambda} + \boldsymbol{c} = 0 \\ & \boldsymbol{\lambda} \geqslant 0. \end{array} \tag{2.9}$$

Then, this problem can be expressed as

$$\begin{array}{ll} \mathit{maximize} & -\boldsymbol{b}^T\boldsymbol{\nu} \\ \mathit{subject\ \ to} & \boldsymbol{A}^T\boldsymbol{\nu}+\boldsymbol{c}\geq 0, \end{array} \tag{2.10}$$

which is LP in inequality form, since $\boldsymbol{\lambda}$ can be viewed as a slack variable [11]. The Lagrange dual of the standard form LP (2.6) is the problem (2.8), which is equivalent to the problem (2.9) and (2.10).

Weak Duality

The optimal value d^* of the Lagrange dual problem is the lower bound on the optimum solution p^* of the primal problem. The weak duality property is simply expressed as,

$$d^* \leq p^*, \tag{2.11}$$

which holds even when the original primal problem is not convex. The weak duality inequality (2.11) holds if both d^* and p^* are infinite. As an example, when we have unbounded below primal problem such that $p^* = -\infty$, we have $d^* = -\infty$. This condition means the infeasibility of the Lagrange dual problem. When we have an unbounded above dual function, such that $d^* = \infty$, we have $p^* = \infty$. This condition means the infeasibility of the primal problem.

The difference between the primal solution and the dual solution,

$$p^* - d^*,$$

is referred as the optimality duality gap of the original problem, since it is the gap between the primal problem solution and the greatest lower bound on it which can be obtained using the Lagrange dual function. This gap is always nonnegative. The inequality (2.11) can be used to find a lower bound on the difficult problem since the dual problem is always convex which in many cases can be efficiently solved to find d^*.

Strong Duality

Strong duality

$$d^* = p^*,$$

holds when the optimality gap is zero. Which means that the Lagrange dual function obtains a tight best bound.

In general, the strong duality does not hold, but it holds only in case of the convexity of the primal problem (2.3) as defined in (2.4), which can also written as,

$$\begin{array}{ll} \textit{minimize} & f_0(\boldsymbol{x}) \\ \textit{subject to} & f_i(\boldsymbol{x}) \le 0, \quad i = 1, ..., m, \\ & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}. \end{array}$$

The functions $f_0, ..., f_m$ are defined as convex functions. we often (but not always) have strong duality. There are some results which verify conditions on the problem, beyond convexity, under which strong duality holds. These conditions are known by the *constraint qualifications. Slater's condition* is a simple constraint qualification which is stated as: There exists $\boldsymbol{x}$ such that

$$f_i(\boldsymbol{x}) < 0, \; i = 1, ..., m, \; \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}.$$

The point $\boldsymbol{x}$ is named *strictly feasible*, since the inequality constraints hold with strict inequality. Slater's theorem states that the strong duality holds, if Slater's condition holds (and the problem is convex).

Consider that we have some of the inequality constraints functions f_i such that they are affine, Slater's condition can be refined. Slater's condition and its refinement imply strong duality for convex problems. In addition, they also imply that the dual optimal value is attained when $d^* > -\infty$, i.e., there exists a dual feasible $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ with $g(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*) = d^* = p^*$. The proof of Slater's condition was presented in [11] using the separating hyperplane theorem for convex sets.

2.5 Optimality Conditions

We note that, we do not assume the problem (2.3) is convex, unless explicitly stated.

Sub-Optimality and Stopping Criteria

The lower bound $p^* \geqslant g(\boldsymbol{\lambda}, \boldsymbol{\nu})$ on the optimum value of the primal problem is established if the dual feasible $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ exists. Thus a dual feasible point $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ provides a proof that $p^* \geqslant g(\boldsymbol{\lambda}, \boldsymbol{\nu})$.

Even if we do not know the exact value of p^*, the dual feasible points allow us to bound how suboptimal a given point is. If $\boldsymbol{x}$ is a primal feasible and $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is a dual feasible, then

$$f_0(\boldsymbol{x}) - p^* \leqslant f_0(\boldsymbol{x}) - g(\boldsymbol{\lambda}, \boldsymbol{\nu}).$$

In particular, this proves that $\boldsymbol{x}$ is ϵ-suboptimal, $\epsilon = f_0(\boldsymbol{x}) - g(\boldsymbol{\lambda}, \boldsymbol{\nu})$. (It also proves that $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is ϵ-suboptimal for the dual problem). There is a duality gap associated with the primal feasible point $\boldsymbol{x}$ and dual feasible point $(\boldsymbol{\lambda}, \boldsymbol{\nu})$. Such gap is the gap between the primal and the dual objectives, $f_0(\boldsymbol{x}) - g(\boldsymbol{\lambda}, \boldsymbol{\nu})$. If we have a zero duality gap of the primal dual feasible pair $\boldsymbol{x}$, $(\boldsymbol{\lambda}, \boldsymbol{\nu})$, i.e., $f_0(\boldsymbol{x}) = g(\boldsymbol{\lambda}, \boldsymbol{\nu})$, then $\boldsymbol{x}$ is primal optimal and $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is dual optimal.

In optimization algorithms, we can use these observations in order to provide non-heuristic stopping criteria. If there is an algorithm that produces a sequence of primal feasible $\boldsymbol{x}_k$ and dual feasible $(\boldsymbol{\lambda}_k, \boldsymbol{\nu}_k)$, for $k = 1, 2, ...$, and $\epsilon_{abs} > 0$ is given required absolute accuracy, then the stopping criterion (i.e., the condition for terminating the algorithm)

$$f_0(\boldsymbol{x}_k) - g(\boldsymbol{\lambda}_k, \boldsymbol{\nu}_k) \leqslant \epsilon_{abs},$$

guarantees that when the algorithm terminates, $\boldsymbol{x}_k$ is ϵ_{abs}-suboptimal. (Strong duality must hold if this method works for arbitrarily small tolerance ϵ_{abs}).

Complementary Slackness

Let the values of the primal and dual optimal be attained and equal (which means that, we have strong duality condition). Suppose $\boldsymbol{x}^*$ is a primal optimal and $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ is a dual optimal point, so the following condition holds.

$$\boldsymbol{\lambda}^{(i)^*} f_i(\boldsymbol{x}^*) = 0, i = 1, ..., m.$$

This condition is called *complementary slackness*. The condition holds for any primal optimal $\boldsymbol{x}^*$ and any dual optimal $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ (when strong duality holds). The complementary slackness condition can be expressed as

$$\boldsymbol{\lambda}^{(i)^*} > 0 \Rightarrow f_i(\boldsymbol{x}^*) = 0,$$

or, equivalently,

$$f_i(\boldsymbol{x}^*) < 0 \Rightarrow \boldsymbol{\lambda}^{(i)^*} = 0,$$

which means that the ith optimal Lagrange multiplier is zero unless the ith constraint is active at the optimum.

KKT Optimality Conditions

Consider the functions $f_0, ..., f_m, h_1, ..., h_p$ in problem (2.3) are differentiable. From this property, we can say that, their domains are open. Note that, there are no assumptions about the convexity.

Suppose $\boldsymbol{x}^*$ and $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ are any primal and dual optimal points respectively with zero duality gap. Since $\boldsymbol{x}^*$ minimizes $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ over $\boldsymbol{x}$, it follows that its gradient must vanish at $\boldsymbol{x}^*$, i.e.,

$$\boldsymbol{\nabla} f_0(\boldsymbol{x}^*) + \sum_{i=1}^{m} \boldsymbol{\lambda}^{(i)^*} \boldsymbol{\nabla} f_i(\boldsymbol{x}^*) + \sum_{i=1}^{p} \boldsymbol{\nu}^{(i)^*} \boldsymbol{\nabla} h_i(\boldsymbol{x}^*) = 0.$$

Thus we obtain

$$\begin{array}{c} f_i(\boldsymbol{x}^*) \ \leqslant 0, \quad i = 1, ..., m \\ h_i(\boldsymbol{x}^*) = 0, \quad i = 1, ..., p \\ \boldsymbol{\lambda}^{(i)^*} \qquad \geqslant 0 \quad i = 1, ..., m \\ \boldsymbol{\lambda}^{(i)^*} f_i(\boldsymbol{x}^*) = 0 \quad i = 1, ...m \\ \boldsymbol{\nabla} f_0(\boldsymbol{x}^*) + \sum\limits_{i=1}^{m} \boldsymbol{\lambda}^{(i)^*} \boldsymbol{\nabla} f_i(\boldsymbol{x}^*) + \sum\limits_{i=1}^{p} \boldsymbol{\nu}^{(i)^*} \boldsymbol{\nabla} h_i(\boldsymbol{x}^*) = 0, \end{array} \tag{2.12}$$

which are known by *Karush-Kuhn-Tucker* (KKT) conditions.

To conclude, if the strong duality holds for any optimization problem with differentiable objective and constraint functions, primal and dual optimal points must satisfy the KKT conditions (2.12).

In case of a primal convex optimization problem, the KKT conditions are also sufficient for the points to be primal and dual optimum. Or we can say, when problem (2.3) has convex functions f_i and affine functions h_i. If $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{\lambda}}$ and $\tilde{\boldsymbol{\nu}}$ are any points that satisfy the KKT conditions (2.12), then $\tilde{\boldsymbol{x}}$ and $(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\nu}})$ are primal dual optimal with zero duality gap. If a convex optimization problem with differentiable objective and constraint functions satisfies Slater's condition, then the KKT conditions provide necessary and sufficient conditions for optimality. Slater's condition implies that the optimal duality gap is zero and the dual optimum is attained, so $\boldsymbol{x}$ is optimal if and only if there is a pair $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ together with $\boldsymbol{x}$ satisfy the KKT conditions.

The KKT conditions are important in optimization problems. In some special cases it is possible to solve the KKT conditions (and hence, the optimization problem) analytically. To generalize, many algorithms for convex optimization can be interpreted as methods for solving the KKT conditions.

Example: *Convex quadratic minimization with equality constraints*
Consider the quadratic programming (QP) problem

$$\begin{array}{ll} \textit{minimize} & (1/2)\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} + r \\ \textit{subject to} & \qquad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \end{array} \tag{2.13}$$

where $\boldsymbol{P} \in \mathbb{S}_+^n$. The KKT conditions for this problem are

$$\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{b}, \;\; \boldsymbol{P}\boldsymbol{x}^* + \boldsymbol{q} + \boldsymbol{A}^T\boldsymbol{\nu}^* = \boldsymbol{0},$$

which can be written as,

$$\begin{bmatrix} \boldsymbol{P} & \boldsymbol{A}^T \\ \boldsymbol{A} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^* \\ \boldsymbol{\nu}^* \end{bmatrix} = \begin{bmatrix} -\boldsymbol{q} \\ -\boldsymbol{b} \end{bmatrix}.$$

We can find the optimal primal and dual variables for problem (2.13) by solving this set of $m + n$ equations in the $m + n$ variables $\boldsymbol{x}^*, \boldsymbol{\nu}^*$. This set of equations is called the KKT system for equality constrained quadratic optimization problem (2.13). The coefficient matrix is known as the *KKT matrix*. When this matrix is nonsingular, we can say that there is a unique optimal primal-dual pair $(\boldsymbol{x}^*, \boldsymbol{\nu}^*)$, while if this matrix is singular, but the KKT system is solvable, then any solution yields an optimal pair $(\boldsymbol{x}^*, \boldsymbol{\nu}^*)$. In case of the KKT system is not solvable, the quadratic optimization problem is unbounded from below. In this case there exist $\boldsymbol{v} \in \mathbb{R}^n$ and $\boldsymbol{\omega} \in \mathbb{R}^p$ such that

$$\boldsymbol{p}\boldsymbol{v} + \boldsymbol{A}^T\boldsymbol{\omega} = \boldsymbol{0}, \;\; \boldsymbol{A}\boldsymbol{v} = \boldsymbol{0}, \;\; -\boldsymbol{q}^T\boldsymbol{v} + \boldsymbol{b}^T\boldsymbol{\omega} > 0.$$

Suppose that $\hat{\boldsymbol{x}}$ is any feasible point. The point $\boldsymbol{x} = \hat{\boldsymbol{x}} + t\boldsymbol{v}$ is feasible for all t and

$$\begin{aligned} f(\hat{\boldsymbol{x}} + t\boldsymbol{v}) &= f(\hat{\boldsymbol{x}}) + t(\boldsymbol{v}^T\boldsymbol{P}\hat{\boldsymbol{x}} + \boldsymbol{q}^T\boldsymbol{v}) + (1/2)t^2\boldsymbol{v}^T\boldsymbol{p}\boldsymbol{v} \\ &= f(\hat{\boldsymbol{x}}) + t(-\hat{\boldsymbol{x}}^T\boldsymbol{A}^T\boldsymbol{\omega} + \boldsymbol{q}^T\boldsymbol{v}) - (1/2)t^2\boldsymbol{\omega}^2\boldsymbol{A}\boldsymbol{v} \\ &= f(\hat{\boldsymbol{x}}) + t(-\boldsymbol{b}^T\boldsymbol{\omega} + \boldsymbol{q}^T\boldsymbol{v}), \end{aligned}$$

which decreases without bound at $t \to \infty$.

Solving the Primal Problem via the Dual

If strong duality holds and a dual optimal solution $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ exists, then any primal optimal point is also a minimizer of $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$. So, we can get the primal optimal solution from the dual optimal solution.

Propose that we have strong duality and a known optimal solution, $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$. Propose also that the minimizer of the function $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$, i.e., the solution of

$$\textit{minimize} \;\; f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \boldsymbol{\lambda}^{(i)^*} f_i(\boldsymbol{x}) + \sum_{i=1}^{p} \boldsymbol{\nu}^{(i)^*} h_i(\boldsymbol{x}), \tag{2.14}$$

is unique. (For a convex problem this is always the case, for example, if $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ is a strictly convex function of $\boldsymbol{x}$). Then, if the solution of (2.14) is primal feasible, it is a primal optimal; if it is not primal feasible, then there is no primal optimal, so we can conclude that the primal optimum is not attained. We can say that, this observation is powerful when the primal problem is harder to solve than the dual problem, for example the dual problem can be solved analytically or it has some special structure that can be exploited (as we will present later in this thesis). The following simple example shows how this can be used in decomposition of a large problem into small subproblems.

Example: *Minimizing a separable function subject to an equality constraint.*
Consider the problem

$$\begin{array}{ll} \textit{minimize} & f_0(\boldsymbol{x}) = \sum_{i=1}^{n} f_i(\boldsymbol{x}^{(i)}) \\ \textit{subject to} & \boldsymbol{a}^T\boldsymbol{x} = b, \end{array}$$

where $\boldsymbol{a} \in \mathbb{R}^n, b \in \mathbb{R}$ and $f_i : \mathbb{R} \to \mathbb{R}$ are differentiable and *strictly convex*. The objective function is called *separable objective function* since it is a sum of functions of individual variables $\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(n)}$. Assume also that the domain of f_0 intersects the constraint set, i.e., there exists a point $\boldsymbol{x}_0 \in domf$ with $\boldsymbol{a}^T\boldsymbol{x}_0 = b$, which implies the problem has a unique optimal point $\boldsymbol{x}^*$.
The Lagrange of this problem is

$$\mathcal{L}(\boldsymbol{x}, \nu) = \sum_{i=1}^{n} f_i(\boldsymbol{x}^{(i)}) + \nu(\boldsymbol{a}^T\boldsymbol{x} - b) = -b\nu + \sum_{i=1}^{n}(f_i(\boldsymbol{x}^{(i)}) + \nu\boldsymbol{a}^{(i)}\boldsymbol{x}^{(i)}),$$

which is also separable, and then the dual function is

$$\begin{aligned} g(\nu) &= -b\nu + \inf_{\boldsymbol{x}}(\sum_{i=1}^{n}(f_i(\boldsymbol{x}^{(i)}) + \nu\boldsymbol{a}^{(i)}\boldsymbol{x}^{(i)}) \\ &= -b\nu + \sum_{i=1}^{n} \inf_{\boldsymbol{x}^{(i)}}(f_i(\boldsymbol{x}^{(i)}) + \nu\boldsymbol{a}^{(i)}\boldsymbol{x}^{(i)}) \\ &= -b\nu - \sum_{i=1}^{n} f_i^*(-\nu\boldsymbol{a}^{(i)}), \end{aligned}$$

where $f_i^*(-\nu\boldsymbol{a}^{(i)}) = \inf_{\boldsymbol{x}^{(i)}}(f_i(\boldsymbol{x}^{(i)}) + \nu\boldsymbol{a}^{(i)}\boldsymbol{x}^{(i)})$. The dual problem is thus

$$maximize \quad -b\nu - \sum_{i=1}^{n} f_i^*(-\nu\boldsymbol{a}^{(i)}),$$

with a scalar variable $\nu \in \mathbb{R}$.

Propose that $f_i^*, i = 1, ..., n$ are easy to evaluate, therefore the dual problem is easy to solve. Propose also that we found an optimal dual variable ν^*. (At that point, there are many methods or algorithms can solve a convex problem with one scalar variable, such as the gradient descent algorithm and the bisection method). Since each f_i is strictly convex, the function $\mathcal{L}(\boldsymbol{x}, \nu^*)$ is strictly convex in $\boldsymbol{x}$, and so it has a unique minimizer $\tilde{\boldsymbol{x}}$. But we also know that $\boldsymbol{x}^*$ minimizes $\mathcal{L}(\boldsymbol{x}, \nu^*)$, so we must have $\tilde{\boldsymbol{x}} = \boldsymbol{x}^*$. We can recover $\boldsymbol{x}^*$ from $\boldsymbol{\nabla}_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}, \nu^*) = 0$, i.e., by solving the equations $f_i'(\boldsymbol{x}^{(i)^*}) = -\nu^*\boldsymbol{a}^{(i)}$.

2.6 Interior Point Algorithms

During the last thirty years, there has been a revolution in methods to solve the optimization problems. In the early $1980s$, quadratic programming and augmented Lagrangian methods were favored for nonlinear problems, while simplex method was basically unchallenged for linear programming. Since then, modern interior point methods have infused virtually every area of continuous optimization.

In this section, we present a summary of the barrier method as a special class of interior point methods. These kind of methods have much advantages such that, the implementation of them is simple and they have a good performance. In this context, we focus on the key algorithmic components for practical implementation of interior point methods. These key components are backtracking line-search and the newton method for equality constrained minimization.

We will present that, the duality role is very efficient in these algorithms. In particular, we will get an exact stopping criteria by using the duality, interior point methods. The desired tolerance of these methods can be determined and when the search terminates, the returned decision vector is guaranteed to be within the specified tolerance of the optimum. This is completely different from the case of other methods which terminate simply when the rate of progress becomes slow, but without any guarantees on the optimality of the returned decision vector.

2.6.1 Descent Methods and Line Search

Let us try to solve the following *unconstrained optimization problem*,

$$\textit{minimize} \;\; f(\boldsymbol{x}), \tag{2.15}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a twice continuously differentiable convex function, and its domain, *dom* f is open. Let this problem be solvable, i.e., there exists an optimal point $\boldsymbol{x}^*$. Let the optimal value denoted by p^*.

As we supposed f is differentiable and convex, a necessary and sufficient condition for a point $\boldsymbol{x}^*$ to be optimal is

$$\boldsymbol{\nabla} f(\boldsymbol{x}^*) = 0. \tag{2.16}$$

Now, to solve the unconstrained minimization problem (2.15) we can easily find the solution of (2.16), which is a set of n equations in the n variables $\boldsymbol{x}^{(1)}, ..., \boldsymbol{x}^{(n)}$. In many cases, we can find a solution to the problem (2.15) by analytically solving the optimality equation (2.16), but usually the problem must be solved by an iterative algorithm. That means an algorithm which computes a sequence of points $\boldsymbol{x}_0, \boldsymbol{x}_1, ... \in$ *domf* with $f(\boldsymbol{x}_k) \to p^*$ as $k \to \infty$. Such a sequence of points is called a *minimizing sequence* of the problem (2.15). The algorithm is terminated when $f(\boldsymbol{x}_k) - p^* \leqslant \epsilon$, where $\epsilon > 0$ is some specified tolerance.

Descent Methods

In this section, we described the algorithms that produce a minimizing sequence $\boldsymbol{x}_k, \ k = 1, ...,$ where

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + t^k \boldsymbol{\Delta x}_k,$$

and $t^k > 0$ ($t^k = 0$ only when $\boldsymbol{x}_k$ is optimal). A vector $\boldsymbol{\Delta x}$ in $\mathbb{R}^n$ is the *step* or the *search direction* and $k = 0, 1, ...$ denotes the number of iterations. The scalar $t^k \geqslant 0$ is the *step size* or the *step length* at k (even though it is not equal to $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|$ unless $\|\boldsymbol{\Delta x}_k\| = 1$).

These methods are called *descent methods*, that means

$$f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k),$$

except when $\boldsymbol{x}_k$ is optimal. Then, for all k we have $\boldsymbol{x}_k \in S$, where S is the initial sub-level set and hence $\boldsymbol{x}_k \in$ *domf*. We know from convexity that $\boldsymbol{\nabla} f(\boldsymbol{x}_k)^T(\boldsymbol{y} - \boldsymbol{x}_k) \geq 0$ implies $f(\boldsymbol{y}) \geq f(\boldsymbol{x}_k)$, so the search direction in a descent method satisfies

$$\boldsymbol{\nabla} f(\boldsymbol{x}_k)^T \boldsymbol{\Delta x}_k < 0,$$

i.e., it makes an acute angle with the negative gradient. This direction is named the *descent direction* (for f, at $\boldsymbol{x}_k$).

The gradient descent algorithm operates as follows. It has two basic steps, the first step determines the descent direction $\boldsymbol{\Delta x}$, and the second step selects the step size t.

The selection of the step size t determines where along the line $\boldsymbol{x} + t\boldsymbol{\Delta x} : t \in \mathbb{R}_+$ the next iterate will be. Then the second step of the gradient descent algorithm is called the line search.

```
Algorithm General descent method.
given a starting point $\boldsymbol{x} \in dom f$.
repeat
    1. Determine a descent direction $\boldsymbol{\Delta x}$.
    2. Line search. Choose step size $t$.
    3. Update. $\boldsymbol{x} := \boldsymbol{x} + t\boldsymbol{\Delta x}$.
until stopping criterion is satisfied.
```

Backtracking Line Search

Among all line search methods is a *backtracking line search* which is very simple and effective method. It depends on two constants α and β with $0 < \alpha < 0.5, 0 < \beta < 1$.
The line search starts with unit step size and then reduces it by the factor β until the stopping condition $f(\boldsymbol{x}+t\boldsymbol{\Delta x}) \leqslant f(\boldsymbol{x})+\alpha t\boldsymbol{\nabla} f(\boldsymbol{x})^T\boldsymbol{\Delta x}$ holds. Hence, it has the name backtracking. Since $\boldsymbol{\Delta x}$ is a descent direction, $\boldsymbol{\nabla} f(\boldsymbol{x})^T\boldsymbol{\Delta x} < 0$, we have for small t

$$f(\boldsymbol{x}+t\boldsymbol{\Delta x}) \approx f(\boldsymbol{x}) + t\boldsymbol{\nabla} f(\boldsymbol{x})^T\boldsymbol{\Delta x} < f(\boldsymbol{x}) + \alpha t\boldsymbol{\nabla} f(\boldsymbol{x})^T\boldsymbol{\Delta x},$$

which shows that the backtracking line search eventually terminates. The constant α can be considered as the fraction of the decrease in f predicted by the linear extrapolation. For more details about how can we choose the parameters t, α, and β you can follow [11].
The descent direction $\boldsymbol{\Delta x}$ is computed using the Newton method for equality constrained minimization.

2.6.2 Newton Method for Equality Constrained Minimization

One of the most powerful methods that solve an equality constrained convex optimization problem is the *Newton method.* Consider the problem,

$$\begin{array}{ll} \textit{minimize} & f(\boldsymbol{x}) \\ \textit{subject to} & \boldsymbol{Ax} = \boldsymbol{b}, \end{array} \tag{2.17}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a twice continuously differentiable and convex, and $\boldsymbol{A} \in \mathbb{R}^{p \times n}$ is of rank $\boldsymbol{A} = p < n$. Hence, the number of equality constraints are less than the number of variables, taking into account that the equality constraints are independent. By noting the assumption of existence of an optimization solution $\boldsymbol{x}^*$, the optimal value is denoted by p^* such that,

$$p^* = \inf \{f(\boldsymbol{x}) | \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\} = f(\boldsymbol{x}^*).$$

By eliminating the equality constraints, the equality constrained minimization problem can be equivalently transformed to unconstrained problem. However, we can focus on the extension of Newton's method which deals with equality rather than those methods that eliminate the inequalities. There are many reasons of that, the first reason is the problem structure such as sparsity is destroyed by elimination or even by forming the dual. While in the other hand, the method that deals with equality constraints can exploit the problem structure. The second reason is conceptual: methods that deal with equality constraints can be thought as methods that solve the optimality conditions,

$$\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{b}, \quad \boldsymbol{\nabla} f(\boldsymbol{x}^*) + \boldsymbol{A}^T \boldsymbol{\nu}^* = 0. \tag{2.18}$$

Instead of solving the equality constrained optimization problem (2.17), it is much easier to find the solution of KKT equations (2.18), which is a set of $n + p$ equations in the $n + p$ variables $\boldsymbol{x}^*, \boldsymbol{\nu}^*$. There are a few problems for which we can analytically solve these optimality conditions as the unconstrained optimization. The quadratic function f is an important case.

The Newton Step

Almost Newton's methods are used in all interior point methods to find the descent direction. Newton's method presents the fastest convergence rates among all known techniques that compute the descent direction. This superior performance is because the Newton method uses second derivative information from the Hessian of the objective in its computation of the descent direction. The direction is chosen along the set that defined by the equality constraints, which minimizes a local quadratic approximation of the objective function at the last iteration. The Newton step $\boldsymbol{\Delta x}_{nt}$ is simply what is added to $\boldsymbol{x}$ to solve the problem when the quadratic approximation is used in place of f in (2.17). The Newton step $\boldsymbol{\Delta x}_{nt}$ for the optimization problem (2.17) is characterized by,

$$\begin{bmatrix} \boldsymbol{\nabla}^2 f(\boldsymbol{x}) & \boldsymbol{A}^T \\ \boldsymbol{A} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta x}_{nt} \\ \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{\nabla} f(\boldsymbol{x}) \\ \boldsymbol{0} \end{bmatrix},$$

where $\boldsymbol{\omega}$ is the associated optimal dual variable for the quadratic problem. We only define the Newton step at points for which the KKT matrix is nonsingular. Consider the unconstrained problems, when the objective function f is exactly quadratic, the Newton update $\boldsymbol{x} + \boldsymbol{\Delta x}_{nt}$ solves the equality constrained minimization problem. The vector $\boldsymbol{\omega}$ is the optimal dual variable for the original problem. Then, when the objective function f is nearly quadratic, $\boldsymbol{x} + \boldsymbol{\Delta x}_{nt}$ should be a good estimate of the solution $\boldsymbol{x}^*$, and $\boldsymbol{\omega}$ should be a good estimate of the dual solution $\boldsymbol{\nu}^*$. Newton's method in unconstrained minimization problems reduces to solve,

$$\boldsymbol{\Delta x}_{nt} = -(\boldsymbol{\nabla}^2 f(\boldsymbol{x}))^{-1} \boldsymbol{\nabla} f(\boldsymbol{x}).$$

Feasible Directions and Newton Decrement

For the equality constrained problem, the Newton decrement is defined as,

$$\lambda(\boldsymbol{x}) = (\boldsymbol{\Delta x}_{nt}^T \boldsymbol{\nabla}^2 f(\boldsymbol{x}) \boldsymbol{\Delta x}_{nt})^{1/2}.$$

The term $\lambda(\boldsymbol{x})^2/2$ gives an estimate of $f(\boldsymbol{x}) - p^*$ and $\lambda(\boldsymbol{x})$ serves as the basis of a very good stopping criterion.

If $\boldsymbol{Av} = \boldsymbol{0}$, then we say that $\boldsymbol{v} \in \mathbb{R}^n$ is *feasible direction.* Let $\boldsymbol{Ax} = \boldsymbol{b}$, so each point of the form $\boldsymbol{x} + t\boldsymbol{v}$ is also feasible, i.e., $\boldsymbol{A}(\boldsymbol{x} + t\boldsymbol{v} = \boldsymbol{b})$. If for small $t > 0$, then $f(\boldsymbol{x} + t\boldsymbol{v}) < f(\boldsymbol{x})$, $\boldsymbol{v}$ is said to be a *descent direction* for f at $\boldsymbol{x}$.

The Newton step is feasible descent direction except when $\boldsymbol{x}$ is optimal, in such case $\boldsymbol{\Delta x}_{nt} = \boldsymbol{0}$. The second set of equations that defines $\boldsymbol{\Delta x}_{nt}$ are $\boldsymbol{A\Delta x}_{nt} = \boldsymbol{0}$, which shows that it is a feasible direction. It is a descent direction follows from the property that the directional derivative of f along $\boldsymbol{\Delta x}_{nt}$ is negative which is exactly $-\lambda(\boldsymbol{x})^2$.

