Linnæus University
Department of Physics and Electrical Engineering

Master’s Thesis in Electrical Engineering
with specialization in Signal Processing and Wave Propagation

Convex Optimization Methods
for System Identification

Author: Dino Dautbegovic
Supervisor: Sven Nordebo
Examiner: Sven Nordebo
Date: 2014.06.09
Course Code: 5ED06E
Subject: Signal Processing
Level: Advanced
Abstract

Faculty of Technology
Department of Physics and Electrical Engineering

Master’s Thesis in Electrical Engineering with specialization in Signal Processing and Wave Propagation (30 Credits)

Convex Optimization Methods for System Identification

by Dino Dautbegovic

The extensive use of a least-squares problem formulation in many fields is partly motivated by the existence of an analytic solution formula which makes the theory comprehensible and readily applicable, but also easily embedded in computer-aided design or analysis tools. While the mathematics behind convex optimization has been studied for about a century, several recent researches have stimulated a new interest in the topic. Convex optimization, being a special class of mathematical optimization problems, can be considered as generalization of both least-squares and linear programming. As in the case of a linear programming problem there is in general no simple analytical formula that can be used to find the solution of a convex optimization problem. There exists however efficient methods or software implementations for solving a large class of convex problems. The challenge and the state of the art in using convex optimization comes from the difficulty in recognizing and formulating the problem.

The main goal of this thesis is to investigate the potential advantages and benefits of convex optimization techniques in the field of system identification. The primary work focuses on parametric discrete-time system identification models in which we assume or choose a specific model structure and try to estimate the model parameters for best fit using experimental input-output (IO) data. By developing a working knowledge of convex optimization and treating the system identification problem as a convex optimization problem will allow us to reduce the uncertainties in the parameter estimation. This is achieved by reflecting prior knowledge about the system in terms of constraint functions in the least-squares formulation.
Acknowledgments

I would like to dedicate a moment in acknowledging all the persons that contributed to the development and progress of this research project by sharing their expertise in the field and providing constructive feedback throughout the work.

My deep gratitude goes to the research supervisor and Professor Sven Nordebo for his enthusiastic involvement, for the encouragements in following own ideas, and for the proper assistance in limiting the extent of the work. In addition, he continuously continued to provide guidance, suggestions, and advices during the whole course. I would also like to express my gratitude to the company Combitech (IS) for their collaboration and student interests, providing me the opportunity to take part in their research work at the (ROS) laboratory. Special thanks goes to the driving company project supervisors Kim Fransson and Oscar Lindhe for taking an active role in the research, helping me with the instrument setup and data collection. Their shared working experience forced me to think outside the ideal simulations by taking into account the practical difficulties that often emerge when dealing with real-time measurements.
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td></td>
<td>I</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td></td>
<td>III</td>
</tr>
<tr>
<td>1 Introduction</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1.1 Opening</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1.2 Problem Formulation and Objective</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1.3 Outline</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>2 Theoretical Background</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>2.1 Mathematical Optimization</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>2.1.1 Linear Programming</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>2.1.2 Least-Squares</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2.1.3 Convex Optimization</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>2.2 Discrete-Time Linear Time-Invariant (LTI) Systems</td>
<td></td>
<td>13</td>
</tr>
<tr>
<td>3 Scientific Research</td>
<td></td>
<td>17</td>
</tr>
<tr>
<td>3.1 Least-Squares System Identification</td>
<td></td>
<td>17</td>
</tr>
<tr>
<td>3.1.1 System Identification Model</td>
<td></td>
<td>18</td>
</tr>
<tr>
<td>3.2 Contribution</td>
<td></td>
<td>22</td>
</tr>
<tr>
<td>3.2.1 Inverse System Modeling</td>
<td></td>
<td>22</td>
</tr>
<tr>
<td>3.2.2 Extending the FIR Model</td>
<td></td>
<td>24</td>
</tr>
<tr>
<td>4 Result and Analysis</td>
<td></td>
<td>27</td>
</tr>
<tr>
<td>4.1 Simulated Channel</td>
<td></td>
<td>27</td>
</tr>
<tr>
<td>4.2 Constrained Optimization Problem</td>
<td></td>
<td>31</td>
</tr>
<tr>
<td>4.3 Conclusion</td>
<td></td>
<td>34</td>
</tr>
<tr>
<td>Appendices</td>
<td></td>
<td>36</td>
</tr>
<tr>
<td>Appendix A</td>
<td></td>
<td>36</td>
</tr>
<tr>
<td>A.1 Discrete Fourier Transform (DFT)</td>
<td></td>
<td>36</td>
</tr>
<tr>
<td>A.2 IIR-system model</td>
<td></td>
<td>37</td>
</tr>
<tr>
<td>References</td>
<td></td>
<td>40</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Opening