Newton Method with Equality Constraints

The Newton method with equality constraints is given by the following algorithm:

```
Algorithm Newton's method for equality constrained minimization.
given starting point $\boldsymbol{x} \in dom f$ with $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, tolerance $\epsilon > 0$.
repeat
    1. Compute the Newton step and decrement $\boldsymbol{\Delta x}_{nt}$, $\lambda(\boldsymbol{x})$.
    2. Stopping criterion, quit if $\lambda^2/2 \leq \epsilon$.
    3. Line search, choose step size $t$ by backtracking line search.
    4. Update, $\boldsymbol{x} := \boldsymbol{x} + t\boldsymbol{\Delta x}_{nt}$.
```

Since all the iterates are feasible, with $f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k)$ (unless $\boldsymbol{x}_k$ is optimal), the method is called a *feasible descent method.* Newton's method requires that KKT matrix be invertible at each $\boldsymbol{x}$.

3 Signal Detection, Estimation and Modulation

In this chapter, an introduction to detection theory, estimation theory and modulation schemes will be given. The aim is to demonstrate how they can be used to solve the communication problems discussed in this thesis. Much of the material in this chapter is based on [50, 84].

A simple digital communication system can be considered as shown in Figure 3.1. As a simple description of this system, the source generates a binary digit every time interval (T seconds). The problem now is how to transmit this sequence of digits from the source to some other destination. The channel is the medium that used in this transmission. As an example of the channel that transmits this sequence is the radio link. In general, when we need to transmit information through the channel, we must convert them into a form that is suitable for propagation over the channel. A direct method would be to establish a device that generates a sine wave,

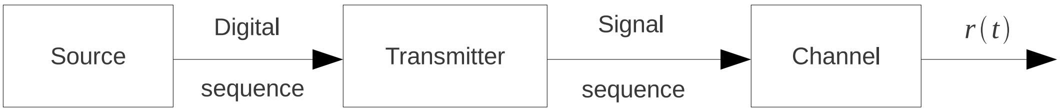


Figure 3.1: Digital communication system.

$$\boldsymbol{s}_1(t) = \sin \omega_1 t,$$

for T seconds in case if the source generated a '**one**' in the previous interval, and a sine wave with different frequency,

$$\boldsymbol{s}_0(t) = \sin\, \omega_0 t,$$

for T seconds in case of the source generated a '**zero**' in the previous interval. The frequencies are chosen such that the signals $\boldsymbol{s}_0(t)$ and $\boldsymbol{s}_1(t)$ will propagate over the radio link. The output of the device transmits through the channel. In this context, we present a simple transmission system that can easily written as,

$$\boldsymbol{r}(t) = \boldsymbol{s}_{\Omega_1}(t) + \boldsymbol{n}(t),$$

if $\boldsymbol{s}_1(t)$ was transmitted, where, $\boldsymbol{r}(t)$ is a waveform which in this case is produced every T-second interval and $\boldsymbol{n}(t)$ is the addition noise. If $\boldsymbol{s}_0(t)$ was transmitted, we have,

$$\boldsymbol{r}(t) = \boldsymbol{s}_{\Omega_0}(t) + \boldsymbol{n}(t).$$

We can generally write these equations as

$$\boldsymbol{r}(t) = \boldsymbol{s}_{\Omega_i}(t) + \boldsymbol{n}(t),$$

where, $\boldsymbol{s}_{\Omega_i}(t)$ is a sample function from a random process centered at ω_i, in case of $i = 0$ or $i = 1$.

Detection Problem

The problem of deciding which of the two possible signals was transmitted is named a *detection problem*. The *processor* is the device that decides which signal was transmitted. It observes $\boldsymbol{r}(t)$ and then according to some set of principles it determines which of $\boldsymbol{s}_{\Omega_1}(t)$ and $\boldsymbol{s}_{\Omega_0}(t)$ was sent. We presented a special case that has only possible source of error (the additive noise) to decide. This case is a simple case of detection problems, which is characterized by the shortage of any deterministic signal component. The difference in the statistical properties of the two random processes from which $\boldsymbol{s}_{\Omega_i}(t)$ is obtained plays an important role in the designing a decision procedure. We just mentioned that we must decide which of two alternatives is true, then this kind of detection problems is called *binary detection problems*. In general, problems that decide which of M ($M > 2$) alternatives is true are called *M ary detection problems.*

Estimation Problems

Now we will present the idea of the second area of our discussion in this chapter named *estimation problem.* Consider the received signal

$$\boldsymbol{r}(t) = \boldsymbol{s}_{\Omega}(t, a) + \boldsymbol{n}(t),$$

where $\boldsymbol{s}_{\Omega}(t, a)$ is a sample function from a random process and a is its amplitude. The receiver has to observe $\boldsymbol{r}(t)$ and uses the statistical properties of both $\boldsymbol{s}_{\Omega}(t, a)$ and $\boldsymbol{n}(t)$, it estimates the value of a. This kind of process is an example of an *estimation problem.*

3.1 Detection Theory

In this section we will study the basic terms in classical detection theory.

3.1.1 Binary Hypothesis

A simple decision theory problem as shown in Figure 3.2 has four components:

- Source.
- Probabilistic transition mechanism.
- Observation space.
- Decision rule.

The *source* is the first component that generates the output. As we presented when we defined the detection problem, we will continue in the same direction, i.e., we first consider the simplest case in which the

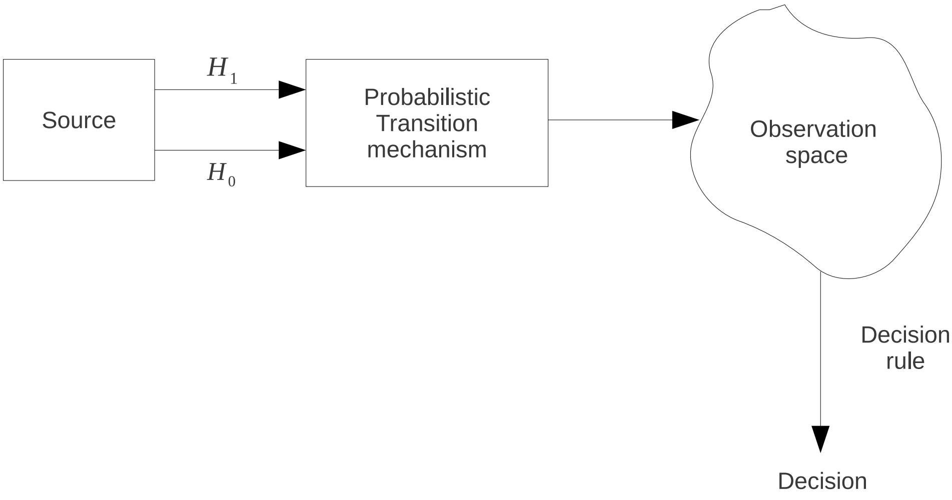


Figure 3.2: Components of a decision theory problem.

output is one of two choices. Through this chapter, we denote these two choices by H_0 and H_1. A digital communication system is a typical source mechanism, it transmits information by sending **zeros** (H_0) or **ones** (H_1). We do not know which hypothesis is true.

The *Probabilistic transition mechanism* is considered as the second component of a decision theory problem. Indeed, the transition mechanism can be proposed as a device that decides which hypothesis is true and hence, it generates a point in the *observation space* (the third component of the decision theory problem) according to some probability law.

Consider a simple example that is given in Figure 3.3*a*. In case of H_1 is true, $+1$ will be generated by the source and when H_0 is true the source generates -1. Now, consider if an independent discrete random variable n has probability density $p_n(N)$ is added to the source output as shown in Figure 3.3*b*. The observed variable r is the sum of the source output and n. Under our two hypotheses, we have,

$$\begin{aligned} H_1 :&\quad r = 1 + n, \\ H_0 :&\quad r = -1 + n. \end{aligned}$$

Figure 3.3*b* also shows the probability densities of r on these two hypotheses $p_{r|H_1}(\boldsymbol{R}|H_1)$ and $p_{r|H_0}(\boldsymbol{R}|H_0)$. In this case, any output can be plotted on a line, so the observation space is one-dimensional.

As an extension of this case, consider when the source generates two numbers in sequence. A random variable n_1 is added to the first number and an independent random variable n_2 is added to the second number.

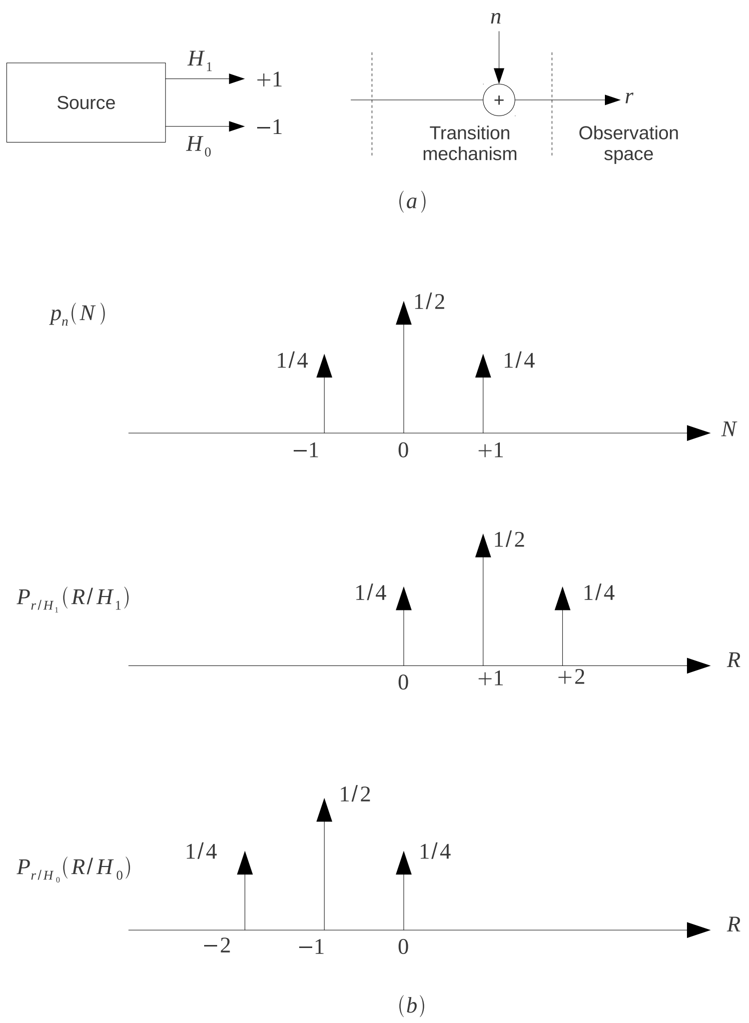


Figure 3.3: A simple decision problem.

Thus

$$\begin{aligned} H_1: \quad & r_1 = 1 + n_1 \\ & r_2 = 1 + n_2, \\ \\ H_0: \quad & r_1 = -1 + n_1 \\ & r_2 = -1 + n_2. \end{aligned}$$

The observation space in this extension case is two dimensional space and any observation can be represented as a point in a plane.

In this section, we only discuss the problems in which the observation space has finite-dimensional. In other words, the observation consists of a set of N numbers and can be represented as a point in an N dimensional space.

A decision rule is the last component of the detection problem. We just observe the outcome of the observation space, then we guess which hypothesis was true, and to accomplish this, we develop decision rules that assign each point to one of the hypotheses.

Simple Binary Hypothesis Tests

In the following, we will study the decision problem when each of the two sources outputs corresponds to a hypothesis. Each hypothesis in this case maps into a point in the observation space. Assume that the observation space corresponds to a set of N observations: $\boldsymbol{r}^{(1)}, \boldsymbol{r}^{(2)}, \ldots \boldsymbol{r}^{(N)}$. So each set can be represented as a point in an N- dimensional space and it can be given by a vector $\boldsymbol{r}$:

$$\boldsymbol{r} = [\boldsymbol{r}^{(1)}, \boldsymbol{r}^{(2)}, \ldots \boldsymbol{r}^{(N)}]^T.$$

Accordance to the two known conditional probability densities $p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)$ and $p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)$, the probabilistic transition mechanism generates points. To develop a suitable decision rule we must use these information.

Decision Criterion

In the binary hypothesis problem we know that either H_0 or H_1 is true. So, when the experiment is conducted, one of the following four cases can happen:

1. H_0 is true; choose H_0.
2. H_0 is true; choose H_1.
3. H_1 is true; choose H_1.
4. H_1 is true; choose H_0.

The first and the third cases describe the correct choices while the second and the fourth cases describe the errors. The method of processing the received data $\boldsymbol{r}$ depends on the decision criterion which we select.

Bayes Criterion

Two assumptions are made in using Bayes criterion. The first is the probability of occurrence of the source outputs (as a simple case, two source outputs), which is known. The second assumption is the assigned cost to each possible decision. Let p_1 and p_0 be the *a priori probabilities* of occurrence of hypothesis H_1 and H_0 respectively. There are four decision cases, then there is a cost that assigned to each case. The cost for these four cases is denoted by C_{00}, C_{10}, C_{11} and C_{01}, respectively. The first subscript refers to the chosen hypothesis while the second subscript refers to the true hypothesis. The objective of Bayes criterion is to design a decision rule so that the average cost which is also known as the risk $\mathcal{R}$, is minimized.

$$\begin{aligned}\mathcal{R} \; &= C_{00}\, p_0\, p_r \;\; (say\;\; H_0|H_0\;\; is\;\; true)\\ &+C_{10}\, p_0\, p_r \;\; (say\;\; H_1|H_0\;\; is\;\; true)\\ &+C_{11}\, p_1\, p_r \;\; (say\;\; H_1|H_1\;\; is\;\; true)\\ &+C_{01}\, p_1\, p_r \;\; (say\;\; H_0|H_1\;\; is\;\; true)\,,\end{aligned}$$

where, $p_r(:|:)$ is the probability that a particular case of action will be taken. Let Z be the whole observation space and we proposed that we only have two decision rule H_1 or H_0. So we can consider this rule as dividing the whole space Z into two subspaces Z_0 and Z_1 [5] as shown in Figure 3.4. If an observation falls in Z_0 then we say H_0, and if an observation falls in Z_1 then we say H_1. Using the transition probabilities

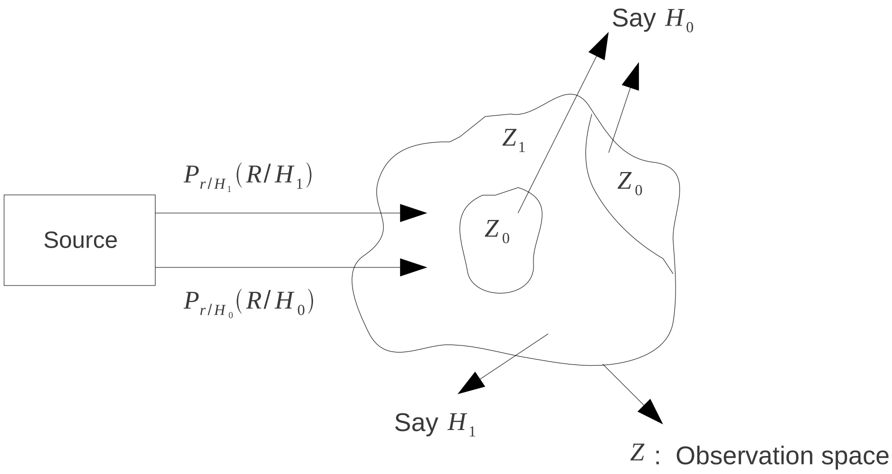


Figure 3.4: Decision regions.

and the decision regions, the expression of the risk can be written as,

$$\begin{aligned} \mathcal{R} &= C_{00}\, p_0 \textstyle\int_{Z_0} p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)\, d\boldsymbol{R} \\ &+C_{10}\, p_0 \textstyle\int_{Z_1} p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)\, d\boldsymbol{R} \\ &+C_{11}\, p_1 \textstyle\int_{Z_1} p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)\, d\boldsymbol{R} \\ &+C_{01}\, p_1 \textstyle\int_{Z_0} p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)\, d\boldsymbol{R}. \end{aligned} \tag{3.1}$$

Assume that the cost of making a wrong decision is higher than the cost of making a correct decision, or it can be stated as,

$$\begin{aligned} C_{10} &> C_{00}, \\ C_{01} &> C_{11}. \end{aligned} \tag{3.2}$$

To find the Bayes test, the risk must be minimized, so let this aim in the mind and try to choose the decision regions Z_0 and Z_1 in a way that achieves this aim. Each point from $\boldsymbol{R}$ in the observation space Z must be assigned to Z_0 or to Z_1. Thus,

$$Z = Z_0 + Z_1 \triangleq Z_0 \cup Z_1.$$

Rewriting (3.1), we have

$$\begin{aligned} \mathcal{R} &= p_0\, C_{00} \textstyle\int_{Z_0} p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)\, d\boldsymbol{R} + p_0\, C_{10} \int_{Z-Z_0} p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)\, d\boldsymbol{R} \\ &+p_1\, C_{01} \textstyle\int_{Z_0} p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)\, d\boldsymbol{R} + p_1\, C_{11} \int_{Z-Z_0} p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)\, d\boldsymbol{R}. \end{aligned} \tag{3.3}$$

Observing that,

$$\int_Z p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)\, d\boldsymbol{R} = \int_Z p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)\, d\boldsymbol{R} = 1,$$

(3.3) reduces to

$$\begin{aligned} \mathcal{R} = p_0\, C_{10} &+ p_1\, C_{11} \\ &+ \int_{Z_0} \{[p_1(C_{01} - C_{11}) p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)] \\ &- [p_0(C_{10} - C_{00}) p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)]\, \}\ d\boldsymbol{R}. \end{aligned}$$

We observe that the first two terms represent the fixed cost which is independent of how points are assigned in the observation space. The

only variable quantity is represented by the region of integration Z_0 which is the cost controlled by those points $\boldsymbol{R}$ that we assigned to Z_0. From (3.2), the two terms inside the brackets are both positive. Therefore, the risk is minimized by choosing the decision region Z_0 to include only points of $\boldsymbol{R}$ for which the second term is higher, and hence the integrand is negative. Similarly, when the first term is greater than the second term, all values of $\boldsymbol{R}$ will be excluded from Z_0 (assigned to Z_1) because they will contribute a positive amount to the integral. There are some values of $\boldsymbol{R}$ can be assigned arbitrarily, such values are the values where the two terms are equal so they have no effect on the cost. We assume that these points are assigned to H_1 and ignore them in our subsequent discussion. Hence, the decision regions are defined by the statement:
If

$$p_1(C_{01} - C_{11})p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1) \geqslant p_0(C_{10} - C_{00})p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0),$$

assign $\boldsymbol{R}$ to Z_1 and consequently say that H_1 is true. Or, assign $\boldsymbol{R}$ to Z_0 and say that H_0 is true.
We can also write,

$$\frac{p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)}{p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)} \gtrless_{H_0}^{H_1} \frac{p_0(C_{10} - C_{00})}{p_1(C_{01} - C_{11})}. \tag{3.4}$$

The quantity on the left is called the *likelihood ratio* and denoted by $\Phi(\boldsymbol{R})$,

$$\Phi(\boldsymbol{R}) = \frac{p_{\boldsymbol{r}|H_1}(\boldsymbol{R}|H_1)}{p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)}.$$

The *likelihood ratio* is a ratio of two functions of a random variable, so it is also a random variable. The right side of (3.4) is the threshold of the test and it is denoted by ξ:

$$\xi = \frac{p_0(C_{10} - C_{00})}{p_1(C_{01} - C_{11})}.$$

Thus Bayes criterion leads to a Likelihood Ration Test (LRT)

$$\Phi(\boldsymbol{R}) \gtrless_{H_0}^{H_1} \xi. \tag{3.5}$$

We observe that, the likelihood ratio test is performed by processing the receiving vector to get the likelihood ratio and then comparing it with the threshold. In practical cases, the cost and the a priori probabilities may change, the threshold changes but the calculation of the likelihood ratio is not affected. Because the natural logarithm is a monotonic function, and both sides of (3.5) are positive, an equivalent test is

$$\ln \Phi(\boldsymbol{R}) \gtrless_{H_0}^{H_1} \ln \xi.$$

3.1.2 M Hypotheses

Now, we consider the case in which the choice is one of M hypotheses, $H_0, H_1, ..., H_{M-1}$. In this case, there are M^2 possible alternatives. The Bayes criterion assigns a cost to each alternative. We assign for each alternative the a priori probabilities $p_0, p_1, ..., p_{M-1}$ respectively and the aim is to minimize the risk.

Bayes Criterion

The cost of each case is defined as C_{ij}. The first subscript indicates that the ith hypothesis is chosen while the second subscript indicates that the jth hypothesis is true. The risk in this case defined as,

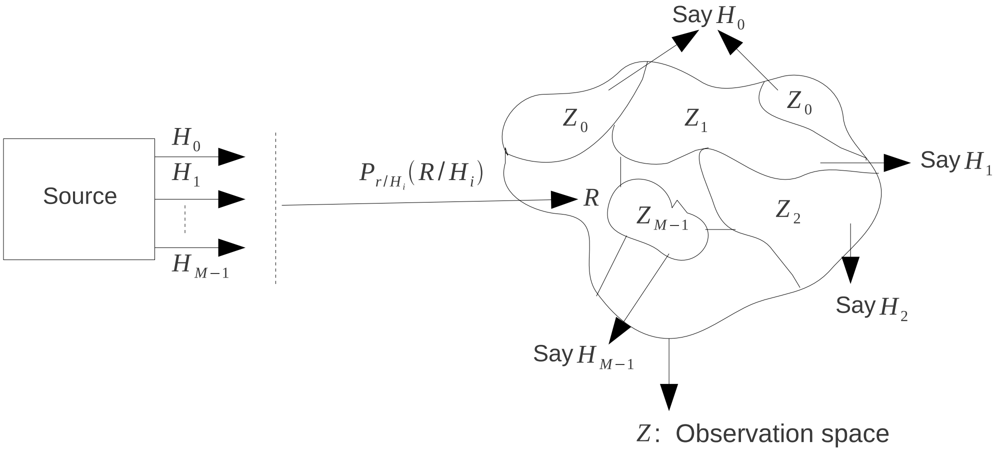


Figure 3.5: M hypotheses problem.

$$\mathcal{R} = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} p_j C_{ij} \int_{Z_i} p_{\boldsymbol{r}|H_j}(\boldsymbol{R}|H_j)\, d\boldsymbol{R},$$

where, Z_i is the region of observation space in which H_i is chosen and p_i is a priori property as shown in Figure 3.5. To verify our aim (optimum Bays test) we vary the Z_i in order to minimize $\mathcal{R}$. As an extension of the technique that is used in the binary case, the likelihood ratio in M-hypotheses as derived in [5] is

$$\Phi_i(\boldsymbol{R}) = \frac{p_{\boldsymbol{r}|H_i}(\boldsymbol{R}|H_i)}{p_{\boldsymbol{r}|H_0}(\boldsymbol{R}|H_0)}, i = 1, 2, ..., M - 1.$$

In the communication problems, it is common to have the costs,

$$C_{ii} = 0,$$

and,

$$C_{ij} = 1.$$

The notation is simplified as: consider the case in which $M = 3$, the Bayes test will be,

$$\Phi_1(\boldsymbol{R}) \gtrless^{H_1 or H_2}_{H_0 or H_2} \frac{p_0}{p_1},$$

$$\Phi_2(\boldsymbol{R}) \gtrless^{H_2 or H_1}_{H_0 or H_1} \frac{p_0}{p_2}.$$

When we transit to $\ln \Phi_1, \ln \Phi_2$ plan, the Bayes test equations will be,

$$\ln \Phi_1(\boldsymbol{R}) \gtrless^{H_1 or H_2}_{H_0 or H_2} \ln \frac{p_0}{p_1},$$

$$\ln \Phi_2(\boldsymbol{R}) \gtrless^{H_2 or H_1}_{H_0 or H_1} \ln \frac{p_0}{p_2}.$$

3.2 Estimation Theory

We considered in Section 3.1 the detection problem where the receiver receives a noisy signal and decides which hypothesis among all M possible hypotheses is true. As the simplest case which occurs in the binary decision, the receiver decides between the hypothesis H_0 and the alternative hypothesis H_1.

In this section, we discuss the parameter estimation problem in which the decision is made by deciding the true hypothesis. Some parameters associated with the signal may unknown. The problem is how to estimate those parameters. The problems in such scenario are called *estimation problems*.

A model of the general estimation problem is shown in Figure 3.6. The model has the following four components:

- **Parameter Space**: The output of the source is a parameter (or variable). We look at this output as a point in the parameter space.

- **Probabilistic Mapping from Parameter Space to Observation Space**: This is the probability rule that maps a selected value θ from the parameter space onto observation space.

- **Observation Space**: It is the set of all outcomes of the mapping of a parameter onto an observation. This will generally be a finite dimensional space. An observation is denoted by $\boldsymbol{R}$.

- **Estimation Rule**: We observe $\boldsymbol{R}$ then, we estimate the value of θ. This estimation is denoted by $\hat{\theta}(\boldsymbol{R})$. This mapping of the observation space into an estimate is called the *estimation rule*.

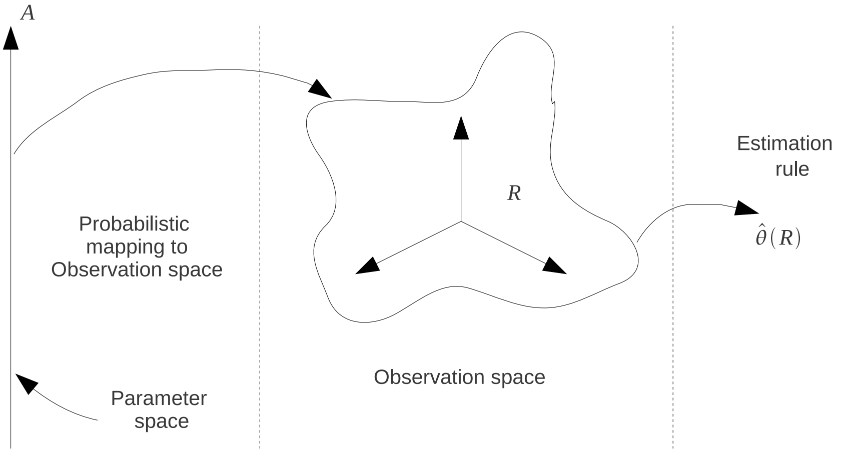


Figure 3.6: Estimation model.

Probabilistic Mapping from Parameter Space to Observation Space is known from the detection theory. The new features now are the parameter space and the estimation rule. Two cases arise in description of the parameter space. In the first case, the parameter is a *random variable*. It means that the behavior of this parameter is governed by the *probability density*. In the second case, we have *unknown quantity* (the parameter) but in this case, it is not a random variable.

3.2.1 Random Parameters: Bayes Estimation

In the Bayes estimation, the cost $C(\theta, \hat{\theta})$ is assigned to all pairs $(\theta, \hat{\theta}(\boldsymbol{R}))$. We consider the cost as a nonnegative real valued function of two random variables θ and $\hat{\theta}(\boldsymbol{R})$. The average value of the cost is known as the risk function that is defined as,

$$\mathcal{R} = E\left\{C[\theta, \hat{\theta}(\boldsymbol{R})]\right\},$$

where E is the expectation over the random variable θ and the observation variables. In order to obtain $\theta(\boldsymbol{R})$, we must minimize the risk function. The error between the estimate and the true value is defined as,

$$\theta_{\epsilon}(\boldsymbol{R}) = \hat{\theta}(\boldsymbol{R}) - \theta.$$

The single variable function $C(\theta_{\epsilon})$ is known as the cost function. We study three cases and their corresponding sketches which are shown in Figure 3.7.

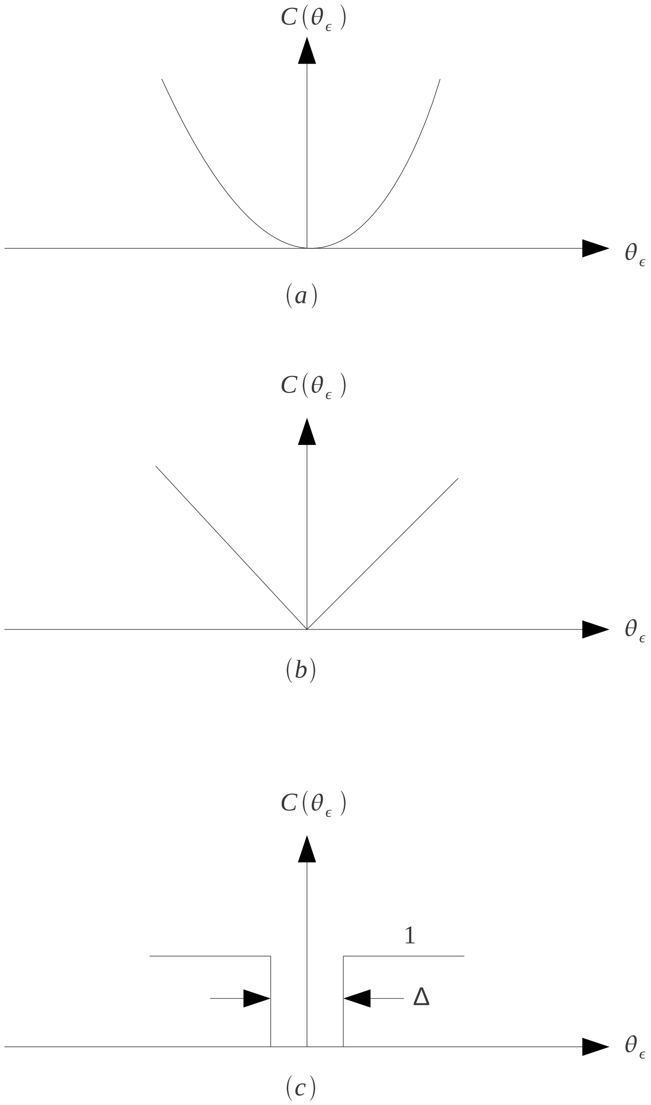


Figure 3.7: Cost functions.

In Figure 3.7a, the cost function is the square of error:

$$C(\theta_\epsilon) = \theta_\epsilon^2. \tag{3.6}$$

This cost is commonly referred to as the squared error cost function. In Figure 3.7b, the cost function is the absolute value of the error,

$$C(\theta_\epsilon) = |\theta_\epsilon|.$$

In Figure 3.7c, we assign a zero cost to all errors $-\Delta/2 < \theta_\epsilon < \Delta/2$. In other words, an error less than $\Delta/2$ in magnitude is as good as no error. If $\theta_\epsilon > \Delta/2$, we assign a uniform value:

$$C(\theta_\epsilon) = 0, \quad |\theta_\epsilon| \leqslant \frac{\Delta}{2},$$

$$C(\theta_\epsilon) = 1, \quad |\theta_\epsilon| \ > \frac{\Delta}{2}.$$

After we have defined the cost function and the a priori probability (denoted by $p_\theta(A)$), the expression of the risk can be written as,

$$\mathcal{R} = E\left\{C[\theta, \hat{\theta}(\boldsymbol{R})]\right\} = \int_{-\infty}^{\infty} dA \int_{-\infty}^{\infty} C[A, \hat{\theta}(\boldsymbol{R})] p_{\theta,\boldsymbol{r}}(A, \boldsymbol{R}) d\boldsymbol{R}. \tag{3.7}$$

For costs that are functions of one variable only, equation (3.7) becomes

$$\mathcal{R} = \int_{-\infty}^{\infty} dA \int_{-\infty}^{\infty} C[A - \hat{\theta}(\boldsymbol{R})] p_{\theta,\boldsymbol{r}}(A, \boldsymbol{R}) d\boldsymbol{R}. \tag{3.8}$$

Minimum Mean Square Error

Substituting (3.6) into (3.8), we get

$$\mathcal{R}_{ms} = \int_{-\infty}^{\infty} dA \int_{-\infty}^{\infty} d\boldsymbol{R}[A - \hat{\theta}(\boldsymbol{R})]^2 p_{\theta,\boldsymbol{r}}(A, \boldsymbol{R}), \tag{3.9}$$

where, $\mathcal{R}_{ms}$ is the risk for the *mean-square error*. We can write the joint density as,

$$p_{\theta,\boldsymbol{r}}(A, \boldsymbol{R}) = p_{\boldsymbol{r}}(\boldsymbol{R}) p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}).$$

Using this relation, (3.9) will be,

$$\mathcal{R}_{ms} = \int_{-\infty}^{\infty} d\boldsymbol{R} p_{\boldsymbol{r}}(\boldsymbol{R}) \int_{-\infty}^{\infty} dA[A - \hat{\theta}(\boldsymbol{R})]^2 p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}). \tag{3.10}$$

We can minimize $\mathcal{R}_{ms}$ by minimizing the inner integral because the inner integral and $p_{\boldsymbol{r}}(\boldsymbol{R})$ are non-negative. The way to find this estimate (denoted by $\hat{\theta}_{ms}(\boldsymbol{R})$) is differentiation of the inner integral with respect to $\hat{\theta}(\boldsymbol{R})$ and then equalize the result to zero, we get

$$\frac{d}{d\hat{\theta}} \textstyle\int_{-\infty}^{\infty} dA[A - \hat{\theta}(\boldsymbol{R})]^2 p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}) =$$
$$-2 \textstyle\int_{-\infty}^{\infty} A p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})\, dA + 2\hat{\theta}(\boldsymbol{R}) \textstyle\int_{-\infty}^{\infty} p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})\, dA.$$

We set the result equal to zero and observe that the second integral equals 1, we have

$$\hat{\theta}_{ms}(\boldsymbol{R}) = \int_{-\infty}^{\infty} dA \;\; A p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}). \tag{3.11}$$

The term in the right side of (3.11) is known as the *mean of the a posteriori density* (or the *conditional mean*). If $\hat{\theta}(\boldsymbol{R})$ in (3.10) is the conditional mean, the inner integral is just the *a posteriori variance*. So, the minimum of $\mathcal{R}_{ms}$ is considered as the average of the *conditional variance* over all observations $\boldsymbol{R}$.

Minimum Mean Absolute Value of Error Estimate

Our aim in this subsection is to find the Bayes estimate for the absolute value criterion in Figure 3.7*b*. To achieve this aim we first write

$$\mathcal{R}_{abs} = \int_{-\infty}^{\infty} d\boldsymbol{R} p_{\boldsymbol{r}}(\boldsymbol{R}) \int_{-\infty}^{\infty} dA \;\; |A - \hat{\theta}(\boldsymbol{R})| \;\; p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}).$$

To minimize the inner integral, we write

$$I(\boldsymbol{R}) = \int_{-\infty}^{\hat{\theta}(\boldsymbol{R})} dA[\hat{\theta}(\boldsymbol{R}) - A] p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}) + \int_{\hat{\theta}(\boldsymbol{R})}^{\infty} dA[A - \hat{\theta}(\boldsymbol{R})] p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}).$$

We differentiate $I(\boldsymbol{R})$ with respect to $\hat{\theta}(\boldsymbol{R})$ and set the result equal to zero, we get

$$\int_{-\infty}^{\hat{\theta}_{abs}(\boldsymbol{R})} dA \;\; p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}) = \int_{\hat{\theta}_{abs}(\boldsymbol{R})}^{\infty} dA \;\; p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}).$$

This is the definition of the *median of the a posteriori density.*

Maximum a Posteriori Estimation

The uniform cost function is the last criterion in Figure 3.7*c*. In this case, we express the risk as:

$$\mathcal{R}_{unf} = \int_{-\infty}^{\infty} d\boldsymbol{R} \; p_{\boldsymbol{r}}(\boldsymbol{R}) \left[1 - \int_{\hat{\theta}_{unf}(\boldsymbol{R}) - \Delta/2}^{\hat{\theta}_{unf}(\boldsymbol{R}) + \Delta/2} p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}) \, dA \right].$$

We have to maximize the inner integral in order to minimize this equation [50]. We consider Δ is an arbitrarily nonzero small number. The a posteriori density is shown in Figure 3.8.

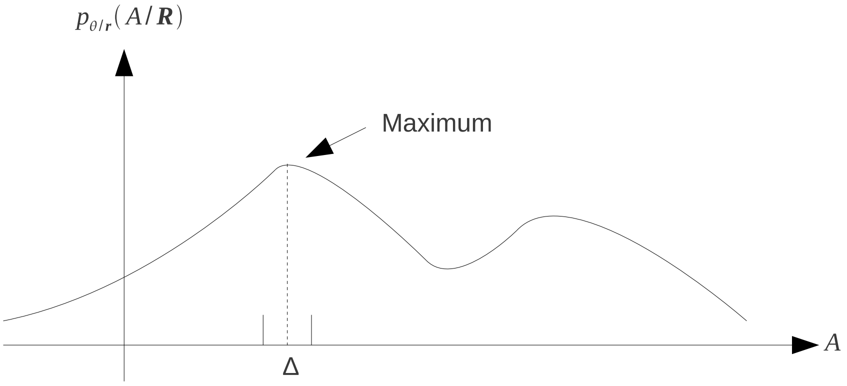


Figure 3.8: An a posteriori density.

The value of A at which the a posteriori density has its maximum is the best choice for $\hat{\theta}(\boldsymbol{R})$. The estimate in this case is denoted as $\hat{\theta}_{map}(\boldsymbol{R})$, *the maximum a posteriori estimate* (MAP). To find $\hat{\theta}_{map}(\boldsymbol{R})$, the location of the maximum of $p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})$ must be given. The logarithm is a monotone function, so we can find the location of the maximum of $\ln p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})$. If the maximum is interior to the range of A and $\ln p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})$ has a continuous first derivative then a necessary condition for a maximum is obtained using the differentiation of $\ln p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})$ with respect to A and equalizing the result to zero:

$$\frac{\partial \ln p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})}{\partial A}|_{A=\hat{\theta}(\boldsymbol{R})} = 0. \tag{3.12}$$

Equation (3.12) is named *MAP* equation. In each case we have to check to see if the solution is an absolute maximum. The expression $p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R})$ can be rewritten to separate the role of the priori knowledge and the observe vector $\boldsymbol{R}$:

$$p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}) = \frac{p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)p_{\theta}(A)}{p_{\boldsymbol{r}}(\boldsymbol{R})}.$$

Taking logarithms,

$$\ln p_{\theta|\boldsymbol{r}}(A|\boldsymbol{R}) = \ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A) + \ln p_{\theta}(A) - \ln p_{\boldsymbol{r}}(\boldsymbol{R}). \tag{3.13}$$

In MAP estimation, our interesting is only to find the value of A where the left-hand side is maximum. The last term on the right hand side of (3.13) is not a function of A, so we can consider just the function

$$l(A) = \ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A) + \ln p_{\theta}(A).$$

The first term gives the probabilistic dependence of $\boldsymbol{R}$ on A and the second describes a priori knowledge. The MAP equation can be written as

$$\frac{\partial l(A)}{\partial A}|_{A=\hat{\theta}(\boldsymbol{R})} = \frac{\partial \ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)}{\partial A}|_{A=\hat{\theta}(\boldsymbol{R})} + \frac{\partial \ln p_{\theta}(A)}{\partial A}|_{A=\hat{\theta}(\boldsymbol{R})} = 0. \tag{3.14}$$

3.2.2 Nonrandom Parameter Estimation

Assume that the parameter to be estimated is nonrandom, and we aim to design an estimation procedure. Using the Bayes estimation in this case fails to lead to useful results [84]. We must consider some other measures of the quality of the estimate.

The expectation of the estimate is the first measure of quality to be considered.

$$E[\hat{\theta}(\boldsymbol{R})] = \int_{-\infty}^{\infty} \hat{\theta}(\boldsymbol{R}) p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)\, d\boldsymbol{R}.$$

There are three possible values of the expectation that can be grouped into three classes

1. If $E[\hat{\theta}(\boldsymbol{R})] = A$, for all values of A, we say that the estimate is *unbiased.* In fact, this estimate means that the estimated average value equals the quantity we are trying to estimate.

2. If $E[\hat{\theta}(\boldsymbol{R})] = A + B$, where B is not a function of A, we say that the estimate has a *known bias.* By subtracting B from $E[\hat{\theta}(\boldsymbol{R})]$ we always obtain an unbiased estimate.

3. If $E[\hat{\theta}(\boldsymbol{R})] = A + B(A)$, we say that the estimate has an *unknown bias*. Because the bias depends on the unknown parameter, we can not simply subtract it out.

The expectation of an estimate is not very satisfactory since it can lead to large errors if the a posteriori density has a large second moment [50].

Maximum Likelihood Estimation

The *variance of estimation error* is the second measure of the quality. When we have a small variance, it indicates that we have a good estimate. The *maximum likelihood estimation* (ML) satisfies this condition. In this procedure, the aim is to maximize the *likelihood function* $p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)$, which is a function of A. We work with the logarithm, $\ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)$ which is denoted by *log likelihood function.* The value of A at which the likelihood function is maximum is the maximum likelihood estimate $\hat{\theta}_{ml}(\boldsymbol{R})$. If the maximum is within the range of A, and $\ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)$ has a continuous first derivative, then we obtain the necessary condition on $\hat{\theta}_{ml}(\boldsymbol{R})$ by differentiating $\hat{\theta}_{ml}(\boldsymbol{R})$ with respect to A and set the result equal to zero:

$$\frac{\partial \ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)}{\partial A}|_{A=\hat{\theta}_{ml}(\boldsymbol{R})} = 0. \tag{3.15}$$

Equation (3.15) is called *likelihood equation.* If we compare (3.14) and (3.15), then we see that ML estimate corresponds mathematically to the limiting case of MAP estimate in which the priori knowledge approaches zero.

In order to know how effective the ML is, we simply compute the bias and the variance but, this is difficult to do [50]. Instead, we can derive the lower bound on the variance on any unbiased estimate. Then we see how the variance of $\hat{\theta}_{ml}(\boldsymbol{R})$ can compared to this lower bound. We state without proof the *Cramer-Rao Inequality.* If $\hat{\theta}_{ml}(\boldsymbol{R})$ is any unbiased estimate of A, then

$$Var[\hat{\theta}_{ml}(\boldsymbol{R}) - A] \geq (E\left\{\left[\frac{\partial \ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)}{\partial A}\right]^2\right\})^{-1}. \tag{3.16}$$

Or, equivalently,

$$Var[\hat{\theta}_{ml}(\boldsymbol{R}) - A] \geq -E\left[\frac{\partial^2 \ln p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)}{\partial A^2}\right]^{-1}, \tag{3.17}$$

where the following conditions are assumed to be satisfied:

$$\frac{\partial p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)}{\partial A} \quad and \quad \frac{\partial^2 p_{\boldsymbol{r}|\theta}(\boldsymbol{R}|A)}{\partial^2 A},$$

exist and they are absolutely integrable. Inequalities (3.16) and (3.17) are referred to as *Cramer-Rao bound.* Cramer has proved that, the equality holds when $\hat{\theta}(\boldsymbol{R})$ is a sufficient statistic for the estimate of the parameter. The *efficient estimate* is any estimate satisfies the bound with an equality. The *Cramer-Rao Inequality* is proved in [50].

3.2.3 Multiple Parameter Estimation

May be, we want to estimate more than one parameter in many problems of interest. A good example of a parameter estimation is the communication application, the problem may be how to estimate arrival time, the amplitude, and a carrier frequency of a received signal. Therefore, the parameter estimation concepts will be extended to multiple parameters. If there exist k parameters, $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ..., \boldsymbol{\theta}^{(k)}$, a parameter vector $\boldsymbol{\theta}$ such that $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ..., \boldsymbol{\theta}^{(k)})^T$ in a K dimensional space can describe these parameters. As an extension of the results that were presented, we present Mean Square (MS), MAP and ML estimations.

The mean square estimation is given by

$$\hat{\boldsymbol{\theta}}_{ms}(\boldsymbol{R}) = \int_{-\infty}^{\infty} \boldsymbol{A} p_{\boldsymbol{\theta}|\boldsymbol{r}}(\boldsymbol{A}|\boldsymbol{R})\, d\boldsymbol{A}.$$

In MAP estimation we have to find the value of $\boldsymbol{A}$ that maximizes $p_{\boldsymbol{\theta}|\boldsymbol{r}}(\boldsymbol{A}|\boldsymbol{R})$. If the maximum is interior and $\frac{\partial \ln p_{\boldsymbol{\theta}}|\boldsymbol{r}(\boldsymbol{A}|\boldsymbol{R})}{\partial \boldsymbol{A}^{(i)}}$ exists at the maximum then we obtain a necessary condition from the MAP equations. We take the logarithm of $p_{\boldsymbol{\theta}|\boldsymbol{r}}(\boldsymbol{A}|\boldsymbol{R})$, differentiate with respect to each parameter $\boldsymbol{A}^{(i)}, i = 1, ..., k$, and set the result equal to zero. The result is a set of K simultaneous equations:

$$\frac{\partial \ln p_{\boldsymbol{\theta}}|\boldsymbol{r}(\boldsymbol{A}|\boldsymbol{R})}{\partial \boldsymbol{A}^{(i)}}|_{\boldsymbol{A}=\hat{\boldsymbol{\theta}}_{map}(\boldsymbol{R})} = 0, \quad i = 1, 2, ..., k. \tag{3.18}$$

We can write (3.18) as a single vector equation,

$$\boldsymbol{\nabla}_{\boldsymbol{A}} \left[\ln p_{\boldsymbol{\theta}}|\boldsymbol{r}(\boldsymbol{A}|\boldsymbol{R})\right] |_{\boldsymbol{A}=\hat{\boldsymbol{\theta}}_{map}(\boldsymbol{R})} = 0,$$

where,

$$\boldsymbol{\nabla}_{\boldsymbol{A}} = [\frac{\partial}{\partial \boldsymbol{A}^{(1)}} \ \frac{\partial}{\partial \boldsymbol{A}^{(2)}} \ ... \frac{\partial}{\partial \boldsymbol{A}^{(K)}} \ ]^T.$$

Similarly, for ML estimates we have to find the value of $\boldsymbol{A}$ that maximizes $p_{\boldsymbol{r}|\boldsymbol{\theta}}(\boldsymbol{R}|\boldsymbol{A})$. If the maximum is interior and $\frac{\partial \ln p_{\boldsymbol{r}|\boldsymbol{\theta}}(\boldsymbol{R}|\boldsymbol{A})}{\partial \boldsymbol{A}^{(i)}}$ exists at the maximum then we obtain a necessary condition from the likelihood equations:

$$\boldsymbol{\nabla}_{\boldsymbol{A}} \left[\ln p_{\boldsymbol{r}|\boldsymbol{\theta}}(\boldsymbol{R}|\boldsymbol{A})\right] |_{\boldsymbol{A}=\hat{\boldsymbol{\theta}}_{ml}(\boldsymbol{R})} = 0.$$

3.3 Digital Modulation

In this section, we present some details about digital modulations, especially Phase Shift Keying (PSK) modulation that is used in this context. Users exchange information by the transmission of signals. The values of data are represented by the signal parameters. In digital transmission, the most important types of signals are the periodic signals. The *sine waves* as carriers can be considered in general form as

$$\boldsymbol{s}(t) = a_t \sin(2\pi f_t t + \phi_t).$$

This signal has three parameters, *amplitude* a_t, *frequency* f_t, and *phase shift* ϕ. The amplitude changes over the time, thus it is represented by a_t. The frequency f_t expresses the periodicity of the signal with the period $T = 1/f_t$. (We can denote the frequency by ω instead of $2\pi f$). The frequency f may also change over the time, thus it is represented by f_t. Finally, the phase shift determines the shift of the signal relative to the same signal without a shift.

In digital modulations, data ('0' and '1') are required to be transformed into an analog signal. This kind of modulation is important when the digital data has to be transmitted over a medium that only allows the analog transmission. Digital transmission is used for example in wired local area networks or within a computer [34, 90]. In wireless networks, however, digital transmission can not be used. Here, the binary bit-stream has to be first converted into an analog signal. We are not interesting in this work with analog modulations, but the reader is referred to [34, 89] for more details about these analog modulation schemes.

3.3.1 Phase Shift Keying

In *Phase Shift Keying (PSK)*, data are represented by shifting in the phase of signals.

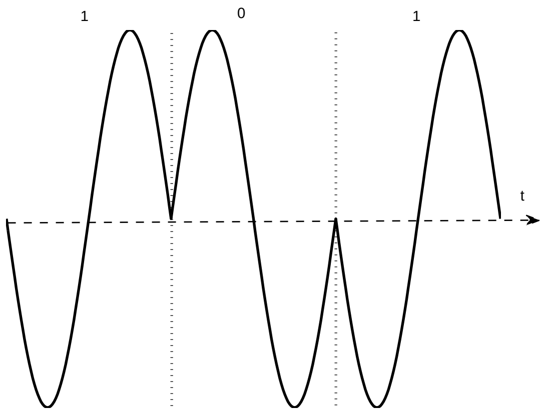


Figure 3.9: Phase Shift Keying (PSK).

Figure 3.9 shows a phase shift of π as '0' follows '1' or as '1' follows '0'. This simple scheme shifts the phase by π each time the value of data changes. It is also called Binary PSK (BPSK).

Quadrature Phase Shift Keying (QPSK)

We described the simple PSK scheme (BPSK) that can be improved in many ways. The basic BPSK scheme only uses one possible phase shift of π.

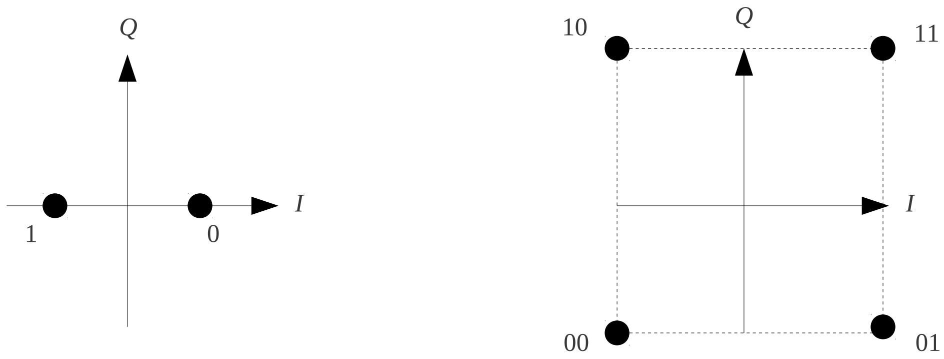


Figure 3.10: BPSK and QPSK in the phase domain.

The left side of Figure 3.10 shows BPSK in the phase domain. The right side of Figure 3.10 shows *Quadrature PSK (QPSK)*, one of the most common PSK schemes. In this scheme, we achieve higher bit rates for the same bandwidth by coding two bits using one phase shift. Alternatively, one can reduce the bandwidth and still achieves the same bit rates as for BPSK.

PSK schemes including QPSK can be realized in two ways. The phase shift can always be relative to a *reference signal* (with the same frequency). If this scheme is used, a phase shift of '0' means that the signal is in phase with the reference signal. A QPSK signal exhibits a phase shift of $\pi/4$ for the data '11', $3\pi/4$ for '10', $5\pi/4$ for '00', and $7\pi/4$ for '01' with all phases shifts being relative to the reference signal. To reconstruct data, the receiver compares the incoming signal with the reference signal.

It is worth mentioning that other modulation schemes yield different discrete constellation sets, but the presented methods in this thesis can be applied to all possible constellation sets.

3.4 Linear Detection

Let us describe the channel model that is used in this thesis. The $\acute{n} \times 1$ vector $\acute{\boldsymbol{x}}$ represents the data to be transmitted through the channel, and it is chosen from a finite equiprobable set S. The channel interference is modeled as linear interference, represented the multiplication of $\acute{\boldsymbol{x}}$ with a $\acute{m} \times \acute{n}$ channel matrix $\acute{\boldsymbol{H}}$. The channel noise is composed of the superposition of many independent actions. Using the central limit theorem [42], the noise can be modeled as a zero mean complex-valued, Additive White Gaussian Noise (AWGN) vector $\acute{\boldsymbol{n}}$. We can obtain the $\acute{m} \times 1$ vector $\acute{\boldsymbol{r}}$ at the receiver as

$$\acute{\boldsymbol{r}} = \acute{\boldsymbol{H}}\acute{\boldsymbol{x}} + \acute{\boldsymbol{n}}. \tag{3.19}$$

We are concerned with detection at the receiver of the transmitted vector $\acute{\boldsymbol{x}}$ based on knowledge of $\acute{\boldsymbol{r}}$, $\acute{\boldsymbol{H}}$ and the statistics of $\acute{\boldsymbol{n}}$. We assume that $\acute{\boldsymbol{H}}$ and the statistics of $\acute{\boldsymbol{n}}$ are known at the receiver.

The equivalent real-valued transmission model is much useful to deal than the complex model (3.19). By separating the real and the imaginary parts in (3.19), we can equivalently write [94],

$$\begin{bmatrix} \Re(\acute{\boldsymbol{r}}) \\ \Im(\acute{\boldsymbol{r}}) \end{bmatrix} = \begin{bmatrix} \Re(\acute{\boldsymbol{H}}) & -\Im(\acute{\boldsymbol{H}}) \\ \Im(\acute{\boldsymbol{H}}) & \Re(\acute{\boldsymbol{H}}) \end{bmatrix} \begin{bmatrix} \Re(\acute{\boldsymbol{x}}) \\ \Im(\acute{\boldsymbol{x}}) \end{bmatrix} + \begin{bmatrix} \Re(\acute{\boldsymbol{n}}) \\ \Im(\acute{\boldsymbol{n}}) \end{bmatrix},$$

which gives an equivalent $m \times n$-dimensional real model of the form,

$$\boldsymbol{r} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}, \tag{3.20}$$

where, $m = 2\acute{m}$ and $n = 2\acute{n}$ with the same definition of $\boldsymbol{r}$, $\boldsymbol{H}$, $\boldsymbol{x}$, and $\boldsymbol{n}$.

Consider a system model as described in (3.20). The receiver has to detect the transmitted signal $\boldsymbol{x}$ from $\boldsymbol{r} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}$, i.e., it constructs an estimate $\hat{\boldsymbol{x}}$, given $\boldsymbol{r}$ and $\boldsymbol{H}$. The diagram of Figure 3.11 describes this operation. We assume that the detector, i.e., the receiver in the transmission system has perfect knowledge of the channel matrix $\boldsymbol{H}$. The data symbol vector $\boldsymbol{x} = [\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, ..., \boldsymbol{x}^{(n)}]^T$ to be transmitted is selected from the constellation set S^{n}, with

$$S = \left\{ e^{j\alpha_i} : \alpha_i = 2\pi i/M, \forall i = 1, ..., M \right\},$$

where M is the order of modulation, i.e. $M = 2$ for BPSK, $M = 4$ for QPSK, and so on.

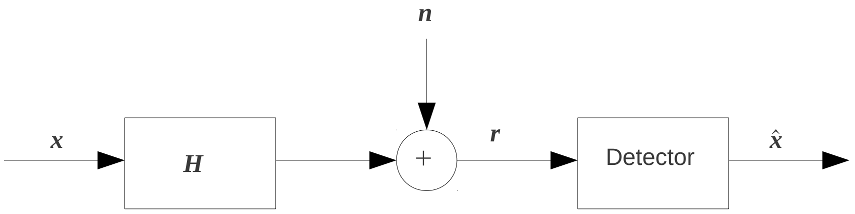


Figure 3.11: General Detection Setup.

In this section, various optimum and sub-optimum detectors are presented using theories and knowledges from chapter 2 and previously discussions in this chapter.

3.4.1 Optimum Detector

We discuss the optimum decision rule on the observation vector $\boldsymbol{r}$. Our objective in this discussion is making a decision on the transmitted signal based on the observation vector $\boldsymbol{r}$ such that the probability of correct decision is maximized. This operation as we discussed before is named a signal detection. To achieve this aim, we consider a decision rule based on the computation of the *a posteriori probabilities* that defined previously as,

$$p(\boldsymbol{x}_m|\boldsymbol{r}), m = 1, 2, ..., M.$$

Corresponding to the maximum of the set of a posteriori probabilities $\{p(\boldsymbol{x}_m|\boldsymbol{r})\}$, the signal is selected. This is done by maximizing the probability of a correct decision and, hence, minimizing the probability of error. This decision criterion as we discussed is called the *maximum a posteriori probability (MAP)* criterion.

We can express the a posteriori probabilities using Bayes' rule,

$$p(\boldsymbol{x}_m|\boldsymbol{r}) = \frac{p(\boldsymbol{r}|\boldsymbol{x}_m)p(\boldsymbol{x}_m)}{p(\boldsymbol{r})}, \tag{3.21}$$

where, $p(\boldsymbol{r}|\boldsymbol{x}_m)$ is the conditional pdf of the observed vector $\boldsymbol{r}$ given $\boldsymbol{x}_m$ and $p(\boldsymbol{x}_m)$ is the *a priori probability* of the mth signal that transmitted.

We can also express the denominator of (3.21) as,

$$p(\boldsymbol{r}) = \sum_{m=1}^{M} p(\boldsymbol{r}|\boldsymbol{x}_m)p(\boldsymbol{x}_m). \tag{3.22}$$

We observe from (3.21) and (3.22) that the computation of the a posterior probabilities $p(\boldsymbol{x}_m|\boldsymbol{r})$ requires knowledge of the a *priori probabilities* $p(\boldsymbol{x}_m)$ and the conditional pdfs $p(\boldsymbol{r}|\boldsymbol{x}_m)$ for $m = 1, ..., M$.

In the (MAP) criterion some simplification occurs when the M signals are equally probable a priori, i.e., $p(\boldsymbol{x}_m) = 1/M$ for all M. In addition, the dominator in (3.21) is independent of which signal is transmitted. Consequently, the decision rule based on how to find a signal that maximizes $p(\boldsymbol{x}_m|\boldsymbol{r})$ is the same as how to find the signal that maximizes $p(\boldsymbol{r}|\boldsymbol{x}_m)$.

From the meaning of the *Likelihood function*, the conditional pdf $p(\boldsymbol{r}|\boldsymbol{x}_m)$ or any monotonic function of it, is called the *Likelihood function*. The *maximum-likelihood* (ML) *criterion* is the decision criterion based on the maximum of $p(\boldsymbol{r}|\boldsymbol{x}_m)$ over M signals. We observe that when the a *priori probabilities* $p(\boldsymbol{x}_m)$ are all equal, i.e., the signals $\boldsymbol{x}_m$ for all $m = 1, ..., M$ are equiprobable, a detector based on the MAP criterion and one that is based on the (ML) criterion make the same decisions. In case of AWGN channel, the likelihood function $p(\boldsymbol{r}|\boldsymbol{x}_m)$ is given by

$$p(\boldsymbol{r}|\boldsymbol{x}_m) = \frac{1}{(\pi\sigma_n^2)^{n/2}} exp\left[-\sum_{k=1}^{n} \frac{(\boldsymbol{r}^{(k)} - \boldsymbol{x}_m^{(k)})^2}{\sigma_n^2}\right],$$

where, σ_n^2 is the noise power spectral density [81]. We may work with the logarithm of $p(\boldsymbol{r}|\boldsymbol{x}_m)$ to simplify the computations. Thus,

$$\ln p(\boldsymbol{r}|\boldsymbol{x}_m) = -\frac{1}{2} n \ln(\pi\sigma_n^2) - \frac{1}{\sigma_n^2} \sum_{k=1}^{n} \left(\boldsymbol{r}^{(k)} - \boldsymbol{x}_m^{(k)}\right)^2.$$

The maximum of $\ln p(\boldsymbol{r}|\boldsymbol{x}_m)$ over $\boldsymbol{x}_m$ is equivalent to find the signal $\boldsymbol{x}_m$ that minimizes the Euclidean distance

$$d(\boldsymbol{r}, \boldsymbol{x}_m) = \sum_{k=1}^{n} \left(\boldsymbol{r}^{(k)} - \boldsymbol{x}_m^{(k)}\right)^2.$$

We call $d(\boldsymbol{r}, \boldsymbol{x}_m), i = 1, 2, ..., M$, the *distance metrics*. So, for the AWGA channel, the decision rule based on the ML criterion reduces to find the signal $\boldsymbol{x}_m$ that is closest in distance to the received signal vector $\boldsymbol{r}$. We can write this problem in the form

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in S^n} \|\boldsymbol{r} - \boldsymbol{H}\boldsymbol{x}\|_2^2 . \tag{3.23}$$

The minimization of (3.23) requires the comparison of 2^n differences [99], so its complexity is exponential in n. In fact, the least-squares problem in (3.23) has been shown also in [99] to be nondeterministic polynomial-time hard (NP-hard).