The extensive use of a least-squares problem formulation in system identification but also in other fields is partly motivated by the existence of an analytic solution formula which makes the theory comprehensible and readily applicable, but also easily embedded in computer-aided design or analysis tools. While the mathematics behind convex optimization has been studied for about a century, several recent researches have stimulated a new interest in the topic. Convex optimization, being a special class of mathematical optimization problems, can be considered as a generalization of both least-squares and linear programming. It is a well established fact that both least-squares and linear programming have a uniform theory and can be solved by means of numerical computations very efficiently. Due to the accessibility of effective algorithms, both methods are now days considered to be mature technologies more than ‘state of the art’ techniques. As in the case of a linear programming problem there is in general no simple analytical formula that can be used to find the solution of a convex optimization problem. There exists however efficient methods or software implementations for solving a large class of convex problems. One advance of convex optimization is the recognition that the interior-point methods, developed in the 1980s to solve linear programming problems, can be used to solve convex optimization problems as well [1]. Thus, from a conceptual point of view, using convex optimization is very much similar to using least-squares or linear programming. Once a problem can be recognized or formulated as a convex optimization problem, it can be solved reliably and efficiently with the available software. The challenge and the state of the art in using convex optimization arise from the difficulty in recognizing and formulating the problem. The full benefits of convex optimization comes only when the problem is a prior known to be convex.

1.2 Problem Formulation and Objective

The main goal of this thesis is to investigate the potential advantages and benefits of convex optimization techniques in the field of system identification. The primary work focuses on parametric discrete-time system identification models in which we assume or choose a specific model structure
and try to estimate the model parameters for best fit using experimental input-output (IO) data. The fact that convex optimization is a generalization of least-square problems and that there exists reliable and efficient software for solving a large class of convex problems has been the major motivation behind this research. In many situations the only objective may be to identify the system, called a plant, by means of a parametric model. However, when dealing with real-time measurements, there is usually noise or some other disturbance source present at the output of the plant. This introduces an uncertainty in the measurements and eventually corrupts the estimates of the model parameters. By developing a working knowledge of convex optimization and treating the system identification problem as a convex optimization problem will allow us to reduce the uncertainties in the parameter estimation. This is achieved by reflecting prior knowledge about the system in terms of constraint functions in the least-squares formulation.

1.3 Outline

With an attempt to achieve a consistent transition between different topics the remaining thesis material is structured into three main parts or sections: §2 Theoretical Background, §3 Scientific Research, and §4 Results and Analysis. The first part, Theoretical Background, serves as an introduction to the theory of mathematical optimization, mainly focusing on least-squares and convex optimization problems. A survey behind the time-domain description of linear time-invariant (LTI) systems is also provided. It is important to note that the word introduction means a brief touch on the topic as to cover the associated terminology and important definitions used throughout the work. The second part presents the Scientific Research or an application of the background theory in the field of system identification or system modeling. Here we show how a FIR-system parametric model can be used to identify an unknown system from experimental input-output (IO) data. In addition, a proposal is suggested on how this parametric model can be improved by treating the system identification problem as a convex optimization problem and reflecting prior knowledge about the system into appropriate constraint functions. In the last part, Result and Analysis, we analyze the system identification model by a simulation in Matlab (CVX). In particular, we illustrate how the effect of noise can be suppressed to some extent by introducing constraints in the estimation model.
2 THEORETICAL BACKGROUND

2 Theoretical Background

2.1 Mathematical Optimization

A (continuous) mathematical optimization problem, or simply optimization problem, can be formulated as

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m.
\end{align*}
\]

(2.1)