3.4.2 Some Well-Known Detectors

The high complexity of the ML detector has invariably precluded its use in practice, so lower-complexity detectors that provide exact and approximate solutions to (3.23) are used. Among these detectors, *Branch and Bound*, which is a general discrete search method [52]. In [56], an optimal algorithm based on the branch and bound method with an iterative bound update was proposed. Another way of detection named, *Sphere Decoding* algorithm was first introduced in [28] and it has been used in space time block codes [18, 62]. Sphere decoding has been first applied in the context of communication in [101] and it is also used in the context of multi-carrier CDMA systems in [14]. A Generalized Sphere Decoder specially adapted to multiple antenna system has been proposed in [17]. In the following, we will review the linear detectors that are more related to our work.

Least Squares Detector

The ML problem in (3.23) can be equivalently written as

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in S^n} \boldsymbol{x}^T\boldsymbol{H}^H\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{r}^T\boldsymbol{H}\boldsymbol{x}. \tag{3.24}$$

Problem (3.24) has an objective function $f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{H}^H\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{r}^T\boldsymbol{H}\boldsymbol{x}$ and the constraint $\boldsymbol{x} \in S^n$. The constraint is discrete, so problem (3.24) is not convex (see chapter 2).

To use the benefits of convex optimization, we relax the constraint set to be the whole space $\mathbb{R}^n$. Problem (3.24) takes the form,

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x} \in \mathbb{R}^n} \boldsymbol{x}^T \boldsymbol{H}^H \boldsymbol{H} \boldsymbol{x} - 2\boldsymbol{r}^T \boldsymbol{H} \boldsymbol{x}. \tag{3.25}$$

Problem (3.25) is an unconstrained convex optimization problem, which enables us to apply the following theorem [11].

Theorem 3.1 *Suppose that the objective function f in an unconstrained convex optimization problem is differentiable, so the well-known necessary and sufficient optimality condition is*

$$\boldsymbol{\nabla} f = 0. \tag{3.26}$$

Applying condition (3.26) to problem (3.25), the necessary and sufficient optimality conditions give the solution

$$\boldsymbol{x}^*_{LS} = \left(\boldsymbol{H}^H \boldsymbol{H}\right)^{-1} \boldsymbol{H}^H \boldsymbol{r},$$

which is the well-known least squares solution.

Minimum Mean Squared Error (MMSE) Detector

When the noise power σ_n^2 is known, using the same relaxation (the whole space) we get the minimum mean square error solution

$$\boldsymbol{x}^*_{MMSE} = \left(\boldsymbol{H}^H \boldsymbol{H} + \sigma_n^2 \boldsymbol{I}\right)^{-1} \boldsymbol{H}^H \boldsymbol{r}. \tag{3.27}$$

Generalized MMSE Detector

One of the convex optimization applications is a generalized minimum mean squared error detection [112]. We will discuss in details about GMMSE detector in the next chapter. Here, we only present the form of GMMSE solution which is,

$$\boldsymbol{x}^*_{GMMSE} = \left(\boldsymbol{H}^H \boldsymbol{H} + \delta^* \boldsymbol{I}\right)^{-1} \boldsymbol{H}^H \boldsymbol{r},$$

where, δ^* is a kind of noise power estimation.

The Semidefinite Relaxation Detector

The semidefinite relaxation (SDR) is a vital convex optimization tool that solves many engineering problems. In the field of signal processing and communications, SDR was introduced in 2000. SDR is known as an efficient high performance approach in MIMO detection [47, 63, 64, 65, 69, 91, 93]. It is also efficiently used for blind MIMO detection [66]. SDR approximation accuracies relative to the ML have been investigated in MIMO detection [48].

The optimization problem of (3.23) for BPSK constellations can be written as

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x} \in \{-1,+1\}^n} \|\boldsymbol{r} - \boldsymbol{H}\boldsymbol{x}\|_2^2 ,$$

which is equivalently be obtained through

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x} \in \{-1,+1\}^n} \boldsymbol{x}^T \boldsymbol{H}^H \boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{r}^T \boldsymbol{H}\boldsymbol{x}. \tag{3.28}$$

This problem has a computational complexity which increases exponentially with n.

In order to use the solution of semidefinite programming, we relax problem (3.28) into a convex optimization one by reformulating it as [37]

$$\hat{\boldsymbol{d}} = \arg \min_{\boldsymbol{d} \in \{-1,+1\}^{n+1}} \boldsymbol{d}^T \boldsymbol{L}\boldsymbol{d}, \tag{3.29}$$

such that,

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{H}^H \boldsymbol{H} & -\boldsymbol{H}^H \boldsymbol{r} \\ -\boldsymbol{r}\boldsymbol{H} & \boldsymbol{0} \end{bmatrix} .$$

For $\boldsymbol{d} \in \{-1,+1\}^{n+1}$ the matrix $\boldsymbol{D} = \boldsymbol{d}\boldsymbol{d}^T$ is positive semidefinite, its diagonal entries are equal to 1, and it is of rank one [37]. Let $\boldsymbol{D}$ be a matrix which satisfies these three characteristic properties, then we can rewrite (3.29) as

$$\hat{\boldsymbol{D}} = \arg \min_{\boldsymbol{D}} \boldsymbol{L}\boldsymbol{D}, diag(\boldsymbol{D}) = \boldsymbol{e}_1, rank(\boldsymbol{D}) = 1, \boldsymbol{D} \succeq 0, \tag{3.30}$$

where $\boldsymbol{e}_1$ is an all ones vector of length $n+1$. Dropping the rank one constraint yields a convex optimization problem which is the basic semidefinite relaxation of (3.30) [37],

$$\hat{\boldsymbol{D}}_1 = \arg \min_{\boldsymbol{D}} \boldsymbol{L}\boldsymbol{D}, diag(\boldsymbol{D}) = \boldsymbol{e}_1, \boldsymbol{D} \succeq 0. \tag{3.31}$$

This form is known as a semidefinite program in the matrix variable $\boldsymbol{D}$, because it is a linear problem in $\boldsymbol{D}$ with the additional semidefinite constraint $\boldsymbol{D} \succeq 0$.

A semidefinite program (3.31) can be solved by employing the primal-dual path-following algorithm [38] as a basic optimization tool. Then the solution of the original problem (3.29) is the selection of $\hat{\boldsymbol{d}}$ to be the sign of the eigenvector corresponding to the largest eigenvalue of $\boldsymbol{D}_1$.

4 Structured GMMSE Detector

The Maximum Likelihood (ML) detection problem is equivalent to the problem of optimizing a quadratic function over the corners of a hypercube [100]. Unfortunately this problem is in general non-deterministic polynomial hard (NP-hard) [99]. This observation resulted in the development of many receivers that have reasonable complexity with near-optimum performance [15, 25, 67, 93, 98], e.g., the well-known Least Squares (LS) and Minimum Mean Squared Error (MMSE) detectors as the most simple cases [61, 67].

The quadratic optimization problem is a discrete optimization problem. It is usually computationally demanding to provide the optimum solution. The general approach is to approximate the solution by working on an easier problem that can be efficiently solved. The easier problem to be solved is a relaxation of the original problem. The solution of the relaxed problem is then mapped to the solution set of the original problem. One good relaxation of these kinds of problems is the convex optimization [58, 60, 112]. Recently convex programming has been successfully employed to convert the discrete optimization problems into continuous ones [11]. Generalized Minimum Mean Squared Error (GMMSE) detector is one important detector that uses convex programming to solve the detection problem using unconstrained gradient descent algorithm [112]. The GMMSE solution has the form of MMSE solution as shown in equation (3.27), but it does not require the knowledge of the ambient noise power level. Thus, it can be used in scenarios where adapted or blind adaptive detection is not suitable, for instance when the channel is changing rapidly, and the ambient noise power is unknown. The disadvantage of the GMMSE detector is the higher computational complexity compared to the MMSE detector.

In this chapter, we introduce a new detector that is a structured form of GMMSE detector. This detector is named *Structured GMMSE detector* which keeps the GMMSE's performance, but it has a lower computational complexity than MMSE. First the banded Toeplitz structure of the channel convolution matrix is taken into consideration. This banded Toeplitz matrix is approximated by a circular matrix in order to significantly reduce the computational complexity. We also analyze the performance of the GMMSE detector and its circular approximation. The noise enhancement of these two cases and thus the quality of the estimates which depends on the matrix condition number is analyzed.

Furthermore, simulation results for different types of channels using BPSK constellation are given.

4.1 Detection Problem and its Relaxations

Consider the system model (3.20) as presented in Section 3.4 as

$$\boldsymbol{r} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n},$$

where, the transmitted symbols $\boldsymbol{x} \in \mathbb{R}^n$ are drawn from Binary Phase Shift Keying (BPSK) constellation, i.e. $\boldsymbol{x} \in \{-1, +1\}^n$.

Under the white Gaussian noise assumption the Maximum Likelihood (ML) detector of $\boldsymbol{x}$ is given by

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x} \in \{-1,+1\}^n} \|\boldsymbol{r} - \boldsymbol{H}\boldsymbol{x}\|_2^2 ,$$

and it can be equivalently written as

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x} \in \{-1,+1\}^n} \boldsymbol{x}^H \boldsymbol{H}^H \boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{r}^H \boldsymbol{H}\boldsymbol{x}. \tag{4.1}$$

Substituting the value of the matched filter output

$$\boldsymbol{y} = \boldsymbol{H}^H \boldsymbol{r}$$

into (4.1), we get

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x} \in \{-1,+1\}^n} \boldsymbol{x}^H \boldsymbol{H}^H \boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^H \boldsymbol{x}. \tag{4.2}$$

Problem (4.2) is NP-hard and solving it by exhaustive search yields a complexity that grows with 2^n [99]. This makes computationally less complex solutions of (4.2) interesting.

We use the advantages of convex programming as an important mathematical optimization tool to solve problem (4.2) by relaxing its constraint set. The constraint set $\boldsymbol{x} \in \{-1, +1\}^n$ that contains only the corners of the unit hypercube is not a convex set. Therefore, we relax this constraint set into a convex set.

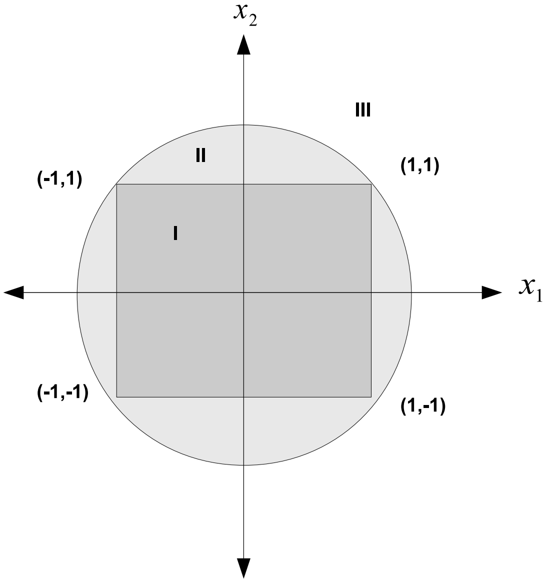


Figure 4.1: Convex relaxations.

Figure 4.1 shows the relaxed constraint sets for $n = 2$ taking into account that the original problem contains only the corners of the unit hypercube. Three relaxations are considered:

- Relaxation of the constraint set to the whole unit hypercube (region I).
- Relaxation of the constraint set to the sphere which covers the unit hypercube (region $I + II$).
- The relaxation to the whole space (region $I + II + III$).

Region I is the constraint set of the *soft interference canceler detector* [76, 112, 113] which is not in our interest. The second relaxation is region $I + II$, which yields the GMMSE solution as we will discuss in the following section. Region $I+II+III$ means that the problem has no constraints and the resulting solution is the MMSE solution as we discussed in Section 3.4.2. In general, the solution in each case can be mapped to the feasible set of the original problem by taking the sign of each component of the relaxed solution vector.

4.2 Generalized MMSE Detector

The constraint on each $\boldsymbol{x}^{(i)} \in \{-1, 1\}$ is equivalent to $(\boldsymbol{x}^{(i)})^2 = 1$ which implies $\boldsymbol{x}^T\boldsymbol{x} = n$. If we relax the constraint set in problem (4.2) to be the sphere which contains the unite hypercube, (region $I+II$) then the detection problem takes the form

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x}^H\boldsymbol{x} \leq n} \boldsymbol{x}^H\boldsymbol{H}^H\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x}. \tag{4.3}$$

Since problem (4.3) has a convex objective function,

$$f(\boldsymbol{x}) = \boldsymbol{x}^H\boldsymbol{H}^H\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x},$$

(because, it satisfies the second order condition (2.2)) over a convex constraint set $\boldsymbol{x}^H\boldsymbol{x} \leq n$, it is a convex optimization problem and it has a unique minimum [10, 11, 83]. The convex duality theorem guarantees that no duality gap exists and one can solve for the dual problem instead [55, 75].

As discussed for the duality problem in Section 2.4, we can express the Lagrange dual function of (4.3) as

$$\mathcal{L}(\boldsymbol{x}, \delta) = \boldsymbol{x}^H\boldsymbol{H}^H\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x} + \delta\left(\boldsymbol{x}^H\boldsymbol{x} - n\right), \tag{4.4}$$

where, $\delta \in \mathbb{R}$ is the Lagrange multiplier associated with the single constraint $\boldsymbol{x}^H\boldsymbol{x} \leq n$. Problem (4.4) is minimized over $\boldsymbol{x}$ and maximized over $\delta \geq 0$. Solving for $\boldsymbol{x}$ in terms of δ we get

$$\boldsymbol{x} = \left(\boldsymbol{H}^H\boldsymbol{H} + \delta\boldsymbol{I}\right)^{-1}\boldsymbol{y}.$$

Substituting back into (4.4), we obtain the dual problem,

$$\max_{\delta \geq 0} -\boldsymbol{y}^H\left(\boldsymbol{H}^H\boldsymbol{H} + \delta\boldsymbol{I}\right)^{-1}\boldsymbol{y} - \delta n, \tag{4.5}$$

which is a one-dimensional optimization problem. Problem (4.5) can be solved by different iterative algorithms [35]. A simple unconstrained gradient descent algorithm (see Section 2.6) is given by

$$\bar{\delta}\,(t+1) = \bar{\delta}\,(t) + \mu \left(\boldsymbol{y}^H \left(\boldsymbol{H}^H \boldsymbol{H} + \bar{\delta}\,(t)\,\boldsymbol{I} \right)^{-2} \boldsymbol{y} - n \right). \tag{4.6}$$

It converges to $\bar{\delta}$ for a reasonable choice of the step size μ. The solution of (4.5) is given by $\delta^* = max(0, \bar{\delta})$. Then, the unique minimizer of (4.3) is

$$\boldsymbol{x}^*_{GMMSE} = \left(\boldsymbol{H}^H \boldsymbol{H} + \delta^* \boldsymbol{I} \right)^{-1} \boldsymbol{y}. \tag{4.7}$$

When $\delta^* = \sigma_n^2$, the GMMSE detector reduces to the MMSE detector. GMMSE does not require the knowledge of the noise power σ^2 if training or blind adaptation is not desired [40]. However, GMMSE detector as we mentioned has the disadvantage that it requires a higher computational complexity than MMSE detector. The existing of this disadvantage makes it reasonable to think about a way to reduce this computational complexity.

4.3 Structured Problem

Many important problems in engineering can be reduced to matrix problems as we have previously seen. Moreover, different applications can introduce a special structure of the corresponding matrices, such that their entries can be presented by a certain compact form. These kinds of structures allow us to obtain elegant solutions to mathematical problems and also to design more efficient algorithms for a variety of applied engineering problems. The most important advantage of making use of the structure of the matrices is reducing the computational complexity [79].

In this section we use the circular approximation of the banded Toeplitz channel matrix $\boldsymbol{H}$ [85, 102] to reduce the computational complexity of GMMSE detector. Before we describe the method, we present some definitions and properties of two important kinds of structure.

4.3.1 Toeplitz Structure

A *Toeplitz matrix* or a *diagonal-constant matrix* is a matrix in which each descending diagonal from left to right is constant. In general, a Toeplitz matrix is an $n \times n$ matrix,

$$T = \left[t_{i,j} : t_{i,j} = \boldsymbol{t}^{(i-j)}, i, j = \{0, 1, ..., n-1\} \right],$$

which takes the form,

$$\boldsymbol{T} = \begin{pmatrix} \boldsymbol{t}^{(0)} & \boldsymbol{t}^{(-1)} & \boldsymbol{t}^{(-2)} & \cdots & \boldsymbol{t}^{(-(n-1))} \\ \boldsymbol{t}^{(1)} & \boldsymbol{t}^{(0)} & \boldsymbol{t}^{(-1)} & & \\ \boldsymbol{t}^{(2)} & \boldsymbol{t}^{(1)} & \boldsymbol{t}^{(0)} & & \vdots \\ \vdots & & & \ddots & \\ \boldsymbol{t}^{(n-1)} & & & \cdots & \boldsymbol{t}^{(0)} \end{pmatrix}.$$

The $m \times n$ matrix is said to be *rectangular Toeplitz* if their entries $t_{i,j} = \boldsymbol{t}^{(i-j)}$ with $(1 \leqslant i \leqslant m, 1 \leqslant j \leqslant n, n \leqslant m)$. There are numerous other applications for this kind of structure in mathematics, information theory, estimation theory, *etc.* The most common and complete reference that discuss how to use the Toeplitz structure in these applications is found in [33].

4.3.2 Circular Structure

Circular matrices are used both to approximate and explain the behavior of Toeplitz matrices. A circular matrix $\boldsymbol{C}$ is one has the form

$$\boldsymbol{C} = \begin{pmatrix} \boldsymbol{c}^{(1)} & \boldsymbol{c}^{(n)} & \boldsymbol{c}^{(n-1)} & \cdots & \boldsymbol{c}^{(2)} \\ \boldsymbol{c}^{(2)} & \boldsymbol{c}^{(1)} & \boldsymbol{c}^{(n)} & & \\ & \boldsymbol{c}^{(2)} & \boldsymbol{c}^{(1)} & & \vdots \\ \vdots & & & \ddots & \\ \boldsymbol{c}^{(n)} & & \cdots & \boldsymbol{c}^{(2)} & \boldsymbol{c}^{(1)} \end{pmatrix}. \tag{4.8}$$

Each row of $\boldsymbol{C}$ is a cyclic shift of the row above it. The matrix $\boldsymbol{C}$ is itself a special type of Toeplitz matrix. In [21, 32, 51], the reader can find the properties of Toeplitz and circular matrices.

In the following, we will use the characteristics of these kinds of matrices to find a closed structured form of the detection problem (4.3).

4.4 Toeplitz GMMSE Problem

Consider we generate a rectangular banded Toeplitz channel matrix $\boldsymbol{H}$ that has a size of $m \times n$, such that, $m = n + L - 1$, where, L is the channel length as shown by (4.9).

$$\boldsymbol{H} = \begin{pmatrix} \boldsymbol{h}^{(1)} & & & & \\ \boldsymbol{h}^{(2)} & \boldsymbol{h}^{(1)} & & & \\ \boldsymbol{h}^{(3)} & \boldsymbol{h}^{(2)} & \boldsymbol{h}^{(1)} & & \\ & \boldsymbol{h}^{(3)} & \boldsymbol{h}^{(2)} & \ddots & \\ \vdots & & \boldsymbol{h}^{(3)} & & \boldsymbol{h}^{(1)} \\ \boldsymbol{h}^{(L)} & \vdots & & & \boldsymbol{h}^{(2)} \\ & \boldsymbol{h}^{(L)} & & & \boldsymbol{h}^{(3)} \\ & & & \ddots & \vdots \\ & & & & \boldsymbol{h}^{(L)} \end{pmatrix}. \tag{4.9}$$

We express the matrix $\boldsymbol{H}^H\boldsymbol{H}$ by its Eigenvalue Decomposition (EVD),

$$\boldsymbol{H}^H\boldsymbol{H} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^H,$$

where $\boldsymbol{V}$ is the matrix whose columns are the eigenvectors of $\boldsymbol{H}^H\boldsymbol{H}$ and $\boldsymbol{\Lambda}$ is a diagonal matrix that contains the corresponding eigenvalues. Problem (4.3) can be rewritten as

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x}^H\boldsymbol{x} \leq n} \boldsymbol{x}^H \left(\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^H\right)\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x}. \tag{4.10}$$

The Lagrangian dual function can be easily expressed as,

$$\mathcal{L}\left(\boldsymbol{x}, \lambda\right) = \boldsymbol{x}^H(\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^H)\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x} + \lambda\left(\boldsymbol{x}^H\boldsymbol{x} - n\right),$$

which is also minimized over $\boldsymbol{x}$ and maximized over $\lambda \geq 0$. Solving for $\boldsymbol{x}$ in terms of λ we get

$$\boldsymbol{x} = \boldsymbol{V}(\boldsymbol{\Lambda} + \lambda\boldsymbol{I})^{-1}\boldsymbol{V}^H\boldsymbol{y}. \tag{4.11}$$

The dual problem for problem (4.10) takes the form

$$\max_{\lambda \geq 0} -\boldsymbol{y}^H\left(\left(\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^H\right) + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{y} - \lambda n. \tag{4.12}$$

Problem (4.12) is a convex optimization problem subject to $\lambda \geq 0$, where, $\lambda \in \mathbb{R}$, which is a one-variable optimization problem. This problem is also called a one-dimensional convex optimization problem which is easier to solve than a multi-dimensional optimization problem [7]. We apply the same strategy as in Section 4.2 to solve problem (4.12). The unconstrained gradient descent algorithm is one (another algorithm is shown in the next chapter) of the efficient algorithms to solve this problem and it takes the form,

$$\bar{\lambda}(t+1) = \bar{\lambda}(t) + \mu\left(\boldsymbol{y}^H \boldsymbol{V}\left(\boldsymbol{\Lambda} + \bar{\lambda}(t)\boldsymbol{I}\right)^{-2}\boldsymbol{V}^H\boldsymbol{y} - n\right). \tag{4.13}$$

Using unconstrained gradient descent algorithm, we get the optimal value of λ, which is denoted by λ^*. To solve problem (4.12), we easily substitute the value of λ^* in (4.11). The GMMSE solution in this case is

$$\boldsymbol{x}^*_{GMMSE} = \boldsymbol{V}\left(\boldsymbol{\Lambda} + \lambda^*\boldsymbol{I}\right)^{-1}\boldsymbol{V}^H\boldsymbol{y}. \tag{4.14}$$

Expressing the matrix $\boldsymbol{H}^H\boldsymbol{H}$ by its EVD, the gradient descent algorithm (4.13) only works on the diagonal matrix $\boldsymbol{\Lambda}$ which has a less number of computations to find λ^*, but the gradient descent algorithm in (4.6) has to do more efforts to find δ^*. So, the solution in (4.14) is less complex than that in (4.7).

4.5 Circular Approximation GMMSE Problem

We present an algorithm that is based on fast convolution method, using the well-known Fast Fourier Transform (FFT) as the basic computational tool. As we will discuss, the FFT algorithm can transform the problem of multiplying with a cyclic convolution matrix to the easier problem of multiplying with a diagonal matrix. We will also formalize this idea, extend it to compute an approximation to the GMMSE solution.

The well-known fast convolution method [82] can be used in its most basic form to efficiently compute the convolution of a finite sequence $\underline{\boldsymbol{d}}$ of length n with an infinite, but repeating sequence $\underline{\boldsymbol{c}}$ with periods of length n. Such a convolution is called a *cyclic convolution* [80]. The result of the cyclic convolution is again infinite, periodic sequence with periods of length n, denoted by $\underline{\boldsymbol{s}}$. Let the vector $\boldsymbol{c} = [\boldsymbol{c}^{(1)}, \boldsymbol{c}^{(2)}, ..., \boldsymbol{c}^{(n)}]^T \in \mathbb{C}^n$

contain the elements of one period of the sequence $\underline{\boldsymbol{c}}$ and let $\boldsymbol{d}, \boldsymbol{s}$ contain the elements of the sequence $\underline{\boldsymbol{d}}$ and one period of $\underline{\boldsymbol{s}}$, respectively. The result of the cyclic convolution can then be expressed in matrix notation as

$$\boldsymbol{s} = \boldsymbol{C}\boldsymbol{d}, \tag{4.15}$$

where $\boldsymbol{C}$ is a circular matrix given by (4.8).

The computation of $\boldsymbol{s}$ as a matrix/vector product requires n^2 multiplications when the structure of $\boldsymbol{C}$ is not exploited. The fast convolution method reduces this number to $O(n \log n)$ by transforming $\underline{\boldsymbol{d}}$ and $\underline{\boldsymbol{c}}$ into the frequency domain where the convolution can be performed by a point-wise multiplication of their spectra. The sequence $\underline{\boldsymbol{s}}$ can be found by transforming the resulting spectrum back into the time domain. In this basic form, the method works only when one sequence is infinite and periodic since otherwise its spectrum would not be discrete and can not be accurately computed with the FFT, which performs a Discrete Fourier Transform (DFT) [87, 106]. We can express the DFT of the vector $\boldsymbol{d}$ into $\boldsymbol{d}_{F_c}$ as a matrix/vector product using the Fourier Transform matrix $\boldsymbol{F}_c \in \mathbb{C}^{n\times n}$,

$$\boldsymbol{d}_{F_c} = \boldsymbol{F}_c\ \boldsymbol{d},$$

with $\boldsymbol{F}_c[i,k] = w^{(i-1)(k-1)}, w = e^{-\frac{2\pi}{n}j}$, where j denotes the imaginary unit. The product $\boldsymbol{F}_c\ \boldsymbol{d}$ can be efficiently computed with FFT algorithm. We keep in mind that the computation of the matrix/vector product $\boldsymbol{F}_c\ \boldsymbol{d}$ needs only $\frac{n}{2}\log n$ multiplication instead of n^2.

The computation of $\boldsymbol{s}$ in the frequency domain can be expressed as

$$\boldsymbol{s} = \boldsymbol{F}_c^{-1}\boldsymbol{\Lambda}_c\boldsymbol{F}_c\ \boldsymbol{d}, \tag{4.16}$$

with $\boldsymbol{\Lambda}_c = diag(\boldsymbol{F}_c\ \boldsymbol{c}) \in \mathbb{C}^{n\times n}$, where, $diag(\boldsymbol{z})$ denotes the construction of a diagonal $n\times n$ matrix that contains the elements of $\boldsymbol{z}$ on its diagonal. Multiplying the diagonal matrix $\boldsymbol{\Lambda}_c$ with the vector $\boldsymbol{F}_c\ \boldsymbol{d}$ corresponds to the point-wise multiplication of the two spectra. The subsequent multiplication with $\boldsymbol{F}_c^{-1}$ transforms the result back into the time domain with an inverse DFT that can again be implemented as an inverse FFT with $\frac{n}{2}\log n$ multiplications. Comparing (4.16) with (4.15), we find the factorization,

$$\boldsymbol{C} = \boldsymbol{F}_c^{-1}\boldsymbol{\Lambda}_c\boldsymbol{F}_c, \tag{4.17}$$

and hence, $\boldsymbol{F}_c\boldsymbol{C}\boldsymbol{F}_c^{-1} = \boldsymbol{\Lambda}_c$. All details about FFT can also be found in [12]

The construction (4.17) works for any circular matrix, so using this fact, the circular matrices can be diagonalized with the Fourier transform matrix. We can reduce the computational complexity of GMMSE detector as we will show in the following.

The banded rectangular Toeplitz matrix $\boldsymbol{H}$ in (4.9) can be approximated by a circular structure $\tilde{\boldsymbol{H}}$ by adding $L-1$ columns to the Toeplitz matrix,

$$\tilde{\boldsymbol{H}} = \begin{pmatrix} \boldsymbol{h}^{(1)} & & & & & \boldsymbol{h}^{(L)} & \boldsymbol{h}^{(L-1)} & \dots & & \boldsymbol{h}^{(2)} \\ \boldsymbol{h}^{(2)} & \boldsymbol{h}^{(1)} & & & & & \boldsymbol{h}^{(L)} & \boldsymbol{h}^{(L-1)} & \dots & \boldsymbol{h}^{(3)} \\ \boldsymbol{h}^{(3)} & \boldsymbol{h}^{(2)} & \boldsymbol{h}^{(1)} & & & & & & & \\ & \boldsymbol{h}^{(3)} & \boldsymbol{h}^{(2)} & \ddots & & & & & & \\ \vdots & & \boldsymbol{h}^{(3)} & & \boldsymbol{h}^{(1)} & & & & & \\ \boldsymbol{h}^{(L)} & \vdots & & & \boldsymbol{h}^{(2)} & & & & & \\ & \boldsymbol{h}^{(L)} & & & \boldsymbol{h}^{(3)} & & & & & \\ & & & \ddots & \vdots & & & & & \\ & & & & \boldsymbol{h}^{(L)} & \boldsymbol{h}^{(L-1)} & \boldsymbol{h}^{(L-2)} & & \dots & \boldsymbol{h}^{(1)} \end{pmatrix}.$$

Now, we will reformulate problem (4.3) by using the circular approximation $\tilde{\boldsymbol{H}}$. Note that, the size of $\tilde{\boldsymbol{H}}$ is $n+L-1 \times n+L-1$. Using the FFT factorization in (4.17) and the fact that $\tilde{\boldsymbol{H}}^H\tilde{\boldsymbol{H}}$ is a circular matrix, we express it as,

$$\tilde{\boldsymbol{H}}^H\tilde{\boldsymbol{H}} = \boldsymbol{F}^H\tilde{\boldsymbol{\Lambda}}\boldsymbol{F},$$

where $\boldsymbol{F}$ is the discrete Fourier transform matrix (computed by FFT) and

$$\tilde{\boldsymbol{\Lambda}} = diag\left(\boldsymbol{F}\cdot(\tilde{\boldsymbol{H}^H\tilde{\boldsymbol{H}}}\,(:,1))\right),$$

where, $(\tilde{\boldsymbol{H}^H\tilde{\boldsymbol{H}}}\,(:,1))$ is the first column of the circular matrix $\tilde{\boldsymbol{H}^H\tilde{\boldsymbol{H}}}$.

Problem (4.3) can be reformulated as,

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}^H\boldsymbol{x}\le n} \boldsymbol{x}^H\left(\boldsymbol{F}^H\tilde{\boldsymbol{\Lambda}}\boldsymbol{F}\right)\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x}. \tag{4.18}$$

Again, we construct the Lagrange dual function of (4.18) as,

$$\mathcal{L}(\boldsymbol{x},\lambda) = \boldsymbol{x}^H\left(\boldsymbol{F}^H\tilde{\boldsymbol{\Lambda}}\boldsymbol{F}\right)\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x} + \lambda\left(\boldsymbol{x}^H\boldsymbol{x} - n\right),$$

which is also minimized over $\boldsymbol{x}$ and maximized over $\lambda \geq 0$. Solving for $\boldsymbol{x}$ in terms of λ, we get,

$$\boldsymbol{x} = \boldsymbol{F}^H \left(\tilde{\boldsymbol{\Lambda}} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{F}\boldsymbol{y}, \tag{4.19}$$

and substituting back, we obtain the dual problem,

$$\max_{\lambda \geq 0} -\boldsymbol{y}^H \left(\left(\boldsymbol{F}^H \tilde{\boldsymbol{\Lambda}} \boldsymbol{F}\right) + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{y} - \lambda n. \tag{4.20}$$

This problem as we stated before, is a one dimension-optimization problem, so it is easy to solve it than problem (4.18). A simple unconstrained gradient descent algorithm is given by,

$$\bar{\lambda}\left(t+1\right) = \bar{\lambda}\left(t\right) + \mu \left(\boldsymbol{y}^H \boldsymbol{F}^H \left(\tilde{\boldsymbol{\Lambda}} + \bar{\lambda}\left(t\right) \boldsymbol{I}\right)^{-2} \boldsymbol{F}\boldsymbol{y} - n\right). \tag{4.21}$$

After solving the dual problem (4.20) using (4.21), we get the optimum value λ^*. Substituting this value into (4.19), we obtain the GMMSE solution,

$$\boldsymbol{x}^*_{GMMSE} = \boldsymbol{F}^H \left(\tilde{\boldsymbol{\Lambda}} + \lambda^* \boldsymbol{I}\right)^{-1} \boldsymbol{F}\boldsymbol{y}. \tag{4.22}$$

It is worthwhile to note that, EVD which used in (4.21) is not only the reason to reduce the complexity to find λ^* (as in Toeplitz case), in addition, using the FFT algorithm. The products $\boldsymbol{y}^H \boldsymbol{F}^H$ and $\boldsymbol{F}\boldsymbol{y}$ in (4.21) can efficiently computed with the FFT algorithm. We will discuss the computational complexity of the proposed detector in Chapter 6. We just now keep in mind that the matrix/vector product $\boldsymbol{F}\boldsymbol{z}$ needs only $O(n\ log\ n)$.

4.6 Performance Analysis

When $\boldsymbol{h} = [\boldsymbol{h}^{(1)}, \boldsymbol{h}^{(2)}, ..., \boldsymbol{h}^{(L)}]^H$ is not a zero vector, the columns of matrix $\boldsymbol{H}$ are linearly independent. Actually, the Toeplitz channel convolution matrix $\boldsymbol{H}$ of size $(n + L - 1) \times n$ always has full rank. The channel correlation matrix $\boldsymbol{H}^H \boldsymbol{H}$ therefore also has full rank. So its inverse exists. Therefore, when the channel matrix is multiplied by its pseudo inverse, the identity matrix is obtained,

$$\boldsymbol{H}^\dagger \boldsymbol{H} = \left(\boldsymbol{H}^H \boldsymbol{H}\right)^{-1} \left(\boldsymbol{H}^H \boldsymbol{H}\right) = \boldsymbol{I},$$

where, $\boldsymbol{H}^{\dagger}$ is the pseudo inverse of $\boldsymbol{H}$. This means that the least squares estimate is computed by

$$\boldsymbol{x}^{*} = \boldsymbol{H}^{\dagger}\boldsymbol{r} = \boldsymbol{x} + \boldsymbol{H}^{\dagger}\boldsymbol{n}.$$

We observe that the least squares estimate is composed of the true data and the noise term. Therefore, in noise-free environments, perfect data detection is guaranteed. In noisy environments, however, the quality of the estimation depends on the noise and its enhancement by $\boldsymbol{H}^{\dagger}$. Assuming a fixed noise power, we measure the noise enhancement and thus the quality of the estimates by the condition number of $\boldsymbol{H}^{\dagger}$. A large condition number indicates large noise enhancement, whereas a small condition number indicates low noise enhancement.

Now, we consider the circular matrix $\tilde{\boldsymbol{H}}$ obtained by the circular approximation of $\boldsymbol{H}$. The first n columns corresponding to the Toeplitz channel matrix are linearly independent. The additional $L-1$ columns may be linearly dependent. Actually, a circular matrix $\tilde{\boldsymbol{H}}$ has a maximum of $L-1$ eigenvalues equal to zero. Therefore in contrast to the Toeplitz case, the existence of the inverse is not guaranteed. The pseudo inverse obtained by least square estimation, is identical to the inverse, if it exists,

$$\tilde{\boldsymbol{H}}^{\dagger} = (\tilde{\boldsymbol{H}}^{H}\tilde{\boldsymbol{H}})^{-1}\tilde{\boldsymbol{H}}^{H} = \tilde{\boldsymbol{H}}^{-1}\tilde{\boldsymbol{H}}^{-H}\tilde{\boldsymbol{H}}^{H} = \tilde{\boldsymbol{H}}^{-1}.$$

Therefore, estimates can be computed for channels, where the inverse circulant matrix exists (note, that for practical channels, this is usual the case):

$$\boldsymbol{x}^{*} = \tilde{\boldsymbol{H}}^{-1}\boldsymbol{r} = \boldsymbol{x} + \tilde{\boldsymbol{H}}^{-1}\boldsymbol{n}.$$

As for the Toeplitz case the quality of the least squares estimate can be measured by the condition number of $\tilde{\boldsymbol{H}}^{-1}$. We compute the condition numbers of the (pseudo) inverse of the Toeplitz and circular matrices for Rayleigh fading channel as specified in [43, 86]. Here, the condition number is the ratio between the largest and smallest singular value. The occurring condition numbers are then assigned to specific bins in the histogram shown in Figure 4.2. Here, the Toeplitz matrix averages a lower condition number than the circulant one. Noise enhancement is, therefore, lower in the Toeplitz case. The highest measured condition number of the Toeplitz matrix lies in the bin [40 80], while in circular case, condition numbers above 320 occur. How this degradation of the condition influences the BER performance is shown by the following simulations.

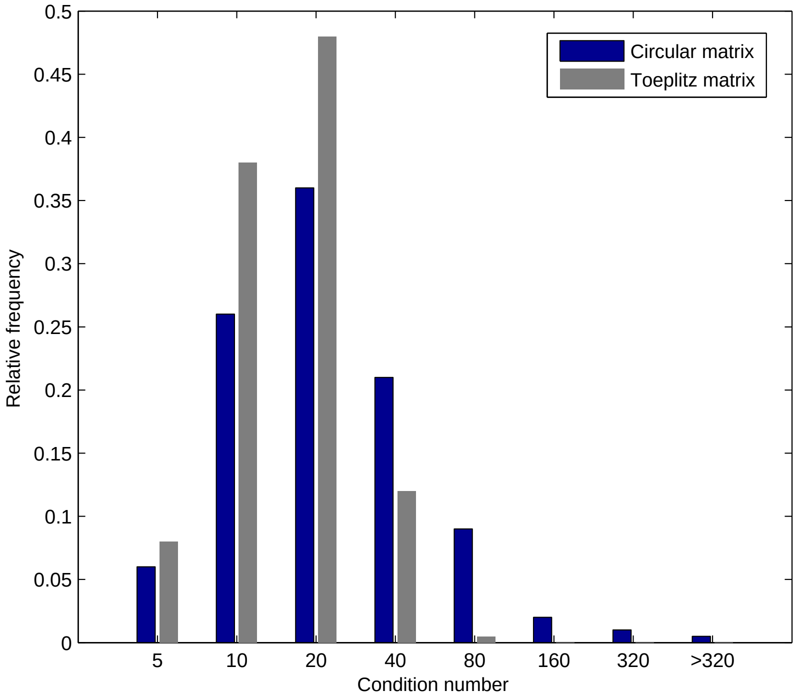


Figure 4.2: Relative frequency of condition numbers for Toeplitz and circular matrices with $n = 100$ using a fading channel of length $L = 15$.

4.7 Simulation Results

Error rate is a function of the signal to noise ration (SNR), which is commonly defined as the logarithmic ratio of signal power to noise power at the matched filter output [82]. Assuming uncorrelated zero-mean transmit symbols $\boldsymbol{x}$ and noise samples $\boldsymbol{n}$, the transmit power and the noise power can be defined by their variances as $\sigma_{\boldsymbol{x}}^2$ and $\sigma_{\boldsymbol{n}}^2$ respectively. Without loss of generality, the transmit power is assumed to be normalized to unit power, i.e. $\sigma_{\boldsymbol{x}}^2 = 1$. Hence, if the transmission is only affected by an Additive White Gaussian Noise (AWGN), the SNR can be expressed as the ration of the transmit power $\sigma_{\boldsymbol{x}}^2$ and the noise power $\sigma_{\boldsymbol{n}}^2$, i.e.

$$SNR[dB] = 10\log_{10}(\frac{\sigma_{\boldsymbol{x}}^2}{\sigma_{\boldsymbol{n}}^2}) = 10\log_{10}(\frac{1}{\sigma_{\boldsymbol{n}}^2}).$$

Simulations results are generated in *Matlab* [1]. The bit error rate (BER) performance of the different detectors is discussed for BPSK constellation. In the simulation we compare the BER performance for LS, MMSE, and GMMSE detectors, taking into account that we have two different structures, banded Toeplitz and its circular approximation. We applied this simulation using two channels:

- Channel (1): is a channel that specified in [36] as,
  $\boldsymbol{h}^{(i)} = 0.5 * (1 + cos(2\pi/w * (i-2)))$; $i = 1, 2, ..., L$,
  where w determines the distortion.

- Channel (2): is a Rayleigh fading channel as specified in [43, 86].

The equalization problem for each channel was simulated for different SNR using $L = 5, 15$ and $n = 100, 1000$.

Figures 4.3, 4.4, 4.5 and 4.6 show that GMMSE detector has almost the same performance as MMSE detector. We see that the circular approximation only slightly degrades the performance of the detectors.

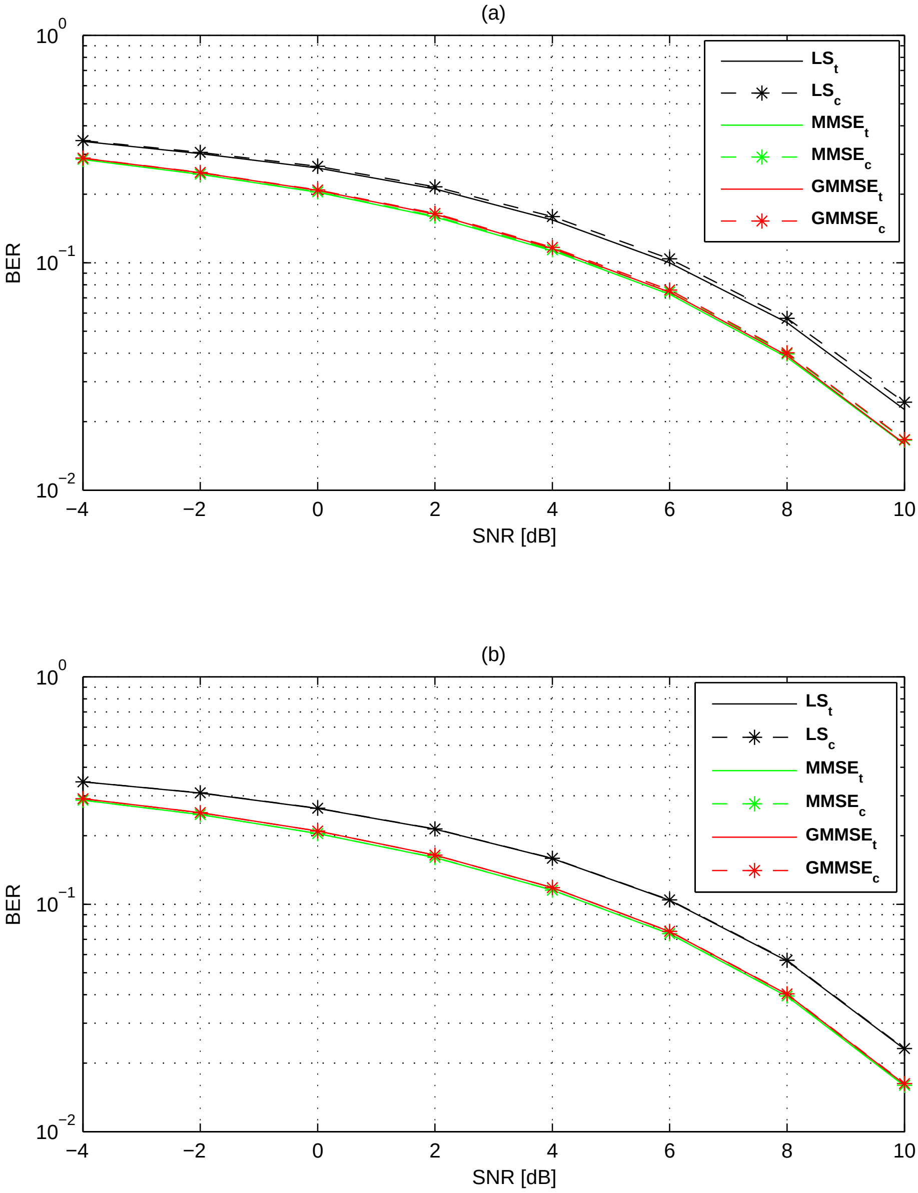
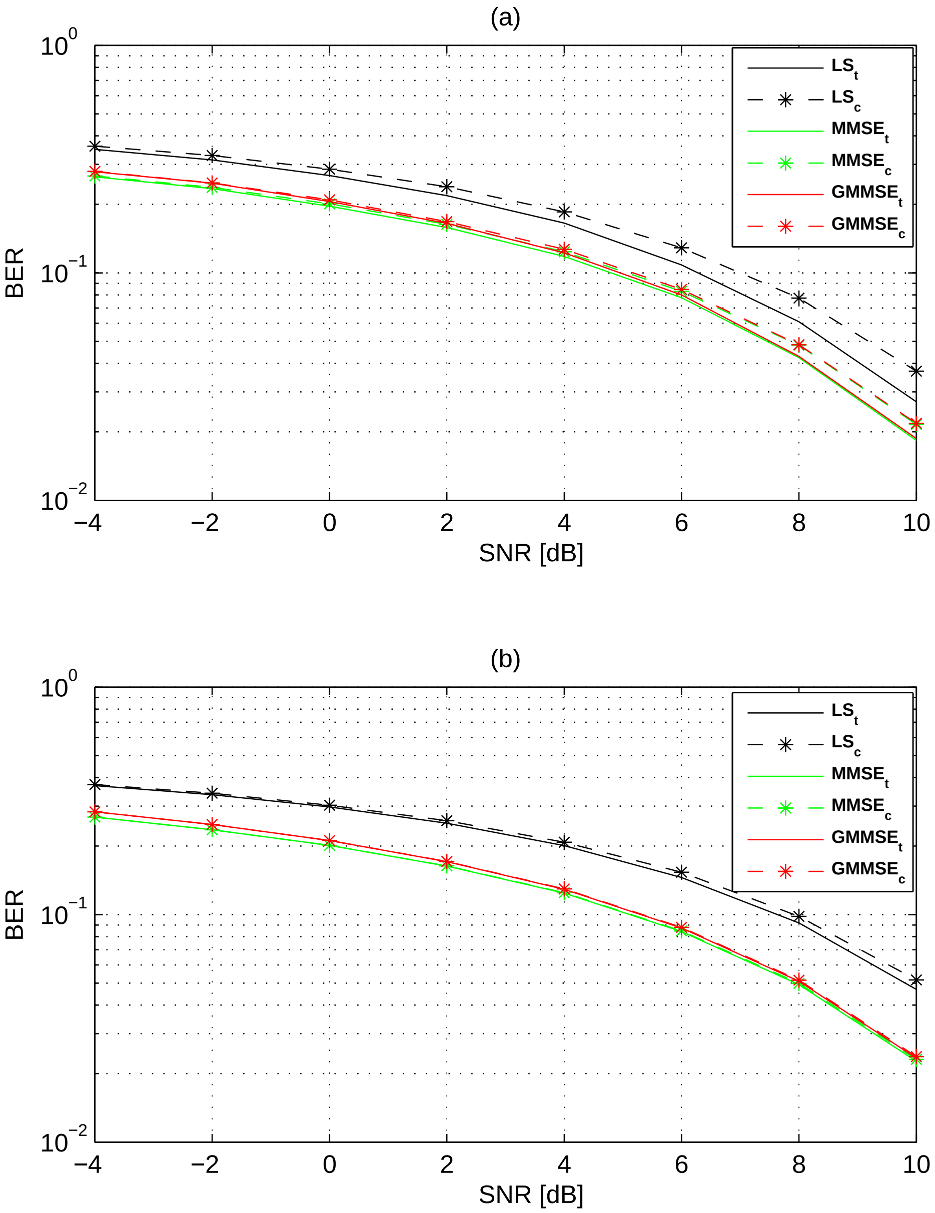


Figure 4.3: BER for structured LS, MMSE, and GMMSE detectors using channel (1) with a channel length $L = 5$, (a) $n = 100$, (b) $n = 1000$ (t: Toeplitz case; c: Circular case).


(a)

(b)

Figure 4.4: BER for structured LS, MMSE, and GMMSE detectors using channel (1) with a channel length $L = 15$, (a) $n = 100$, (b) $n = 1000$ (t: Toeplitz case; c: Circular case).

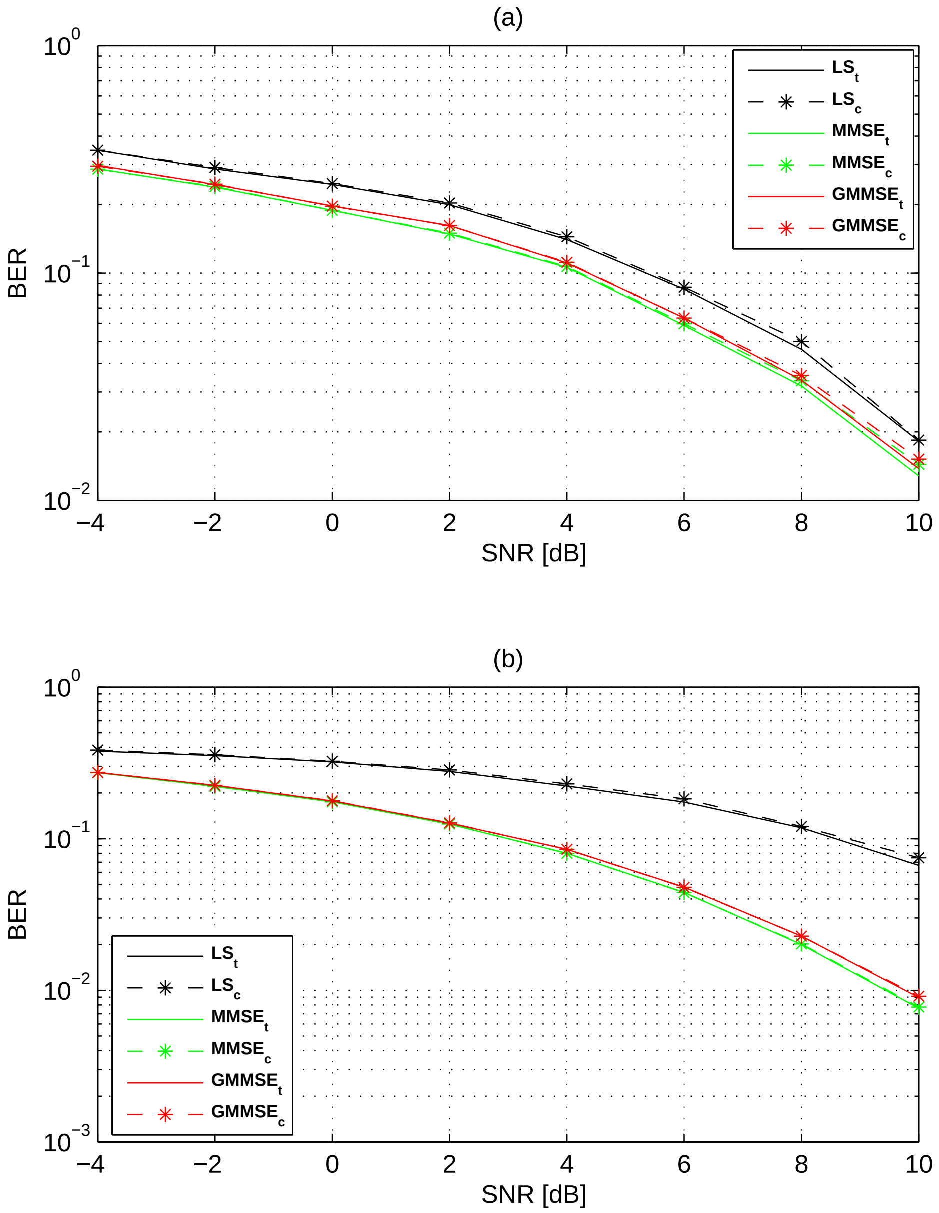


Figure 4.5: BER for structured LS, MMSE, and GMMSE detectors using channel (2) with a channel length $L = 5$, (a) $n = 100$, (b) $n = 1000$ (t: Toeplitz case; c: Circular case).

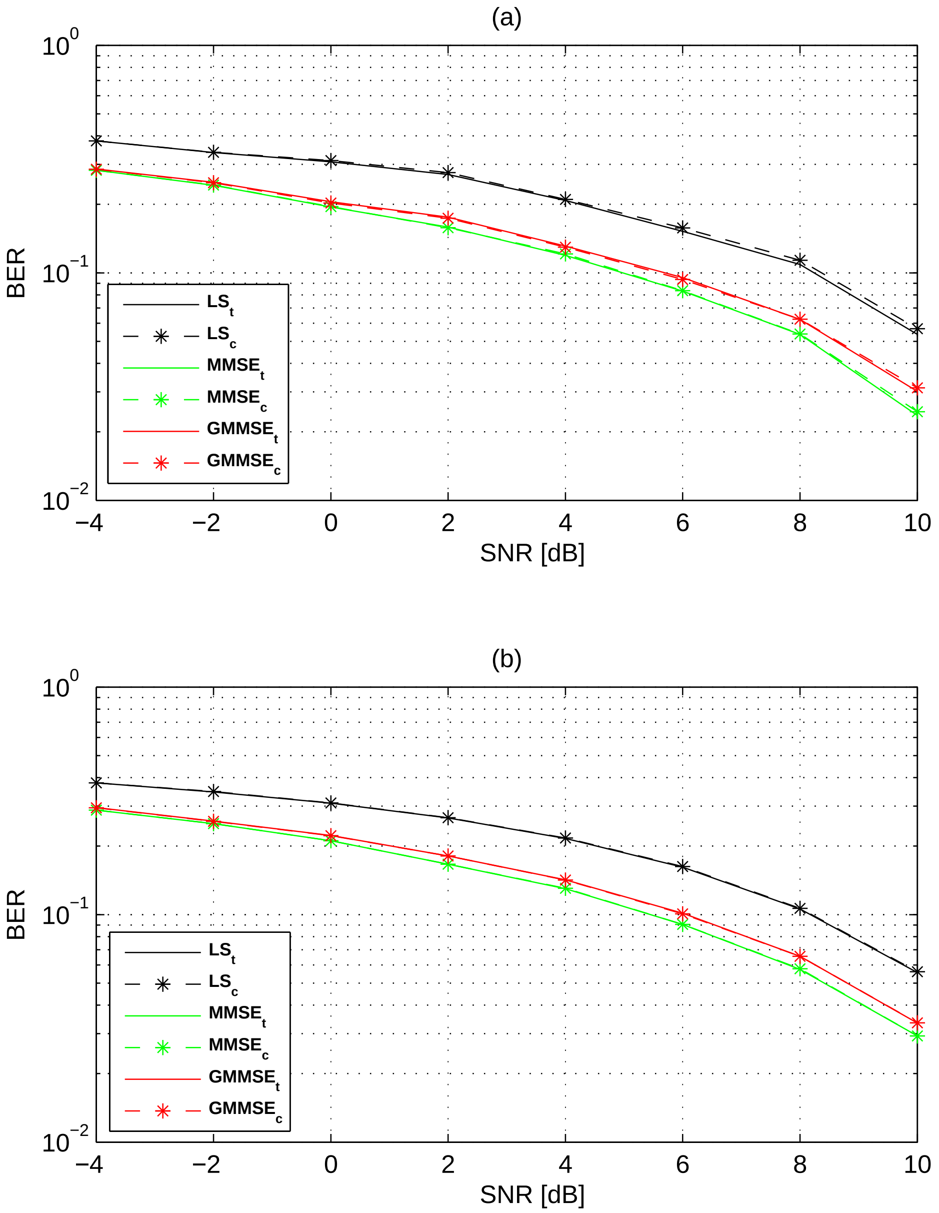


Figure 4.6: BER for structured LS, MMSE, and GMMSE detectors using channel (2) with a channel length $L = 15$, (a) $n = 100$, (b) $n = 1000$ (t: Toeplitz case; c: Circular case).

5 Structured GMMSE Using Hidden Convexity

One of the most popular relaxation methods of the ML problem (4.2), is semidefinite programming (SDR) (3.31). It is a convex optimization problem that can be solved in polynomial time [97], it provides a good approximation of the ML solution [64, 105]. However, its practical applications are limited since, its computational load is very high. This high complexity urged us to think about how can we use convex relaxations to produce a receiver that has a performance near to ML with a reduced complexity.

In Chapter 4, we relaxed problem (4.2) to a convex relaxation (4.3) and we solved this problem by using the circular approximation. The resulting solution (4.22) has the same structure and performance as MMSE detector and it does not require the knowledge of the noise power.

In this chapter, we relax the ML problem (4.2) to a convex optimization problem again. We solve its dual problem to find what is named, the estimation of the noise power to get the same solution form as (4.22). The main difference between (4.22) and the proposed solution in this chapter is the way how the dual problem is solved. The bisection method [45] is used to solve the dual problem.
The advantages of the bisection method are:

- The bisection method is always convergent.
- The error can be controlled.

5.1 Hidden Convexity Relaxation

In this section, we propose a relaxation whose bit error rate performance is almost the same as MMSE detector and it has a solution form as in our proposed detector (4.22). Using hidden convexity methodology [9], we show that problem (4.2) can be rewritten as,

$$\hat{\boldsymbol{x}} = \arg \min_{\|\boldsymbol{x}\|^2=n} \boldsymbol{x}^H \boldsymbol{H}^H \boldsymbol{H} \boldsymbol{x} - 2\boldsymbol{y}^H \boldsymbol{x}. \tag{5.1}$$

Let $\boldsymbol{V}$ be the matrix whose columns are the eigenvectors of $\boldsymbol{H}^H\boldsymbol{H}$ and $\boldsymbol{\Lambda} = diag(\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)}, ..., \boldsymbol{\lambda}^{(n)})$ contains the eigenvalues of $\boldsymbol{H}^H\boldsymbol{H}$ with $\boldsymbol{x} = \boldsymbol{V}\boldsymbol{z}$, problem (5.1) is equivalent to

$$\min_{\|\boldsymbol{z}\|^2=n} \sum_{j=1}^{n} (\boldsymbol{\lambda}^{(j)} {\boldsymbol{z}^{(j)}}^2 - 2\boldsymbol{b}^{(j)}\boldsymbol{z}^{(j)}), \tag{5.2}$$

where $\boldsymbol{b} = \boldsymbol{V}^H\boldsymbol{y}$. The following lemma enables us to convert problem (5.2) to a convex optimization problem [27].