Here the function \( f : \mathbb{R}^n \to \mathbb{R} \) is the objective function, the vector \( x = (x_1, \ldots, x_n) \) is the optimization variable of the given problem, the functions \( f_i : \mathbb{R}^n \to \mathbb{R} \) are the inequality constraint functions, and the constants \( b_i \) are the limits or bounds of the constraint functions. Note that in the general case it is possible to also include equality constraints of the form \( h_i(x) = d_i, \quad i = 1, \ldots, p \), where \( h_i : \mathbb{R}^n \to \mathbb{R} \). For simplicity, we will for the moment exclude the use of such equality constraint functions. An optimization problem with no constraints (i.e. \( m = 0 \) and \( p = 0 \)) is said to be unconstrained. The set of points for which the objective and all constraint functions are defined,

\[
\mathcal{D} = \text{dom}(f) \cap \bigcap_{i=1}^{m} \text{dom}(f_i),
\]

is called the domain\(^1\) of the optimization problem (2.1). The constraint functions determine the set \( \mathcal{S} = \{ x \in \mathcal{D} \mid f_i(x) \leq b_i, \quad i = 1, \ldots, m \} \) of all permissible solutions, or feasible points. Moreover, the problem (2.1) is said to be feasible if there exists at least one point in the set \( \mathcal{S} \), otherwise it is infeasible. A vector \( x^* \in \mathcal{S} \) is optimal, or a solution to the problem (2.1), if it has the smallest objective value among all vectors that satisfy the constraints. Hence, for any \( z \in \mathcal{S} \), \( x^* \) satisfies the inequality \( f(x^*) \leq f(z) \).

The stated problem (2.1) is formulated as a minimization problem. In contrast, an optimization problem in which we want to maximize the objective function is referred to as a maximization problem. The problem of maximizing an objective function \( g(x) \) is equivalent to minimizing the function \( -g(x) \).

\(^1\)Following the convention adopted in [1], the notation \( f : \mathbb{R}^n \to \mathbb{R} \) means that the function \( f \) maps some \( n \)-vectors into the real numbers \( \mathbb{R} \) and does not necessarily imply that \( \text{dom}(f) = \mathbb{R}^n \). Thus, \( f : A \to B \) should be interpreted as a mapping from the set \( \text{dom}(f) \subseteq A \) to the set \( B \). This convention is similar to function declaration in computer languages.
tion $f(x) = -g(x)$. Other types of constraint conditions can be easily con-
structed. The inequality constraint $f_i(x) \geq b_i$ is equivalent to $-f_i(x) \leq -b_i$ and the equality constraint $f_i(x) = b_i$ implies that both $f_i(x) \leq b_i$ and $f_i(x) \geq b_i$. Thus, from a theoretical perspective, the equality constraints are in fact redundant. Despite this, for practical purposes it may be desired to treat them explicitly. It is interesting to make the observation that the constrained optimization problem (2.1) can be viewed as an unconstrained optimization problem in which the domain of the objective function is re-
stricted to the set $S$, that is, $f : S \rightarrow \mathbb{R}$. Thus, a minimization problem consists of finding a vector $x^*$ in the restricted domain of $f$ that has the smallest objective value.

Depending on the specific form of the objective and constraint functions, but also on the restrictions posed on the optimization variable, will generally lead to different classes of optimization problems. A solution method for a particular class of optimization problems relies on an algorithm that computes the optimal point to some given accuracy. Our ability to solve a general optimization problem varies considerably with the efficiency of these algorithms. In particular, the solution of the general nonlinear optimization problem (2.1) turns out to be a very difficult task involving some sort of compromise between the possibility of not finding the solution and a very long computation time. Generally, we make the distinction between two different approaches used to solve nonlinear optimization problems. In local optimization, one focuses on finding a solution that is only locally optimal with the compromise of rejecting the attempt of finding a solution that min-
imizes the objective function over the set of all feasible points. In contrast, global optimization techniques attempt to find a true global solution of the optimization problem with the cost of compromising efficiency in terms of the computation time. There are however some exception to the general rule that most optimization problems are difficult to solve. Two important examples for which effective and reliable algorithms exists are linear pro-
grams and least-squares problems described in §2.1.1 and §2.1.2. Another exception is convex optimization described in §2.1.3.
2.1.1 Linear Programming

The optimization problem (2.1) is called a linear programming problem (LP) if both the objective and constraint functions are linear. Hence, for all \( x, y \in \mathbb{R}^n \) and all \( \alpha, \beta \in \mathbb{R} \), they satisfy

\[
\begin{align*}
    f(\alpha x + \beta y) &= \alpha f(x) + \beta f(y) \\
    f_i(\alpha x + \beta y) &= \alpha f_i(x) + \beta f_i(y).
\end{align*}
\]  

(2.2)