Lemma 5.1 *Let* $\boldsymbol{\omega} = (\boldsymbol{\omega}^{(1)}, \boldsymbol{\omega}^{(2)}, ..., \boldsymbol{\omega}^{(n)})$ *be an optimal solution of* $\min\limits_{\|\boldsymbol{z}\|^2=n} q(\boldsymbol{z})$ *where,* $q(\boldsymbol{z}) = \sum_{j=1}^{n} (\boldsymbol{\lambda}^{(j)} {\boldsymbol{z}^{(j)}}^2 - 2\boldsymbol{b}^{(j)}\boldsymbol{z}^{(j)})$. *Then* $\boldsymbol{\omega}^{(j)}\boldsymbol{b}^{(j)} \geq 0$ *for* $1 \leq j \leq n$.

Proof: Let $\boldsymbol{\omega}_k = \left(\boldsymbol{\omega}^{(1)}, \boldsymbol{\omega}^{(2)}, ..., \boldsymbol{\omega}^{(k-1)}, -\boldsymbol{\omega}^{(k)}, \boldsymbol{\omega}^{(k+1)}, ..., \boldsymbol{\omega}^{(n)}\right)$. Then, $\|\boldsymbol{\omega}_k\|^2 = \|\boldsymbol{\omega}\|^2 = n$, so that $\boldsymbol{\omega}_k$ is feasible. Since $\boldsymbol{\omega}$ is optimal, $q(\boldsymbol{\omega}) \leq q(\boldsymbol{\omega}_k)$ for $1 \leq k \leq n$, which implies that

$$-\sum_{j=1}^{n} 2\boldsymbol{b}^{(j)}\boldsymbol{\omega}^{(j)} \leq -\sum_{j=1, j\neq k}^{n} 2\boldsymbol{b}^{(j)}\boldsymbol{\omega}^{(j)} + 2\boldsymbol{b}^{(k)}\boldsymbol{\omega}^{(k)}.$$

Therefore, $\boldsymbol{b}^{(k)}\boldsymbol{\omega}^{(k)} \geq 0$, and the result follows.

Using lemma 5.1, we can define a new objective variable (change of variables) $\boldsymbol{u}$ such that, $\boldsymbol{z}^{(j)} = sign(\boldsymbol{b}^{(j)})\sqrt{\boldsymbol{u}^{(j)}}$, with $\boldsymbol{u}^{(j)} \geq 0$. Problem (5.2) is now written as,

$$\min_{\boldsymbol{u}^{(j)}\geq 0}\sum_{j=1}^{n}(\boldsymbol{\lambda}^{(j)}\boldsymbol{u}^{(j)}-2|\boldsymbol{b}^{(j)}|\sqrt{\boldsymbol{u}^{(j)}}):\sum_{j=1}^{n}u_j=n. \tag{5.3}$$

Since problem (5.3) is convex with linear constraints, we develop the dual problem using its Lagrangian,

$$\mathcal{L}(\boldsymbol{u},\eta)=\sum_{j=1}^{n}(\boldsymbol{\lambda}^{(j)}+\eta)\boldsymbol{u}^{(j)}-2|\boldsymbol{b}^{(j)}|\sqrt{\boldsymbol{u}^{(j)}}-\eta n. \tag{5.4}$$

Differentiating (5.4) with respect to $\boldsymbol{u}^{(j)}$ and equating to zero yields,

$$\boldsymbol{u}^{(j)}=\frac{\boldsymbol{b}^{(j)^2}}{(\boldsymbol{\lambda}^{(j)}+\eta)^2}, 1\leq j\leq n,$$

subject to $\eta\geq-\boldsymbol{\lambda}^{(j)}$ for $1\leq j\leq n$. The dual function is

$$h(\eta)=\min_{\boldsymbol{u}^{(j)}\geq 0}\mathcal{L}(\boldsymbol{u},\eta)=-\sum_{j=1}^{n}\frac{\boldsymbol{b}^{(j)^2}}{(\boldsymbol{\lambda}^{(j)}+\eta)}-\eta n,$$

and the dual problem of (5.3) is

$$\max_{\eta} h(\eta):\eta\geq\alpha,$$

where $\alpha=\max_{1\leq j\leq n}\{-\boldsymbol{\lambda}^{(j)}\}$.

Differentiating $h(\eta)$ with respect to η and equating to 0 yields,

$$\sum_{j=1}^{n}\frac{\boldsymbol{b}^{(j)^2}}{(\boldsymbol{\lambda}^{(j)}+\eta)^2}=n.$$

Denoting

$$G(\eta)=\sum_{j=1}^{n}\frac{\boldsymbol{b}^{(j)^2}}{(\boldsymbol{\lambda}^{(j)}+\eta)^2}-n, \tag{5.5}$$

it follows that the optimal η^* is the root of $G(\eta)$. Since $G(\eta)$ is continuous and monotonically decreasing for $\eta>\alpha$, there is only one root in the domain (α,∞). One of the most efficient methods that solves this problem is the bisection method.

5.2 Bisection Method

In this section, we give a summary of the bisection method. Bisection method is an algorithm that solves nonlinear equations by finding roots of these equations. The method is based on the following theorem [46].

Theorem 5.2 *An equation $f(x) = 0\ \forall x \in \mathbb{R}$, where $f(x)$ is a real continuous function, has at least one root between x_l (lower bound) and x_u (upper bound) if $f(x_l)f(x_u) < 0$.*

There are some possible cases as:

- **First case**: if $f(x_l)f(x_u) > 0$, there may or may not be any root between x_l and x_u.
- **Second case**: if $f(x_l)f(x_u) < 0$, then there may be more than one root between x_l and x_u.

Figure 5.1 shows the theorem. Figure 5.2 and Figure 5.3 represent the first case, while Figure 5.4 is an example of the second case.

Bisection method as described in [45, 46] is based on how to find the root between two points, so it is one of the bracketing methods. Because the root is bracketed two points x_l and x_u, we can find the midpoint x_m between x_l and x_u. This property gives us two new intervals $[x_l, x_m]$ and $[x_m, x_u]$. We have to determine, if the root is in $[x_l, x_m]$ or in $[x_m, x_u]$. Then, find the sign of $f(x_l)f(x_m)$, and if $f(x_l)f(x_m) < 0$, then the new bracket is between x_l and x_m, otherwise, it is between x_m and x_u. So, we are literally halving the interval. By repeating this process, we can let the width of the interval $[x_l, x_u]$ to be smaller.

Bisection Method Algorithm

The bisection algorithm steps that find the root of the equation $f(x) = 0$ are:

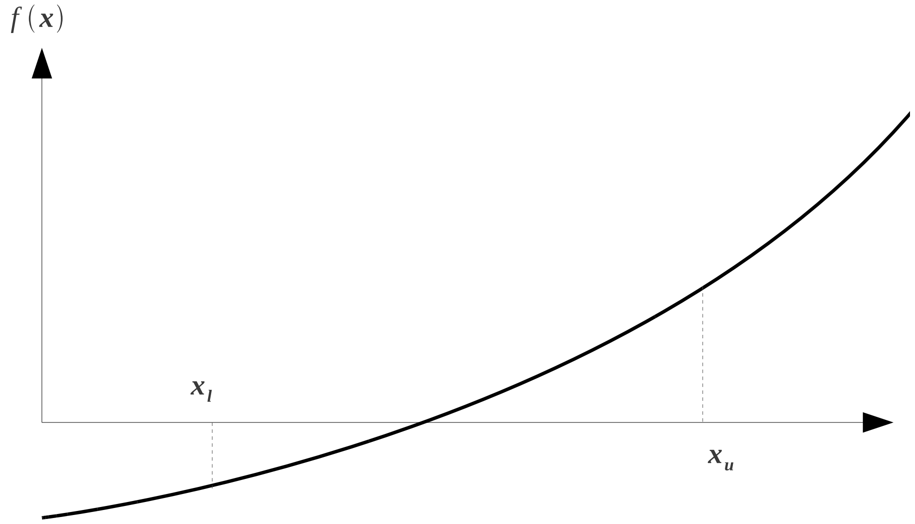


Figure 5.1: At least one root exists between the two points.

1. Choose x_l and x_u as two guesses for the root such that

$$f(x_l)f(x_u) < 0,$$

which means that $f(x)$ changes its sign between x_l and x_u.

2. Find an estimate x_m for the root of the equation $f(x) = 0$ as the mid-point between x_l and x_u as

$$x_m = \frac{x_l + x_u}{2}.$$

3. Check the following

   - If $f(x_l)f(x_m) < 0$, then the root lies between x_l and x_m; then $x_l = x_l$ and $x_u = x_m$.
   - If $f(x_l)f(x_m) > 0$, then the root lies between x_m and x_u; then $x_l = x_m$ and $x_u = x_u$.
   - If $f(x_l)f(x_m) = 0$, then the root is x_m, stop the algorithm.

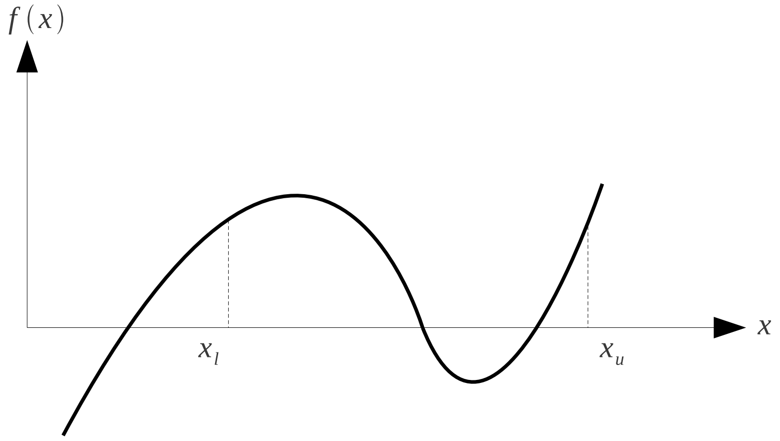


Figure 5.2: Roots of equation may still exist between the two points.

4. Find the new estimate of the root

$$x_m = \frac{x_l + x_u}{2}.$$

Find the absolute relative approximate error as

$$||\epsilon_a|| = ||\frac{x_m^{new} - x_m^{old}}{x_m^{new}}||,$$

where, x_m^{new} is the estimated root from the present iteration while, x_m^{old} is the estimated root from the previous iteration.

5. Compare the absolute relative approximation error $||\epsilon_a||$ with the pre-specified relative error tolerance $||\epsilon_s||$. If $||\epsilon_a|| > \epsilon_s$, then go to step 3, else stop the algorithm.

Example: Find a root of the nonlinear equation $f(x) = 0$, where, $f(x) = x^3 - 0.165x^2 + 3.993 * 10^{-4}$. In this example, the lower initial guess is $x_l = 0$ and the upper initial guess is $x_u = 0.11$ as shown in Figure 5.5. Figures 5.6 - 5.8 show the first three iterations of the bisection method

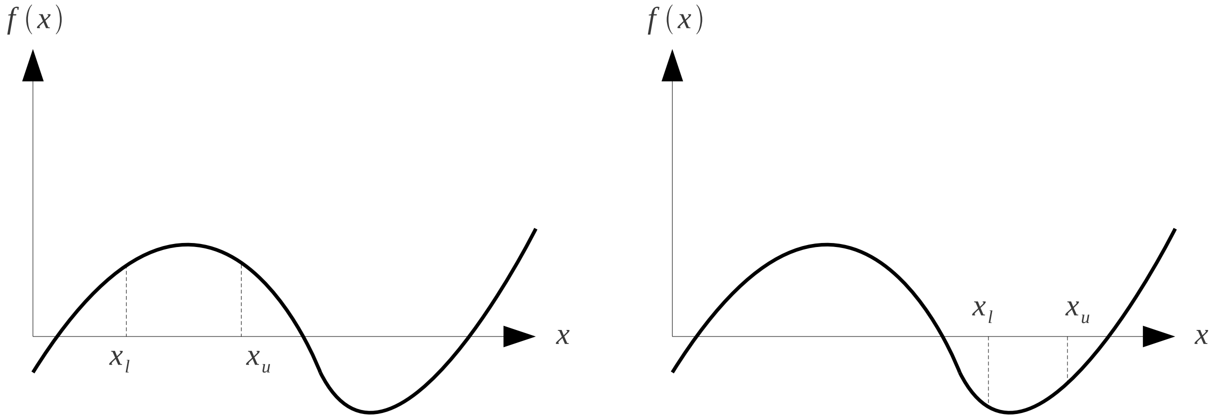


Figure 5.3: There may not be any roots between the two points.

for this example [46]. The task of the bisection method in the example at the first iteration is presented in Figure 5.6 as:

- Estimates,
$$x_m = (x_u + x_l)/2 = 0.055.$$

- Calculates,
$$f(x_l) = 0.0003993;\ f(x_u) = -0.0002662; f(x_m) = -6.655e - 05.$$

- Checks the root lies in (x_l, x_u) or (x_m, x_u).

- Chooses, $x_u = 0.11$ and $x_l = 0.055$.

Figure 5.7 shows the second iteration for this example. In this iteration, the algorithm task is:

- Estimates:
$$x_m = (x_u + x_l)/2 = 0.0825.$$

- Calculates,
$$f(x_l) = 6.655e - 05;\ f(x_u) = -0.0002662; f(x_m) = -0.00016222.$$

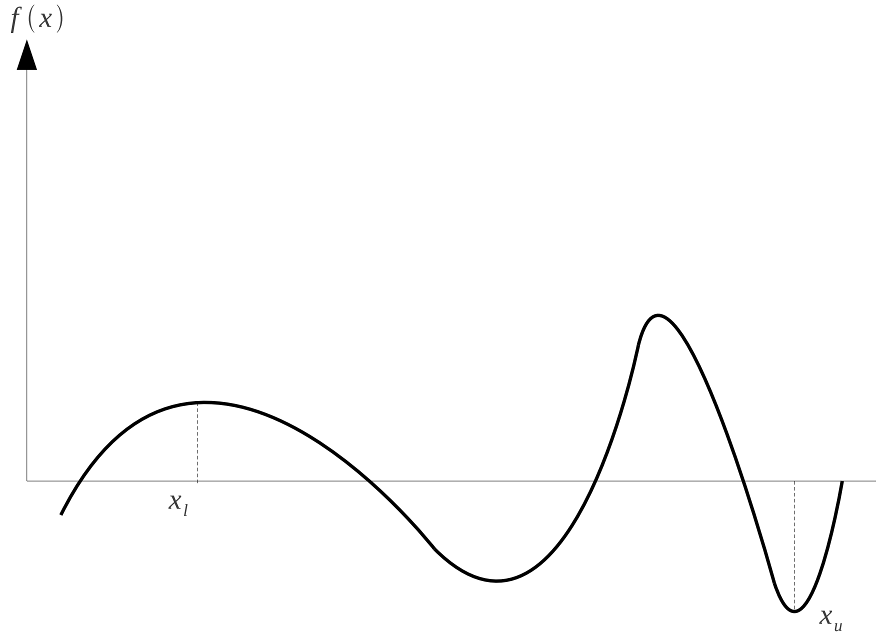


Figure 5.4: More than one root may exist between the two points.

- Approximates the error (absolutely relative),

$$ea = abs((x_m^{new} - x_m^{old})/x_m^{new}) = 0.3333.$$

- Checks the root lies in (x_l, x_u) or (x_m, x_u).
- Chooses, $x_u = 0.0825$ and $x_l = 0.055$.

Figure 5.8 shows the third iteration for this example. In this iteration, the algorithm task is:

- Estimates,

$$x_m = (x_u + x_l)/2 = 0.06875.$$

- Calculates,

$$f(x_l) = 6.655e{-}05;\ \ f(x_u) = -0.00016222; f(x_m) = -5.5632e{-}05.$$

- Approximates the error (absolutely relative),

$$ea = abs((x_m^{new} - x_m^{old})/x_m^{new}) = 0.2.$$

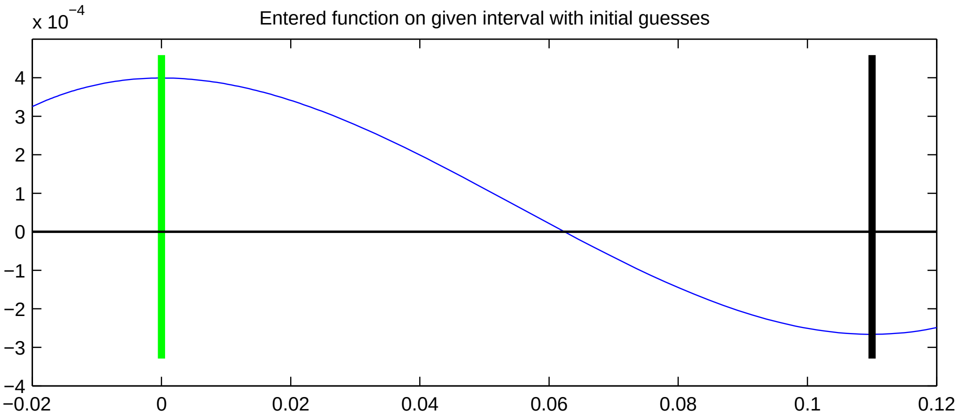


Figure 5.5: Bisection algorithm: Initial guess.

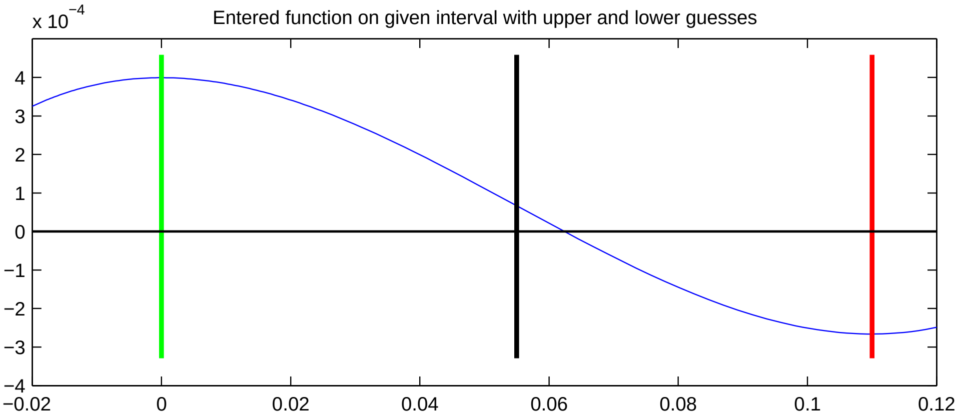


Figure 5.6: Bisection algorithm: First iteration.

- Checks the root lies in (x_l, x_u) or (x_m, x_u).
- Chooses, $x_u = 0.06875$ and $x_l = 0.055$.

5.3 Hidden Convexity GMMSE Detector

The solution of (5.1) is $\boldsymbol{x} = \boldsymbol{V}\boldsymbol{z}$ where, $\boldsymbol{z}^{(j)} = \boldsymbol{b}^{(j)}/(\boldsymbol{\lambda}^{(j)} + \eta), \boldsymbol{b} = \boldsymbol{V}^H\boldsymbol{y}$, and $\eta^* \geq \alpha$ is the unique root of $G(\eta)$ (given by the bisection method), then the GMMSE solution takes the form,

$$\boldsymbol{x}^*_{GMMSE} = \boldsymbol{V}(\Lambda + \eta^* I)^{-1}\boldsymbol{V}^H\boldsymbol{y}. \tag{5.6}$$

Using the circular approximation $\tilde{\boldsymbol{H}}$ of the banded Toeplitz channel

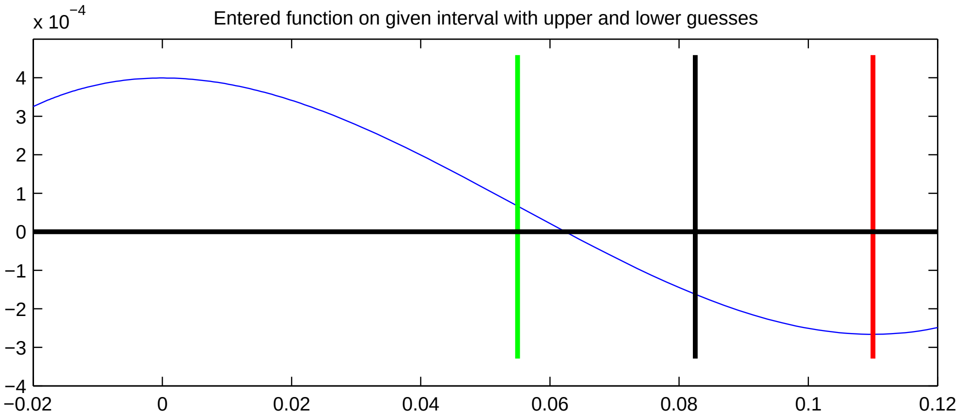


Figure 5.7: Bisection algorithm: Second iteration.

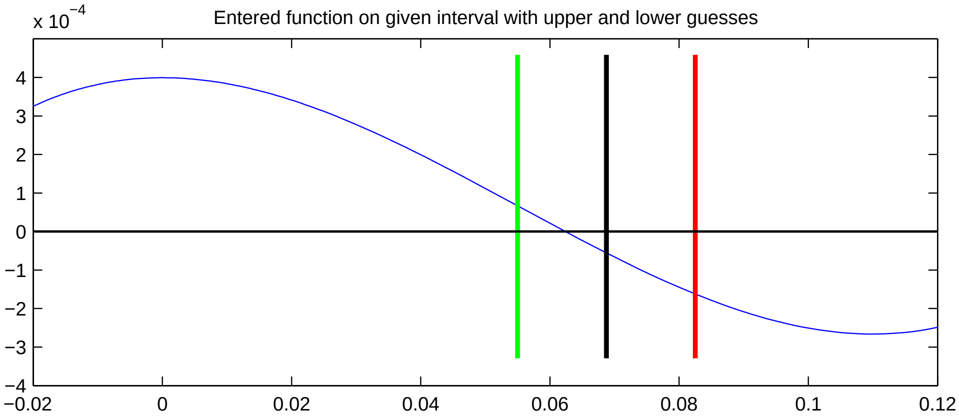


Figure 5.8: Bisection algorithm: Third iteration.

matrix $\boldsymbol{H}$, where $\tilde{\boldsymbol{H}}^H\tilde{\boldsymbol{H}} = \boldsymbol{F}^H\tilde{\boldsymbol{\Lambda}}\boldsymbol{F}$, $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{y}$, and $\boldsymbol{x} = \boldsymbol{F}^H\boldsymbol{z}$, the GMMSE solution is given by

$$\boldsymbol{x}^*_{GMMSE} = \boldsymbol{F}^H(\tilde{\boldsymbol{\Lambda}} + \eta^* I)^{-1}\boldsymbol{F}\boldsymbol{y}. \tag{5.7}$$

5.4 Simulations Results

We applied the presented algorithms to the same detection problems which are presented in the previous chapter. Figures 5.9 and 5.10 present the simulation results using the first channel, while Figures 5.11 and 5.12 are the results for the fading channel. $GMMSE_G$ refers to GMMSE solution using the gradient descent algorithm to get η^* and $GMMSE_B$ indicates of using bisection method to get η^*.

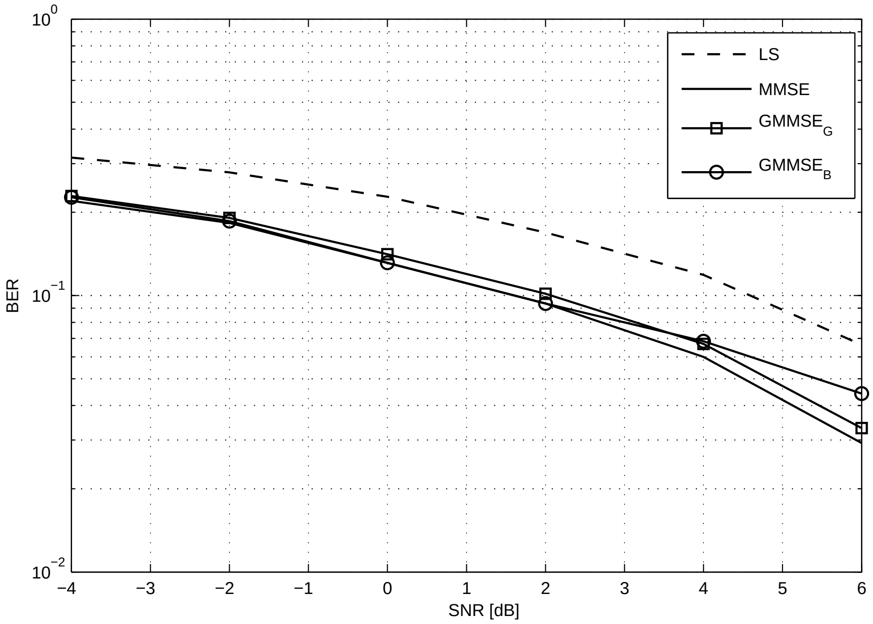


Figure 5.9: BER performance, using channel(1), $n = 1000$ with $L = 5$.

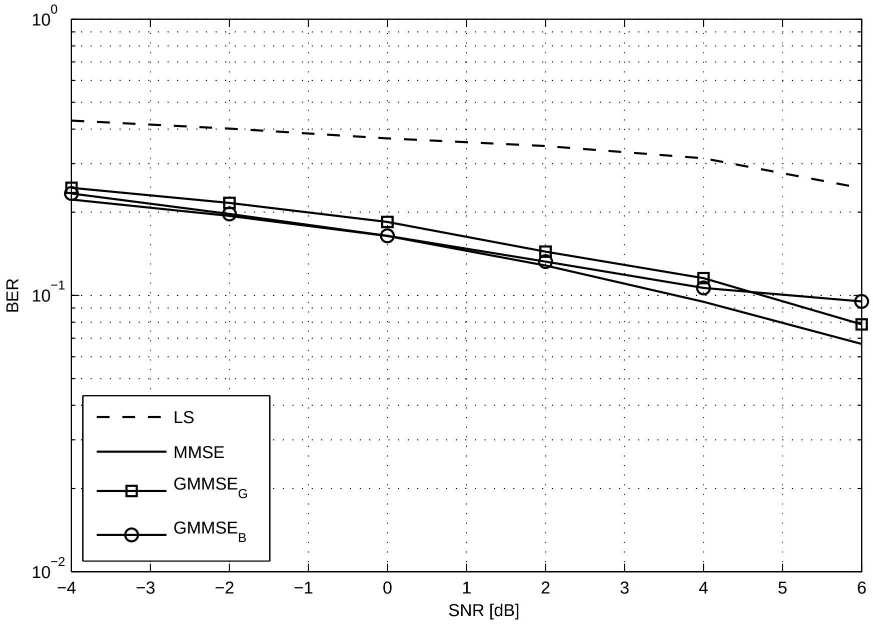


Figure 5.10: BER performance, using channel(1), $n = 1000$ with $L = 15$.

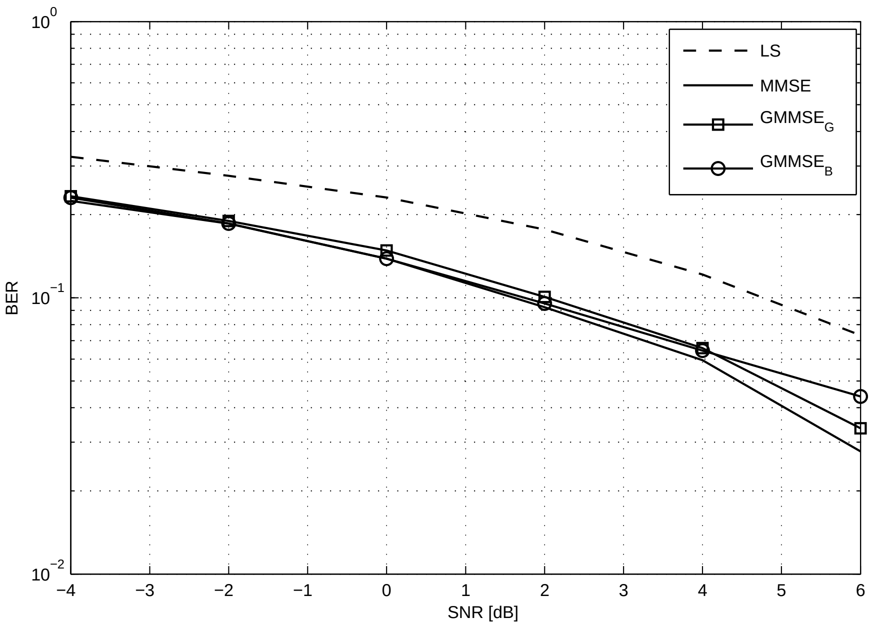


Figure 5.11: BER performance, using channel(2), $n = 1000$ with $L = 5$.

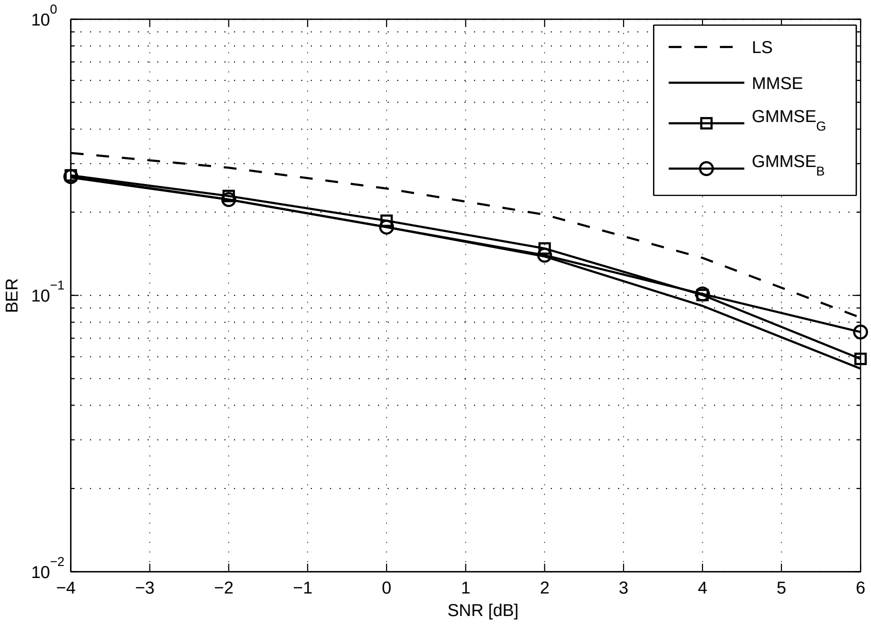


Figure 5.12: BER performance, using channel(2), $n = 1000$ with $L = 15$.

6 Near Optimum Detector and Computational Complexity

A structured GMMSE detector was presented in the last two chapters using two different convex relaxations. This detector has a bit error rate (BER) performance almost the same as that of MMSE detector with a lower computational complexity than the standard GMMSE detector (as we will see in this chapter).

However, the performance of the structured GMMSE is not close to the BER performance of the optimum Maximum likelihood (ML) detector. Combining the proposed relaxations with a local search algorithm results a detector whose performance is near to that of the ML receiver. Its computational complexity is of the same order as that of the proposed GMMSE detector.

6.1 Near Optimum Detector Using Local Search Algorithm

Local search is a mathematical method for solving computational hard optimization problems. It can be used for problems that can be formulated to find a solution among a number of candidate solutions. Local search algorithm moves from a local optimum to another in the search space until a global solution is found. Local search moves in Figure 6.1 from x_1 to x_3 passing through x_2. All these three points are local optimum, but only x_2 is the global one.

Problem (4.2) is non-convex NP-hard and there is no efficient algorithm to solve this problem. Therefore, we relaxed it to convex optimization problems (4.3) and (5.3). So, the resulting solution in each case is a solution of a relaxed problem not the solution of the original problem. The domain of the original problem is a subset of the domain

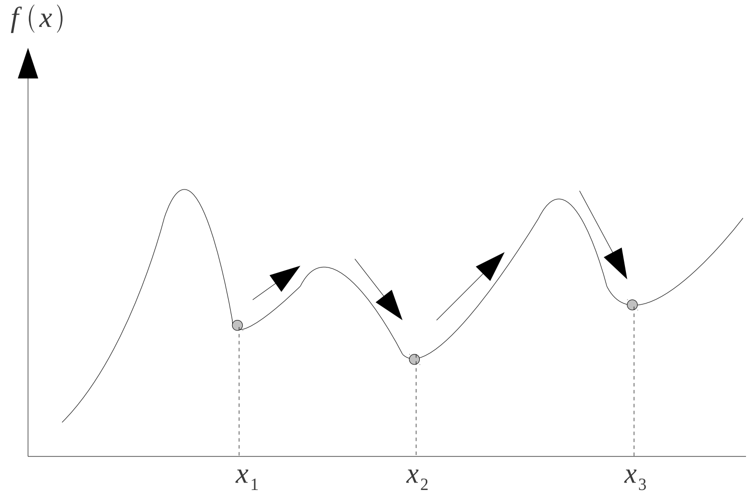


Figure 6.1: Local search moves from local optimum to another.

of its relaxation, hence, we can use the resulting solution of the relaxed problem to be a good initial guess for the local search method.

In this section, we apply the local search method to problem (4.2),

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x}\in\{-1,+1\}^n} \boldsymbol{x}^H \boldsymbol{H}^H \boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^H\boldsymbol{x},$$

which is non-convex optimization problem. Because of the non-convexity property, this problem may have more than one local solutions (convex problem has only one global solution). The solution in (4.22),

$$\boldsymbol{x}^*_{GMMSE} = \boldsymbol{F}^H \left(\tilde{\boldsymbol{\Lambda}} + \lambda^* \boldsymbol{I}\right)^{-1} \boldsymbol{F}\boldsymbol{y},$$

or the solution in (5.7),

$$\boldsymbol{x}^*_{GMMSE} = \boldsymbol{F}^H(\tilde{\boldsymbol{\Lambda}} + \eta^* I)^{-1}\boldsymbol{F}\boldsymbol{y},$$

can be improved by local search algorithm. The local search algorithm for a problem of the form $min_{\boldsymbol{x}\in\{-1,1\}^n} g(\boldsymbol{x})$ is defined as follows:

- Initial step: Choose an arbitrary guess $\boldsymbol{x}_0$

- General step: If there exists $\boldsymbol{z} \in \{-1, 1\}^n$, which is different from

$\boldsymbol{x}$ in one component and satisfies $g(\boldsymbol{z}) < g(\boldsymbol{x})$ then $\boldsymbol{x} := \boldsymbol{z}$, otherwise STOP.

We apply the method for both cases, Toeplitz and circular, but to avoid the redundancy we only present the circular case. In this case,

$$g(\boldsymbol{x}) = \boldsymbol{x}^H \boldsymbol{F}^H \tilde{\boldsymbol{\Lambda}} \boldsymbol{F} \boldsymbol{x} - 2\boldsymbol{y}^H \boldsymbol{x}$$

and the initial guess of the local search method $\boldsymbol{x}_0$ is chosen as the solution of (4.22) or (5.7). In each update of the algorithm to move from local minimum to another, the problem must satisfy the necessary optimality conditions [8]

$$\boldsymbol{X} \boldsymbol{F}^H \tilde{\boldsymbol{\Lambda}} \boldsymbol{F} \boldsymbol{X} \boldsymbol{e} - \boldsymbol{X} \boldsymbol{y} \leq Diag(\boldsymbol{H}^H \boldsymbol{H}),$$

where, $\boldsymbol{X} = diag(\boldsymbol{x})$ and $\boldsymbol{e} = (1, 1, ..., 1)^T$. We note that each iteration of the local search algorithm requires $O(n)$ operations.

6.2 Simulation Results

In this section, we apply the local search algorithm using the global optimization matlab toolbox to solve problem (4.2). The simulation has four scenarios:

- **Using channel (1)**: Choose initial guess using the gradient descent algorithm.

- **Using channel (2)**: Choose initial guess using the gradient descent algorithm.

- **Using channel (1)**: Choose initial guess using the bisection method.

- **Using channel (2)**: Choose initial guess using the bisection method.

In all scenarios, we used channel length $L = 15$ for both channel (1) and channel (2) as introduced in Section 4.7. We compare the bit error rate performance of the enhanced GMMSE detector (we refer it by NML) with the standard GMMSE and ML detectors. Figures 6.2 to 6.5 represent these four scenarios. The enhanced GMMSE (NML) has BER performance close to that of ML detector.

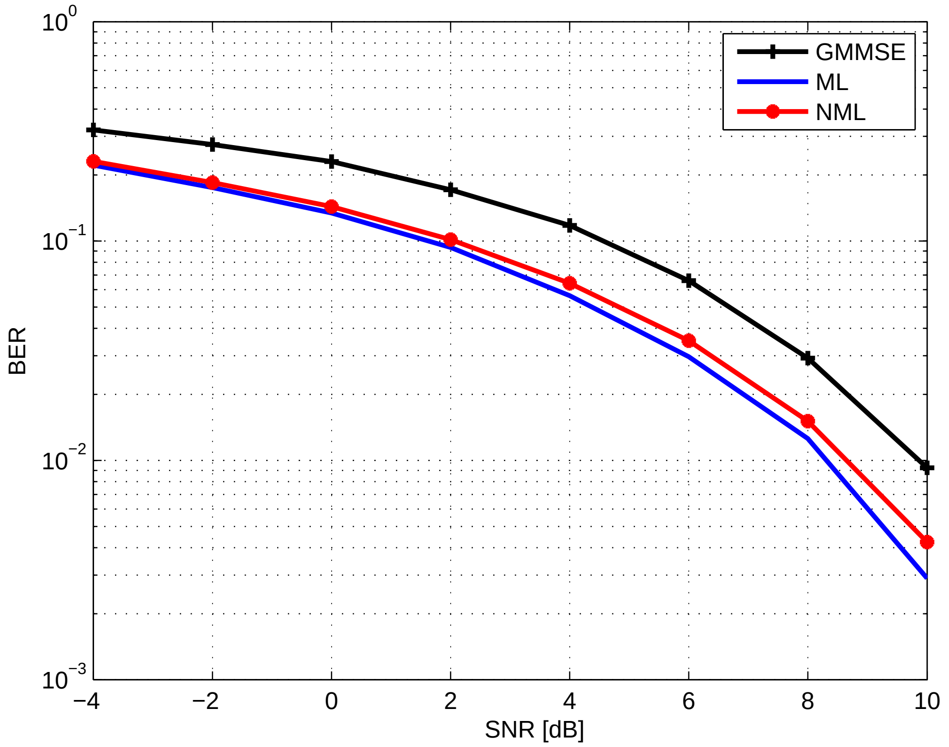


Figure 6.2: BER performance: Initial guess for local search is a solution of GMMSE using gradient descent algorithm for channel (1).

6.3 Computational Complexity

The computational complexity of the maximum likelihood detector is 2^n which is extremely high compared to the sub-optimum receivers. The MMSE detector is one example of these sub-optimum detectors, but it has a high computational complexity. Using the Toeplitz structure of the channel matrix enables us to approximate the problem by using the eigenvalue decomposition which reduce the computational complexity of the proposed detector. The eigenvalue decomposition itself requires an extra effort, so it also has a required computational complexity. To

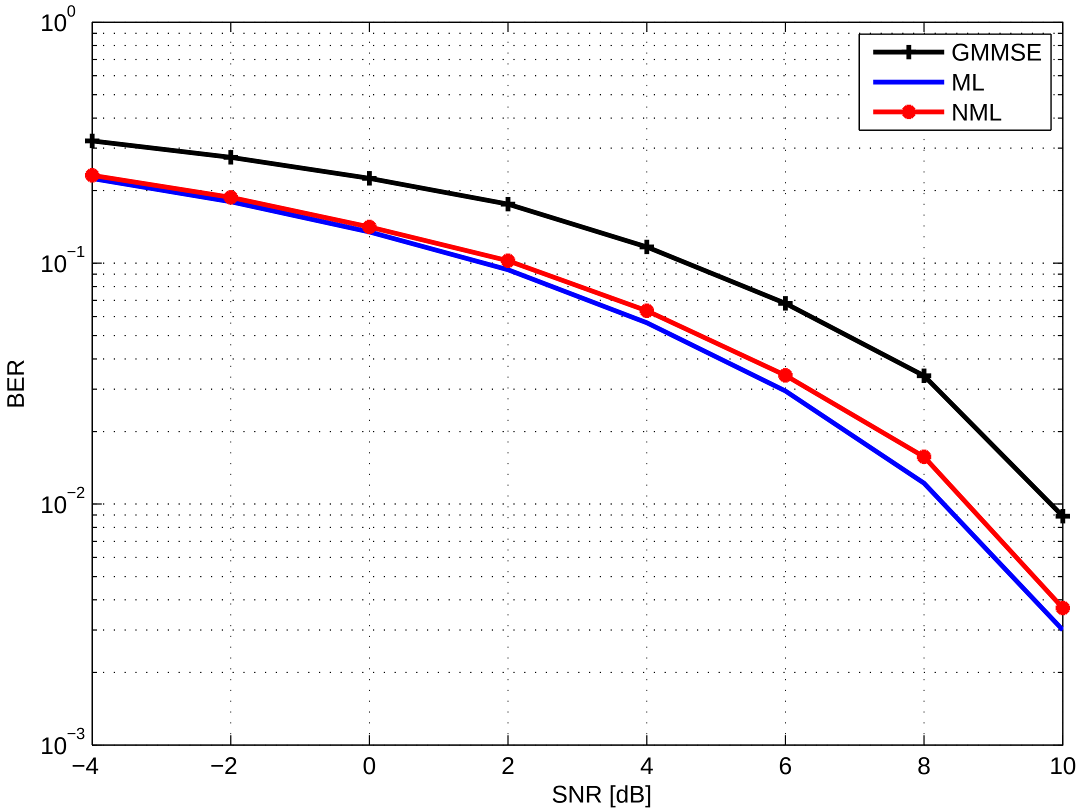


Figure 6.3: BER performance: Initial guess for local search is a solution of GMMSE using gradient descent algorithm for channel (2).

produce further reduction of the computational complexity, we approximated the Toeplitz structure by a circular structure which enables us to use the FFT decomposition.

We discuss the computational complexity of NML detector which is decomposed into two main parts:

- **Part** 1: The computational complexity to find the GMMSE solution (as the initial guess of the local search).

- **Part** 2: The computational complexity of the local search method.

Taking into account that, the complexity of part one is also composed in two sub-parts:

- **Part** 1a: The complexity of the solution of the system of equations (4.7), (4.14), (4.22), (5.6) or (5.7), which is the same as for MMSE.

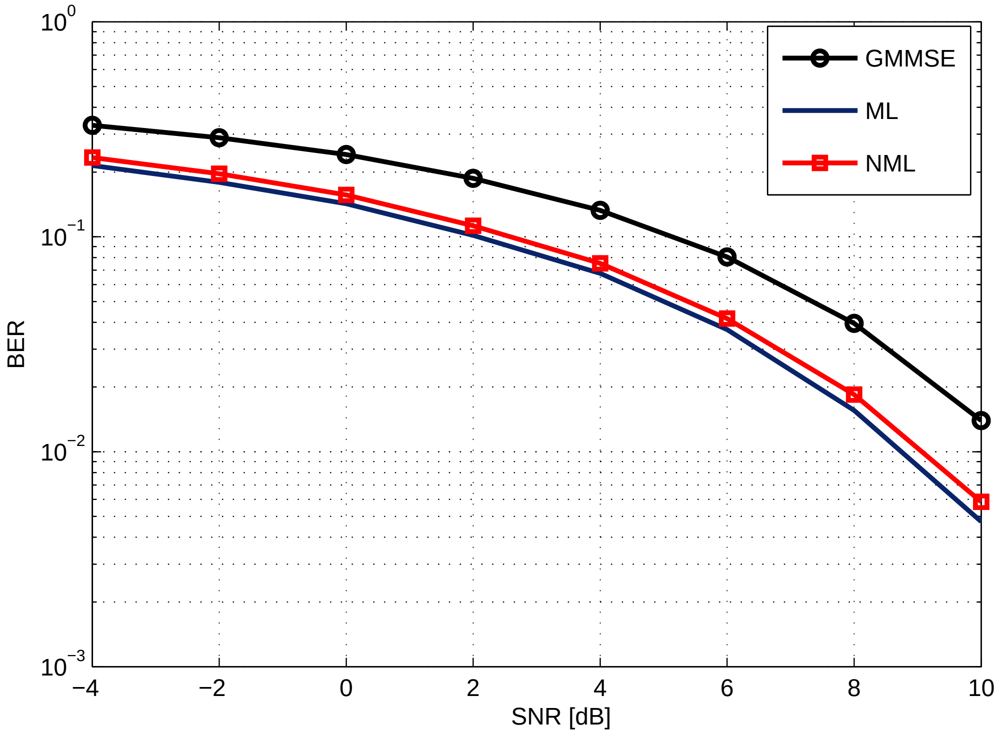


Figure 6.4: BER performance: Initial guess for local search is a solution of GMMSE using bisection method for channel (1).

- **Part** $1b$: The complexity of the iterations required for the gradient descent algorithm to find δ^* in case of (4.7), or to find λ^* in case of (4.14) or (4.22) and the complexity of the bisection method to find η^* in case of (5.6) or (5.7).

In part $1a$, the system is banded without structure then, the solution is obtained by the Cholesky algorithm with complexity $nL^2+8nL+n$ [30]. When there is a banded Toeplitz structure as in (4.14) or (4.22), the solution is given by the Schur algorithm with complexity $4Ln$ [31]. If we approximate the banded Toeplitz matrix by a banded circular structure as in ((5.6) and (5.7)), the solution is obtained using the FFT decomposition with complexity $\frac{3}{2}(n+L-1)log(n+L-1)+(n+L-1)$ [30]. Therefore, the circular approximation results in a significantly reduced computational complexity.

In part $1b$, the gradient descent algorithm adds some complexity. However, for the structured cases (4.13) or (4.21), the iterations of the gradient descent algorithm are only applied to diagonal matrices $(\mathbf{\Lambda})$ or $(\tilde{\mathbf{\Lambda}})$ such that the computational complexity is only of $O(n)$ per iteration. We solve problem (5.5) using the bisection method. This method

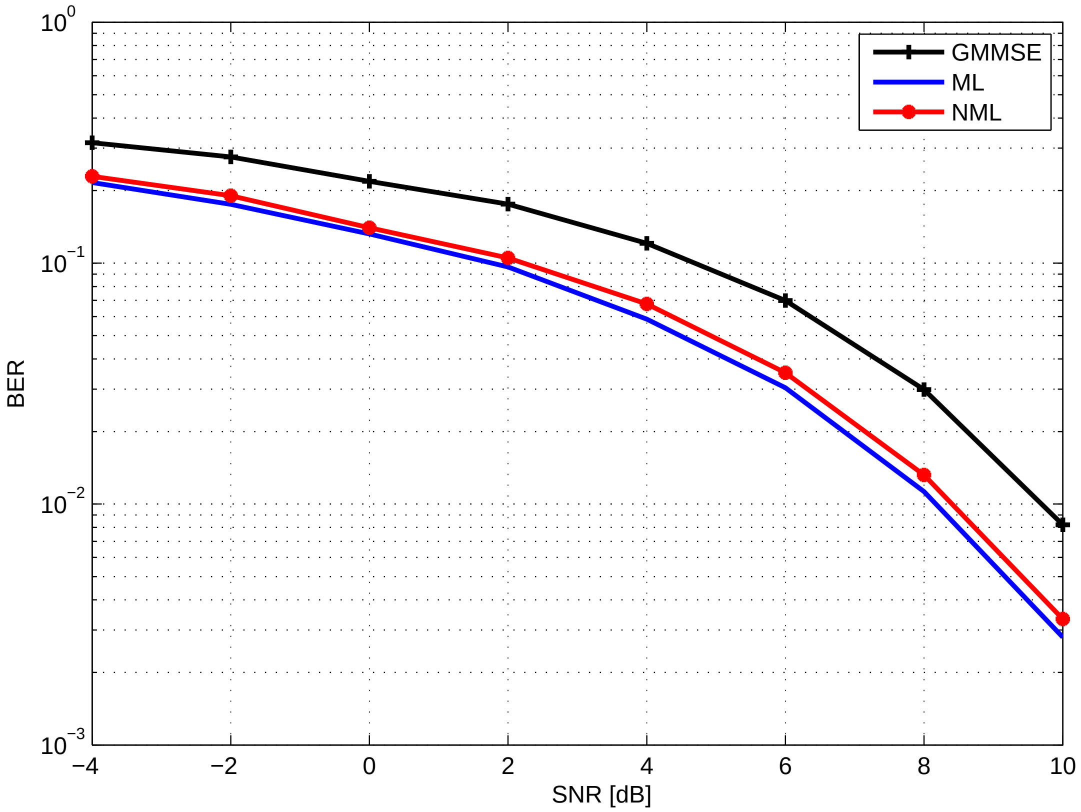


Figure 6.5: BER performance: Initial guess for local search is a solution of GMMSE using bisection method for channel (2).

is linear in the size of the problem so, it requires only $O(n)$. Figure 6.6 shows the mean number of iterations for the gradient descent algorithm (Part (a) for channel (1) and part (b) for channel (2)). Figure 6.7 shows the mean number of iterations for the bisection method (Part (a) for channel (1) and part (b) for channel (2)). In both cases, the number of iterations is small. Therefore, the complexity can be neglected compared to the complexity of part $1a$.

In part 2, the local search algorithm has a complexity of $O(n)$. The overall complexity of the proposed NML detector is given in Figure 6.8 and Figure 6.9. We note that, the curves that represent the banded circular case in both two figures are almost congruent.

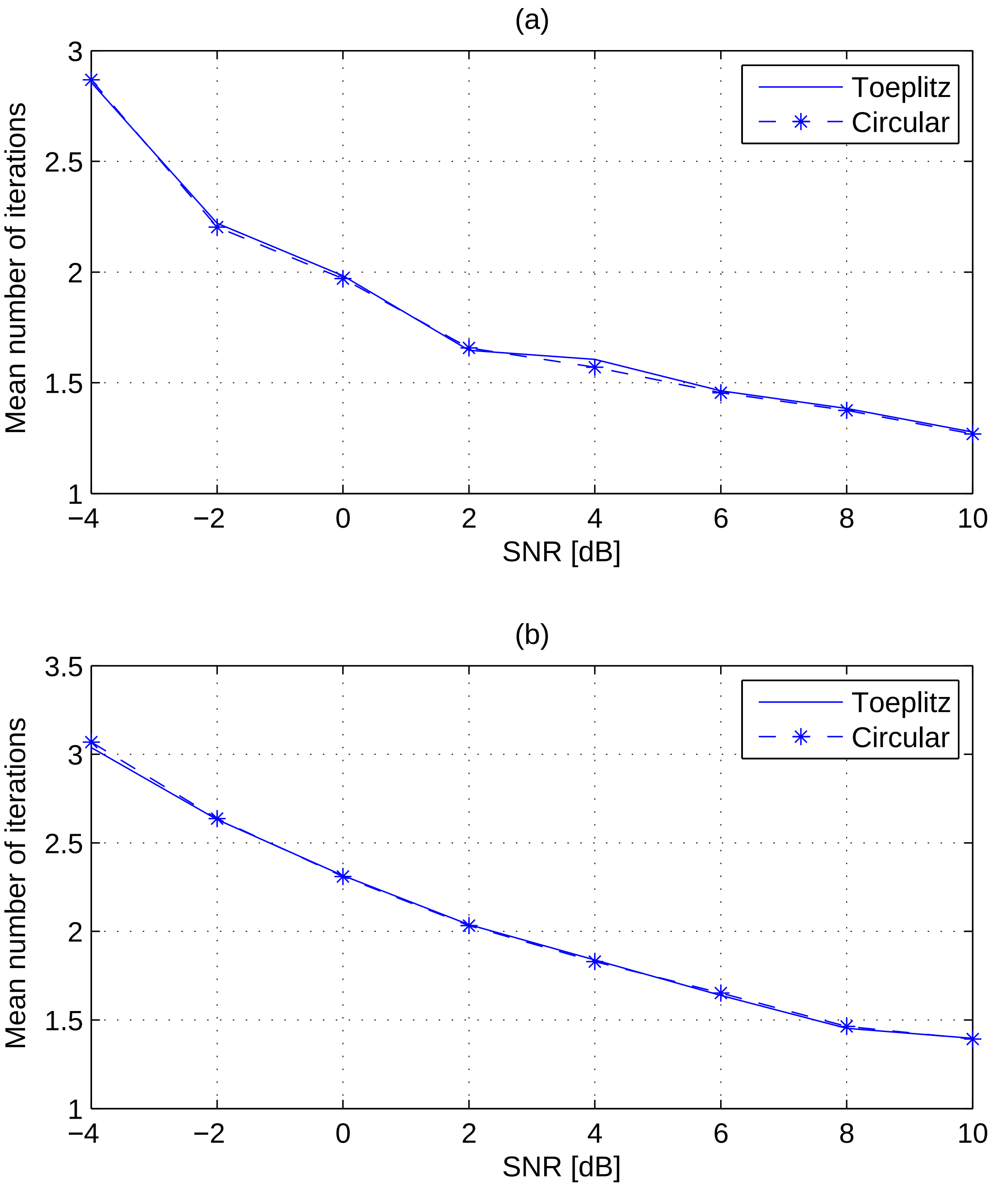


Figure 6.6: Mean number of iterations for gradient descent algorithm, (a) for channel (1) and (b) for channel (2).

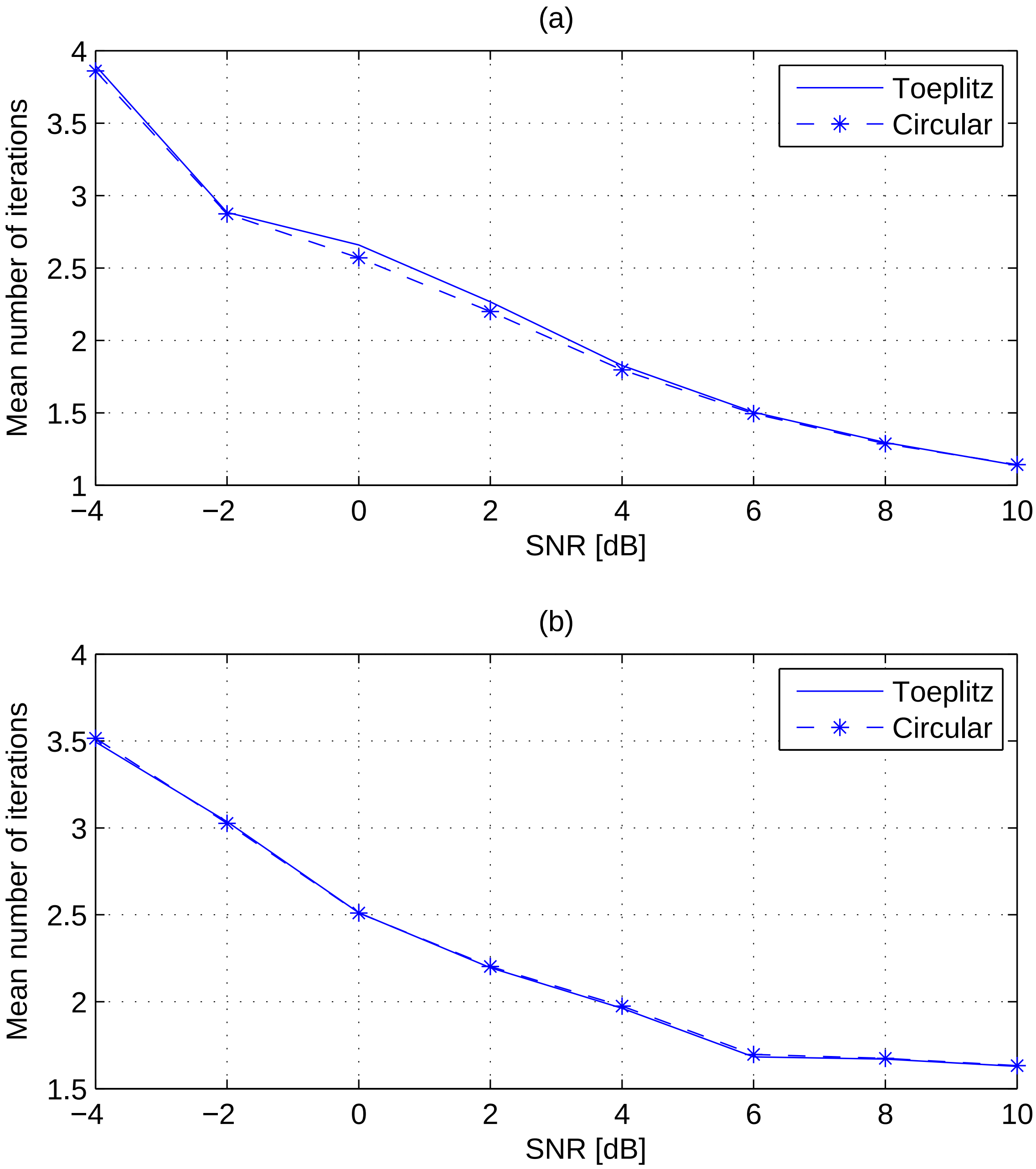


Figure 6.7: Mean number of iterations for bisection method, (a) for channel (1) and (b) for channel (2).