Note that the term nonlinear optimization is used to describe an optimization problem where at least one of the functions \( f, f_1, \ldots, f_m \) is not linear, but at the same time not known to be convex (see §2.1.3). In general, any linear function \( f: \mathbb{R}^n \rightarrow \mathbb{R} \) can be expressed in the form \( f(x) = c_1x_1 + \cdots + c_nx_n \) where \( c_i \in \mathbb{R} \) are arbitrary constants. By introducing the column vector \( \mathbf{c} \in \mathbb{R}^n \), the given form of a linear function can be written more compactly as \( f(x) = \mathbf{c}^T\mathbf{x} \). Thus, in accordance with (2.1), a linear programming problem can be formulated as

\[
\begin{align*}
    \text{minimize} & \quad \mathbf{c}^T \mathbf{x} \\
    \text{subject to} & \quad \mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i = 1, \ldots, m.
\end{align*}
\]  

(2.3)

Here the vectors \( \mathbf{c}, \mathbf{a}_1, \ldots, \mathbf{a}_m \in \mathbb{R}^n \) and the scalars \( b_1, \ldots, b_m \in \mathbb{R} \) are parameters that specify the objective and constraint functions. It is a standard procedure to introduce the \( m \times n \) matrix \( \mathbf{A} = [a_{ij}] \) and the \( m \times 1 \) column vector \( \mathbf{b} = [b_i] \). In this case, the linear program (2.3) attains an alternative form given by

\[
\begin{align*}
    \text{minimize} & \quad \mathbf{c}^T \mathbf{x} \\
    \text{subject to} & \quad \mathbf{A} \mathbf{x} \leq \mathbf{b}.
\end{align*}
\]  

(2.4)

There is no simple analytic formula for the solution of a linear program. Instead, a solution approach depends on an algorithm that computes the optimal vector \( \mathbf{x}^* \) to some given accuracy within an acceptable time course. The classical tool for solving a linear programming problem in practice is the simplex algorithms proposed and developed by G. Dantzig [2] while the most recent approach are the interior-point methods described in the book Convex Optimization by S. Boyd and L. Vandenberghe [1].
2 THEORETICAL BACKGROUND

2.1.2 Least-Squares

Recall that a general linear system of \( m \) equations in the \( n \) unknowns \( x_1, x_2, \ldots, x_n \) can be written as

\[
\begin{align*}
    a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= y_1 \\
    a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= y_2 \\
    \vdots & \quad \vdots \\
    a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= y_m.
\end{align*}
\]

By introducing the matrix \( A \in \mathbb{R}^{m \times n} \), and the column vectors \( x \in \mathbb{R}^n \) and \( y \in \mathbb{R}^m \), where

\[
A = \begin{bmatrix}
    a_{11} & a_{12} & \cdots & a_{1n} \\
    a_{21} & a_{22} & \cdots & a_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    a_{m1} & a_{m2} & \cdots & a_{mn}
\end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \text{and} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix},
\]

the linear system (2.5) can be written more compactly using the standard matrix notation \( Ax = y \). In general, we say that a linear system is consistent if it has at least one solution and inconsistent if it has no solutions. Let us express the matrix \( A \in \mathbb{R}^{m \times n} \) in terms of its columns \( a_j \in \mathbb{R}^m \) as \( A = [a_1 \ a_2 \ \ldots \ a_n] \). The product \( Ax \) can then be viewed as an linear combination of the column vectors of \( A \) in which the coefficients are the entries of \( x \):

\[
Ax = x_1a_1 + x_2a_2 + \cdots + x_na_n.
\]

From this form we can conclude that a linear system \( Ax = y \) is consistent if and only if \( y \) can be expressed as a linear combination of the column vectors of \( A \). Hence, there exists real numbers \( x_1, x_2, \ldots, x_n \) such that \( x_1a_1 + x_2a_2 + \cdots + x_na_n = y \) for some \( y \in \mathbb{R}^m \). This statement is equivalent to the requirement that \( y \) is in the column space of \( A \).

Suppose that \( A \in \mathbb{R}^{m \times n} \) is a strictly skinny matrix, i.e. the number of equations \( m \) is strictly greater then the number of variables \( n \). Then the overdetermined linear system \( Ax = y \) is inconsistent for at least one vector \( y \in \mathbb{R}^m \). Consequently, unless \( y \) randomly happens to be in the range of \( A \), the system \( Ax = y \) cannot be solved. Since no exact solution exists, one approach is to look for an vector \( x \) that approximately solves \( Ax = y \). Let
us define the *residual* or error vector \( \mathbf{r} = \mathbf{Ax} - \mathbf{y} \) as an measurement unit for the error in the approximation. Then, a vector \( \mathbf{x} = \mathbf{x}_{ls} \) that minimizes \( \| \mathbf{r} \| \) with respect to the *Euclidean inner product* on \( \mathbb{R}^n \) is referred to as the *least-squares* solution. An important observation that allows us to derive an explicit formula for the least-squares solution is that minimization of the norm of the residual \( r = \| \mathbf{Ax} - \mathbf{y} \| \) also minimizes the norm squared \( r^2 = \| \mathbf{Ax} - \mathbf{y} \|^2 \). Hence, to find \( \mathbf{x} \), we minimize

\[
\| \mathbf{r} \|^2 = \mathbf{r}^T \mathbf{r} = \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} - 2 \mathbf{y}^T \mathbf{A} \mathbf{x} + \mathbf{y}^T \mathbf{y}.
\]

Calculating the gradient with respect to \( \mathbf{x} \) of the above quadratic form and setting the resulting expression equal to zero, we obtain

\[
\nabla_{\mathbf{x}} \| \mathbf{r} \|^2 = 2 \mathbf{A}^T \mathbf{A} \mathbf{x} - 2 \mathbf{A}^T \mathbf{y} = 0,
\]

which yields the *normal equation* or the *normal system*

\[
\mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{y}.
\]  \( (2.6) \)

For every inconsistent linear system \( \mathbf{Ax} = \mathbf{y} \), the associated normal system (2.6) is consistent with all the solutions being least-squares solutions of \( \mathbf{Ax} = \mathbf{y} \). In general, least-squares solutions are not unique. However, if \( \mathbf{A} \) is a \( m \times n \) matrix with \( n \) linearly independent column vectors, the square matrix \( \mathbf{A}^T \mathbf{A} \) will be invertible \cite{3}. Consequently, for every \( \mathbf{y} \in \mathbb{R}^m \), the linear system \( \mathbf{Ax} = \mathbf{y} \) has a unique least-squares (approximate) solution given by

\[
\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y} = \mathbf{A}^\dagger \mathbf{y}.
\]  \( (2.7) \)

The matrix \( \mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \in \mathbb{R}^{n \times m} \) is referred to as the *pseudo-inverse*, or the left inverse, of a *full rank*\(^2\) skinny matrix \( \mathbf{A} \). It is straightforward to show that the least-squares solution \( \mathbf{x} = \mathbf{A}^\dagger \mathbf{y} \) reduces to the exact solution \( \mathbf{x} = \mathbf{A}^{-1} \mathbf{y} \) whenever \( \mathbf{A} \) is a full rank square matrix. This consistency allows us to replace the strict inequality \( m > n \) with \( m \geq n \) without affecting the analytic formula as given by (2.7).

In connection with the optimization problem (2.1) we realize that the

\(^2\)The rank of a \( m \times n \) matrix \( \mathbf{A} \), denoted by \( \text{rank}(\mathbf{A}) \), is defined as the number of linearly independent rows or columns of \( \mathbf{A} \). It can be showed that the maximum possible rank of \( \mathbf{A} \) satisfies the relation \( \text{rank}(\mathbf{A}) \leq \min(m, n) \). A matrix that has a rank as large as possible is said to have full rank; otherwise, the matrix is rank deficient.
least-squares problem can be formulated as an optimization problem with no constraints. If we denote the objective function by \( f(x) = \|Ax - y\|^2 \), the least-squares minimization problem is simply given by

\[
\text{minimize} \quad f(x) = \|Ax - y\|^2 = \sum_{i=1}^{m} (a_i^T x - y_i)^2. \tag{2.8}
\]

Here \( A \in \mathbb{R}^{m \times n}, \ a_i^T \in \mathbb{R}^n \) are the row vectors of \( A \), and the vector \( x \in \mathbb{R}^n \) is the optimization variable. Since \( A \) is assumed to be full rank, the minimization problem (2.8) has a unique solution given by \( x^* = A^\dagger y \).

### 2.1.3 Convex Optimization

Before proceeding with the formal definition of a general convex optimization problem it is necessary to cover some basic convexity theory and the associated terminology. This includes the definition of feasible sets, global and local optimum points, convex sets, and convex functions.