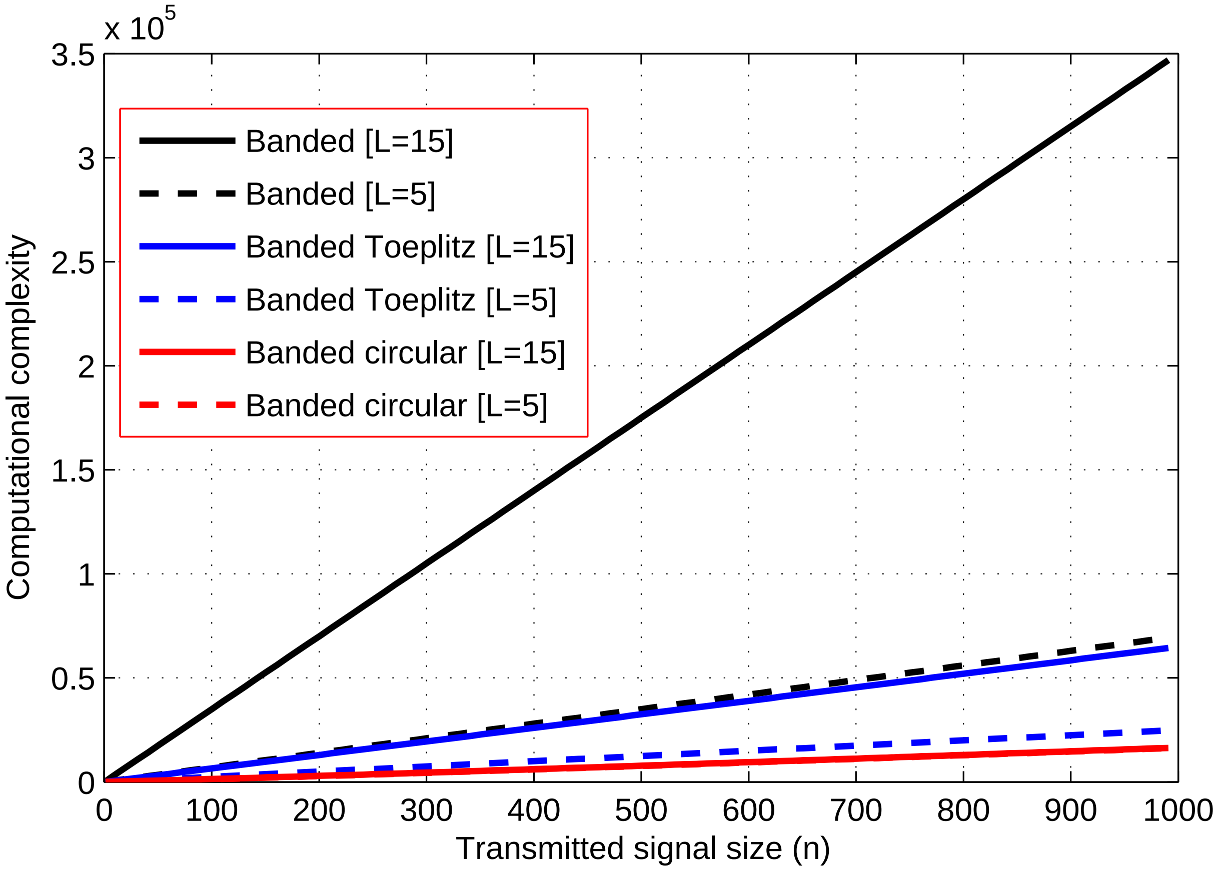


Figure 6.8: The overall computational complexity when the transmitted signal size varies between 1 and 1000 with two different channel lengths, $L = 15$ and $L = 5$.

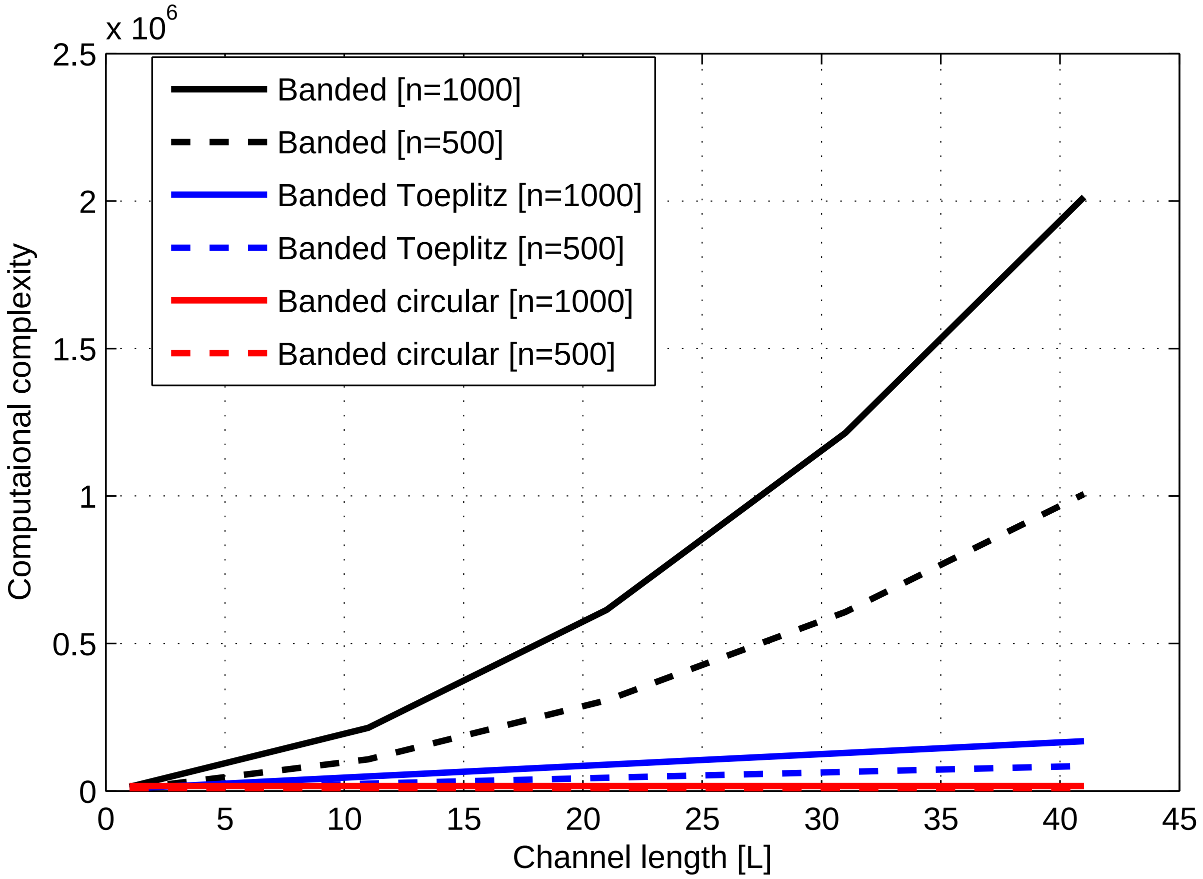


Figure 6.9: The overall computational complexity when the channel length varies between 1 and 40 with two different transmitted signal sizes, $n = 1000$ and $n = 500$.

7 Conclusions

Approaches to solve the detection problem for wireless communication system using convex optimization were presented. The resulting near optimum Maximum Likelihood (NML) detector has significantly less computational complexity than ML. Binary Phase Shift Keying (BPSK) and Quadrature Shift Keying (QPSK) constellation sets were relaxed into convex sets. Using these relaxations the ML problem was relaxed to convex optimization problems. For solving the convex optimization problems Generalized Minimum Mean Squared Error (GMMSE) algorithms were presented. The dual problem of the relaxed problem was solved using the gradient descent algorithm and the bisection method respectively. Reducing the computational complexity was achieved by using the structure of the channel matrix.

First, the banded Toeplitz structure of the channel matrix was used. Using Eigenvalue Decomposition (EVD) of the banded channel matrix the algorithms were executed on diagonal matrices which reduced the number of multiplication operations. The computational complexity was reduced from $nL^2 + 8nL + n$ for the banded channel matrix (without structure) by applying the Cholesky algorithm to $4Ln$ by applying the Schur algorithm for the banded channel matrix with Toeplitz structure.

Second, circular approximation of this banded Toeplitz channel matrix was used. The dual problem was again solved using the above mentioned algorithms. EVD of circular matrices, which can be determined by using the Fast Fourier Transform (FFT) significantly reduced the computational complexity. So, using the fact that the computational complexity of the FFT is $O(n\ log\ n)$, the computational complexity of the proposed GMMSE solution using circular matrices is $\frac{3}{2}(n + L - 1)log(n + L - 1) + (n + L - 1)$.

In addition to this complexity, the complexity of both gradient descent algorithm or the bisection method was taken into consideration. This computational complexity in both cases was of $O(n)$ per iteration.

The difference between these two algorithms is that: The initial step in gradient descent algorithm is perfectly chosen while the initial step of the bisection method depends on the bracketing the domain of solution which may take more iterations than the gradient descent algorithm. Both algorithms, in general, only require a small number of iterations to solve the dual problem, so their computational complexity can be neglected compared to the complexity of the entire GMMSE algorithms.

Third, the local search algorithm was used to enhance the solution of the proposed GMMSE detector. The relaxed problem is a convex optimization problem, but the original problem is not. So the proposed GMMSE solution is a solution of the relaxed problem (which is global only for the relaxed problem) and may or may not be global for the original problem. The initial guess of the local search algorithm was taken as the solution of the proposed (GMMSE) solution. Then, the local search algorithm was applied to the original problem with its well-known strategy that it moves within the domain of solution from local optimum to another until the optimality conditions are satisfied. The computational complexity of the local search algorithm is of $O(n)$ which is also neglected compared to the overall solutions complexity. The result is a Near Optimum ML (NML) detector with significantly reduced computational complexity.

Further research can be done to solve the Maximum Likelihood (ML) detection problem using Semidefinite Programming Relaxation (SDR). This method has BER performance almost the same as ML, but it has a computationally complex solution. As we have seen in this work, using the structure of the channel matrix is a good way to reduce the complexity. Applying the presented technique combined with SDR is appealing. Furthermore the methodologies and algorithms that were proposed in this thesis can be applied to Code Division Multiple Access (CDMA) and Orthogonal Frequency Division Multiplex (OFDM).

A Notation and Abbreviations

Notation

x	Scalars
X	Constant, scalar system parameters
$\boldsymbol{x}$	Vectors
$\boldsymbol{X}$	Matrices
X	Sets
$\boldsymbol{x}^{(i)}$	The ith element of vector $\boldsymbol{x}$
$\boldsymbol{X}_{i,j}$	The element in row i, column j of matrix $\boldsymbol{X}$
$\boldsymbol{A}^H$	Conjugate transpose of $\boldsymbol{A}$
$\boldsymbol{A}^T$	Transpose of $\boldsymbol{A}$
$\boldsymbol{A}^{-1}$	Inverse of $\boldsymbol{A}$
$\boldsymbol{A}^{\dagger}$	Pseudo-inverse of $\boldsymbol{A}$
$\mathrm{E}\{\cdot\}$	Expectation operator
$\mathrm{diag}(a)$	Diagonal matrix with elements of a on diagonal

Abbreviations

AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CDMA	Code Division Multiple Access
DFT	Discrete Fourier Transform
FFT	Fast Fourier Transform

GMMSE	Generalized Minimum Mean Square error
KKT	Karush-Kuhn-Tucker
LP	Linear Program
LS	Least Square
MAP	Maximum a Posteriori Estimate
MIMO	Multiple-Input, Multiple-Output
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MS	Mean Square
NP	Non-deterministic Polynomial
OFDM	Orthogonal Frequency Division Multiplex
Pdf	Probability density function
PSD	Positive Semidefinite
PSK	Phase Shift Keying
QP	Quadratic Programming
QPSK	Quadrature Phase Shift Keying
SDR	Semidefinite Relaxation
SNR	Signal-to-Noise Ratio

Bibliography

[1] *Matlab2010a. The mathworks inc. Technical report, Natick, Massachusetts, 2010.*

[2] B. Alkire and L. Vandenberghe. Handling nonnegative constraints in spectral estimation. In *Conference Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers, 2000*, volume 1, pages 202–206. IEEE, 2000.

[3] B. Alkire and L. Vandenberghe. Interior-point methods for magnitude filter design. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001.*, volume 6, pages 3821–3824. IEEE, 2001.

[4] W.U. Bajwa, J.D. Haupt, G.M. Raz, S.J. Wright, and R.D. Nowak. Toeplitz-structured compressed sensing matrices. In *IEEE/SP 14th Workshop on Statistical Signal Processing, 2007*, pages 294–298. IEEE, 2007.

[5] Mourad Barkat. *Signal Detection and Estimation.* ARTECH HOUSE, INC, 2005.

[6] Jonathan M. Barwein and Andrian S. Lewis. *Convex Analysis and Nonlinear Optimization Theory and Examples.* New York: Springer-Verlag, second edition edition, 2000.

[7] M.S. Bazaraa. *Nonlinear Programming Theory and Algorithms.* 1994.

[8] A. Beck and M. Teboulle. Global optimality conditions for quadratic optimization problems with binary constraints. *SIAM Journal on Optimization*, 11:179, 2000.

[9] A. Ben-Tal and M. Teboulle. Hidden convexity in some nonconvex quadratically constrained quadratic programming. *Mathematical Programming*, 72(1):51–63, 1996.

[10] J.M. Borwein and A.S. Lewis. *Convex analysis and nonlinear optimization: theory and examples*, volume 3. Springer Verlag, 2006.

[11] S.P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2003.

[12] R.N. Bracewell. *The fourier transform and its applications 3rd Ed.* 2000.

[13] E.O. Brigham and RE Morrow. The fast fourier transform. *IEEE Spectrum*, 4(12):63–70, 1967.

[14] L. Brunel. Optimum and sub-optimum multiuser detection based on sphere decoding for multi-carrier code division multiple access systems. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 3, pages 1526–1530. IEEE, 2002.

[15] X.W. Chang and Q. Han. Solving box-constrained integer least squares problems. *IEEE Transactions on Wireless Communications*, 7(1):277–287, 2008.

[16] J.K. Cullum and R.A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations: Theory*, volume 1. Society for Industrial Mathematics, 2002.

[17] M.O. Damen, K. Abed-Meraim, and J.C. Belfiore. Generalised sphere decoder for asymmetrical space-time communication architecture. *Electronics letters*, 36(2):166–167, 2000.

[18] M.O. Damen, A. Safavi, and K. Abed-Meraim. On cdma with space-time codes over multipath fading channels. *IEEE Transactions on Wirless Communications*, 2(1):11–19, 2003.

[19] T.N. Davidson, Z.Q. Luo, and J.F. Sturm. Linear matrix inequality formulation of spectral mask constraints. In *Pro-*

ceedings.(ICASSP'01). 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001., volume 6, pages 3813–3816. IEEE, 2001.

[20] T.N. Davidson, Z.Q. Luo, and K.M. Wong. Design of orthogonal pulse shapes for communications via semidefinite programming. *IEEE Transactions on Signal Processing*, 48(5):1433–1445, 2000.

[21] P.J. Davis. Circulant matrices. a wiley-interscience publication. *Pure and Applied Mathematics. John Wiley and Sons, New York-Chichester-Brisbane*, 1979.

[22] P.J. Davis. *Circulant matrices.* Chelsea Pub Co, 1994.

[23] R. de Miguel and RR Miiller. Convex precoding for vector channels in high dimensions. In *IEEE International Zurich Seminar on Communications, 2008*, pages 120–123. IEEE, 2008.

[24] Z. Ding and Z.Q. Luo. A fast linear programming algorithm for blind equalization. *IEEE Transactions on Communications*, 48(9):1432–1436, 2000.

[25] A. Duel-Hallen. Decorrelating decision-feedback multiuser detector for synchronous code-division multiple-access channel. *IEEE Transactions on Communications*, 41(2):285–290, 1993.

[26] B. Dumitrescu, I. Tabus, and P. Stoica. On the parameterization of positive real sequences and ma parameter estimation. *IEEE Transactions on Signal Processing*, 49(11):2630–2639, 2001.

[27] Y.C. Eldar and A. Beck. Hidden convexity based near maximum-likelihood cdma detection. In *IEEE 6th Workshop on Signal Processing Advances in Wireless Communications, 2005*, pages 61–65. IEEE, 2005.

[28] U. Fincke and M. Pohst. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. *Mathematics of computation*, pages 463–471, 1985.

[29] I. Gohberg and V. Olshevsky. Complexity of multiplication with

vectors for structured matrices. *Linear Algebra and Its Applications*, 202:163–192, 1994.

[30] G.H. Golub and C.F. Van Loan. *Matrix computations*, volume 3. Johns Hopkins Univ Pr, 1996.

[31] J. Götze and H. Park. Schur-type methods based on subspace considerations. Citeseer, 1997.

[32] R.M. Gray. *Toeplitz and circulant matrices: A review.* Now Pub, 2006.

[33] U. Grenander and G. Szegő. *Toeplitz forms and their applications.* Chelsea Pub Co, 1984.

[34] F. Halsall. *Data Communications, Computer Networks, and open systems.* Addison-Wesley, 1996.

[35] P. Hansen. Methods of nonlinear 0-1 programming. *Annals of Discrete Mathematics*, 5:53–70, 1979.

[36] S. Haykin. *Adaptive filter theory (ISE).* 2003.

[37] C. Helmberg and F. Rendl. Solving quadratic (0, 1)-problems by semidefinite programs and cutting planes. *Mathematical Programming*, 82(3):291–315, 1998.

[38] C. Helmberg, F. Rendl, R.J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on Optimization*, 6(2):342–361, 1996.

[39] H. Hindi. A tutorial on convex optimization ii: Duality and interior point methods. In *American Control Conference*, 2006.

[40] M. Honig, U. Madhow, and S. Verdu. Blind adaptive multiuser detection. *IEEE Transactions on Information Theory*, 41(4):944–960, 1995.

[41] T. Hurlimann. *Mathematical Modeling and Optimization: An Essay for the Design of Computer-Based Modeling Tools.* Kluwer

Academic Publishers, 1999.

[42] O. Johnson. *Information theory and the central limit theorem.* World scientific publishing company, 2004.

[43] M. Hall T. Korhonen K. Ruttik, M. Honkanen and V. Porra. A wideband radio channel model for simulation of chaotic communicatio systems. In *ECCTD'97*, 1997.

[44] J. Kallrath. *Modeling languages in mathematical optimization*, volume 88. Springer, 2004.

[45] A. Kaw, N. Collier, M. Keteltas, J. Paul, and G. Besterfield. Holistic but customized resources for a course in numerical methods. *Computer Applications in Engineering Education*, 11(4):203–210, 2003.

[46] Autar K. Kaw and Egwu Eric Kalu. *Numerical Methods with Applications.* http://www.autarkaw.com, 2011.

[47] M. Kisialiou, X. Luo, and Z.Q. Luo. Efficient implementation of quasi-maximum-likelihood detection based on semidefinite relaxation. *IEEE Transactions on Signal Processing*, 57(12):4811–4822, 2009.

[48] M. Kisialiou and Z.Q. Luo. Performance analysis of quasi-maximum-likelihood detector based on semi-definite programming. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP'05)*, volume 3, pages 433–436. IEEE, 2005.

[49] R.L. Kosut, W. Chung, C.R. Johnson Jr, and S.P. Boyd. On achieving reduced error propagation sensitivity in dfe design via convex optimization. In *Proceedings of the 39th IEEE Conference on Decision and Control, 2000*, volume 5, pages 4320–4323. IEEE, 2000.

[50] Harry L. and Van Trees. *Detection, Estimation, and Modulation Theory Part I.* JOHN WILEY & SONS, INC, 2001.

[51] P. Lancaster and M. Tismenetsky. *The theory of matrices: with applications*. Academic Pr, 1985.

[52] A. H. Land and A. G. Doig. An automatic method for solving discrete programming problems. *Econometrica*, 28:497–520, 1960.

[53] H. Lebret and S. Boyd. Antenna array pattern synthesis via convex optimization. *IEEE Transactions on Signal Processing*, 45(3):526–532, 1997.

[54] L. Li, Z.Q. Luo, T.N. Davidson, K.M. Wong, and E. Bossé. Robust filtering via semidefinite programming with applications to target tracking. *SIAM Journal on Optimization*, 12(3):740–755, 2002.

[55] D.G. Luenberger and Y. Ye. *Linear and nonlinear programming*, volume 116. Springer Verlag, 2008.

[56] J. Luo, K. Pattipati, P. Willett, and L. Brunel. Branch-and-bound-based fast optimal algorithm for multiuser detection in synchronous cdma. In *ICC'03. IEEE International Conference on communications, 2003*, volume 5, pages 3336–3340. IEEE, 2003.

[57] Z. Luo, W. Ma, A.M.C. So, Y. Ye, and S. Zhang. Semidefinite relaxation of quadratic optimization problems. *IEEE Signal Processing Magazine*, 27(3):20–34, 2010.

[58] Z.Q. Luo. Applications of convex optimization in signal processing and digital communication. *Mathematical programming*, 97(1):177–207, 2003.

[59] Z.Q. Luo, T.N. Davidson, G.B. Giannakis, and K.M. Wong. Transceiver optimization for multiple access through isi channels. In *PROCEEDINGS OF THE ANNUAL ALLERTON CONFERENCE ON COMMUNICATION CONTROL AND COMPUTING*, volume 39, pages 855–856. The University; 1998, 2001.

[60] Z.Q. Luo and W. Yu. An introduction to convex optimization for communications and signal processing. *IEEE Journal on Selected Areas in Communications*, 24(8):1426–1438, 2006.

[61] R. Lupas and S. Verdu. Linear multiuser detectors for synchronous code-division multiple-access channels. *IEEE Transactions on Information Theory*, 35(1):123–136, 1989.

[62] K. Abed-Meraim M. O. Damen and J. C. Belfiore. Sphere decoding of space-time codes. In *ISIT*, 2000.

[63] W.K. Ma, P.C. Ching, and Z. Ding. Semidefinite relaxation based multiuser detection for m-ary psk multiuser systems. *IEEE Transactions on Signal Processing*, 52(10):2862–2872, 2004.

[64] W.K. Ma, T.N. Davidson, K.M. Wong, Z.Q. Luo, and P.C. Ching. Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous cdma. *IEEE Transactions on Signal Processing*, 50(4):912–922, 2002.

[65] W.K. Ma, C.C. Su, J. Jaldén, T.H. Chang, and C.Y. Chi. The equivalence of semidefinite relaxation mimo detectors for higher-order qam. *IEEE Journal of Selected Topics in Signal Processing*, 3(6):1038–1052, 2009.

[66] W.K. Ma, B.N. Vo, T.N. Davidson, and P.C. Ching. Blind ml detection of orthogonal space-time block codes: Efficient high-performance implementations. *IEEE Transactions on Signal Processing*, 54(2):738–751, 2006.

[67] U. Madhow and M.L. Honig. Mmse interference suppression for direct-sequence spread-spectrum cdma. *IEEE Transactions on Communications*, 42(12):3178–3188, 1994.

[68] B. Mariere, Z.Q. Luo, and T.N. Davidson. Blind constant modulus equalization via convex optimization. *IEEE Transactions on Signal Processing*, 51(3):805–818, 2003.

[69] A. Mobasher, M. Taherzadeh, R. Sotirov, and A.K. Khandani. A near-maximum-likelihood decoding algorithm for mimo systems based on semi-definite programming. *IEEE Transactions on Information Theory*, 53(11):3869–3886, 2007.

[70] T. Morsy and J. Götze. Reducing complexity of generalized min-

imum mean square error detection. In *EUSIPCO2010 (European signal processing conference)*, 2010.

[71] T. Morsy and J. Götze. Near optimum maximum likelihood detector for structured communication problems. In *Wireless Telecommunications Symposium (WTS2012)*, London, UK, April 2012.

[72] T. Morsy, J. Götze, and H. Nassar. Convex programming for detection in structured communication problems. *Advances in Radio Science*, 8:307–312, 2010.

[73] T. Morsy, J. Gotze, and H. Nassar. Using hidden convexity in structured communication problems. In *IEEE 30th International Performance Computing and Communications Conference (IPCCC), 2011*, pages 1–7. IEEE, 2011.

[74] T. Morsy, K. Hueske, and J. Gotze. Analysis of a reduced complexity generalized minimum mean square error detector. In *2010 7th International Symposium on Wireless Communication Systems (ISWCS)*, pages 496–500. IEEE, 2010.

[75] A. NASH. *Linear and nonlinear programming (ISE)*, volume 67. 1996.

[76] L.B. Nelson and H.V. Poor. Iterative multiuser receivers for cdma channels: An em-based approach. *IEEE Transactions on Communications*, 44(12):1700–1710, 1996.

[77] J. Nocedal and S.J. Wright. *Numerical optimization.* Springer verlag, 1999.

[78] H.J. Nussbaumer. Fast fourier transform and convolution algorithms. *Berlin and New York, Springer-Verlag(Springer Series in Information Sciences.*, 2, 1982.

[79] V. Olshevsky. *Structured Matrices in Mathematics, Computer Science, and Engineering I*, volume 1. 2001.

[80] A.V. Oppenheim, R.W. Schafer, and J.R. Buck. *Discrete-Time Signal Processing, 1989.* Prentice-Hall, 1999.

[81] J.G. Proakis. *Digital communications*, volume 1221. McGraw-hill, 1987.

[82] J.G. Proakis and D.G. Manolakis. *Digital signal processing*, volume 1. 1996.

[83] R.T. Rockafellar. *Convex analysis*, volume 28. Princeton Univ Pr, 1997.

[84] Louis L. Scharf. *Statistical Signal Processing, Detection, Estimation, and Time Series Analysis.* Addison-Wesley Publishing Company, Inc., 1991.

[85] C.V. Sinn and J. Götze. Computationally efficient block transmission systems with and without guard periods. *Signal Processing*, 87(6):1421–1433, 2007.

[86] V. Sinn. *Efficient Block Transmission Systems for High Speed Wireless Communications.* PhD thesis, 2005.

[87] J.O. Smith III. *Mathematics of the Discrete Fourier Transform: with Audio Applications.* Booksurge Llc, 2007.

[88] J.A. Snyman. *Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms*, volume 97. Springer Verlag, 2005.

[89] W. Stalling. *Wireless communications and networking.* Prentic Hall, 2002.

[90] W. Stallings. *Data and computer communications.* Prentice Hall, 1997.

[91] B. Steingrimsson, Z.Q. Luo, and K.M. Wong. Soft quasi-maximum-likelihood detection for multiple-antenna wireless channels. *IEEE Transactions on Signal Processing*, 51(11):2710–2719, 2003.

[92] P. Stoica, T. McKelvey, and J. Mari. Ma estimation in polynomial time. *IEEE Transactions on Signal Processing*, 48(7):1999–2012,

2000.

[93] P.H. Tan and L.K. Rasmussen. The application of semidefinite programming for detection in cdma. *IEEE Journal on Selected Areas in Communications*, 19(8):1442–1449, 2001.

[94] E. Telatar. Capacity of multi-antenna gaussian channel. *European transactions on telecommunications*, 10:585–595, 1995.

[95] J. Tugan and PP Vaidyanathan. Globally optimal two channel fir orthonormal filter banks adapted to the input signal statistics. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, 1998.*, volume 3, pages 1353–1356. IEEE, 1998.

[96] J.B.H. Urruty and C. Lemaréchal. *Convex analysis and minimization algorithms*, volume 1. Springer, 1996.

[97] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM review*, pages 49–95, 1996.

[98] M.K. Varanasi and B. Aazhang. Multistage detection in asynchronous code-division multiple-access communications. *IEEE Transactions on Communications*, 38(4):509–519, 1990.

[99] S. Verdu. Computational complexity of multiuser detection. *Algorithmica*, 4:303–312, 1989.

[100] S. Verdu. *Multiuser detection.* Cambridge Univ Pr, 1998.

[101] E. Viterbo and J. Boutros. A universal lattice code decoder for fading channels. *IEEE Transactions on Information Theory*, 45(5):1639–1642, 1999.

[102] M. Vollmer, M. Haardt, and J. Gotze. Comparative study of joint-detection techniques for td-cdma based mobile radio systems. *IEEE Journal on Selected Areas in Communications*, 19(8):1461–1475, 2001.

[103] S.A. Vorobyov, A.B. Gershman, and Z.Q. Luo. Robust adaptive

beamforming using worst-case performance optimization: A solution to the signal mismatch problem. *IEEE Transactions on Signal Processing*, 51(2):313–324, 2003.

[104] X. Wang, W.S. Lu, and A. Antomiou. Constrained minimum-ber multiuser detection. *IEEE Transactions on Signal Processing*, 48(10):2903–2909, 2000.

[105] XM Wang, WS Lu, and A. Antoniou. A near-optimal multiuser detector for ds-cdma systems using semidefinite programming relaxation. *IEEE Transactions on Signal Processing*, 51(9):2446–2450, 2003.

[106] M. Weeks. *Digital signal processing using matlab and wavelets.* Infinity Science Pr Llc, 2007.

[107] C. Windpassinger and R.F.H. Fischer. Low-complexity near-maximum-likelihood detection and precoding for mimo systems using lattice reduction. In *Information Theory Workshop, 2003. Proceedings. 2003 IEEE*, pages 345–348. IEEE, 2003.

[108] S.P. Wu, S. Boyd, and L. Vandenberghe. Fir filter design via semidefinite programming and spectral factorization. In *Proceedings of the 35th IEEE Decision and Control, 1996.*, volume 1, pages 271–276. Ieee, 1996.

[109] D. Wubben, R. Bohnke, V. Kuhn, and K.D. Kammeyer. Near-maximum-likelihood detection of mimo systems using mmse-based lattice reduction. In *Communications, 2004 IEEE International Conference on*, volume 2, pages 798–802. IEEE, 2004.

[110] L. Xiao, M. Johansson, H. Hindi, S. Boyd, and A. Goldsmith. Joint optimization of communication rates and linear systems. *IEEE Transactions on Automatic Control*, 48(1):148–153, 2003.

[111] Z. Xie, R.T. Short, and C.K. Rushforth. A family of suboptimum detectors for coherent multiuser communications. *IEEE Journal on Selected Areas in Communications*, 8(4):683–690, 1990.

[112] A. Yener, R.D. Yates, and S. Ulukus. Cdma multiuser detec-

tion: A nonlinear programming approach. *IEEE Transactions on Communications*, 50(6):1016–1024, 2002.

[113] X. Zhang and D. Brady. Asymptotic multiuser efficiencies for decision-directed multiuser detectors. *IEEE Transactions on Information Theory*, 44(2):502–515, 1998.

[114] E. Zitzler. *Evolutionary algorithms for multiobjective optimization: Methods and applications.* Shaker, 1999.

Personal Information

Name:	Tharwat Elsayed Hamed Morsy
Born in:	El-Sharqia / Egypt
Marital status:	married, three children.

Curriculum Vitae

1992 – 1996	Study computational science. Suez canal university. Egypt
1999 – 2003	Master of science. Mathematics-Suez Canal university. Egypt.
2008 – 2012	Ph.D. student at the Information Processing Lab Dortmund University of Technology.