Consider a general optimization problem written in the form

\[
\text{minimize} \quad f(x) \\
\text{subject to} \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m \\
\quad h_i(x) = d_i, \quad i = 1, \ldots, p. \tag{2.9}
\]

As before we have that \( f : \mathbb{R}^n \to \mathbb{R} \) is the objective function, the vector \( x \) is the optimization variable, the functions \( f_i : \mathbb{R}^n \to \mathbb{R} \) are the inequality constraints, and the constants \( b_i \) are the boundaries of the inequality constraint functions. The only difference in the formulation between the optimization problem (2.1) and (2.9) is that we now have included the equality constraint functions \( h_i : \mathbb{R}^n \to \mathbb{R} \) bounded by the constants \( d_i \). If we let \( f(x) = f_0(x) \), the domain \( D \) of the optimization problem (2.9) can be expressed as

\[
D = \bigcap_{i=0}^{m} \text{dom}(f_i) \cap \bigcap_{i=1}^{p} \text{dom}(h_i),
\]

and the set \( S \) of all feasible points is now given by

\[
S = \{ x \in D \mid f_i(x) \leq b_i, \ i = 1, \ldots, m, \ h_i(x) = d_i, \ i = 1, \ldots, p \}.
\]

Consequently, a vector \( x^* \in S \) is optimal if and only if, for all \( x \in S \), \( x^* \)
satisfies the inequality $f(x^*) \leq x$. Alternatively, by defining the optimal value

\[ p^* = \inf \{ f(x) \mid f_i(x) \leq b_i, \; i = 1, \ldots, m, \; h_i(x) = d_i, \; i = 1, \ldots, p \}, \]

where $p^*$ is allowed to take on the extended values $\pm \infty$, one can simply state that a vector $x^* \in S$ is optimal if $f(x^*) = p^*$. When there exists an optimal point for the problem (2.9), we say that the optimal value is attained, and the problem is solvable. Further, the $i$th inequality constraint $f_i(x) \leq b_i$ is said to be active at the feasible point $x$ if $f_i(x) = b_i$. Else if $f_i(x) < b_i$, the constraint $f_i(x) \leq b_i$ is inactive. A constraint that does not affect the feasible set when excluded is said to be redundant.

With the given definition of an optimal point we have implicitly taken into account the condition that any optimal point is also a globally optimal point. A feasible point $x^*$ is said to be locally optimal if there exists a $\epsilon$-neighborhood around the given point for which there is no other point $x$ with a better objective value, i.e., $f(x^*) \leq f(x)$ for all $x \in S$ such that $\|x^* - x\| < \epsilon, \; \epsilon > 0$. A point that is globally optimal is also locally optimal but the converse is generally not true. For most optimization problems it is desired to find a globally optimal point since it minimizes the objective function over the set of all feasible points. Despite this, many of the available solution methods are designed to find a point that is only locally optimal. Fortunately, a fundamental property of convex optimization problems is that any locally optimal point is also a globally optimal point.

A convex optimization problem is one of the form

\[
\begin{align*}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & x \in S,
\end{align*}
\]

where $f(x)$ is a convex function and $S \subseteq \mathbb{R}^n$ is a convex set. A set $S$ is convex if the line segment between any two point points in $S$ also lies in $S$, i.e., if for any two points $x, y \in S$ and any $\theta \in \mathbb{R}$ with $0 \leq \theta \leq 1$, we have that

\[ \theta x + (1 - \theta)y \in S. \]

\[ ^3 \text{If the problem is infeasible, then by following the standard convention that the infimum of an empty set is } \infty, \text{ we let } p^* = \infty. \text{ If there are feasible points } x_k \text{ such that } f(x_k) \to -\infty \text{ as } k \to \infty, \text{ then } p^* = -\infty \text{ and we say the problem is unbounded below.} \]
A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if the $\text{dom}(f)$ is a convex set and if for all $x, y \in \text{dom}(f)$, and $\theta$ with $0 \leq \theta \leq 1$, we have that

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y).$$

Geometrically, this inequality indicates that the line segment between any two points $(x, f(x))$ and $(y, f(y))$ lies above the graph of $f$, see Figure 1. A function is strictly convex if the above inequality holds strictly whenever $x \neq y$ and $0 < \theta < 1$. We say that a function $f$ is concave if $-f$ is convex and strictly concave if $-f$ is strictly convex. An optimization problem in which we instead want to maximize a concave function $f$ over a convex set $S$ is readily solved by minimizing the convex objective function $-f$. For this reason we refer to the concave maximization problem as being a convex optimization problem.

Usually we want to explicitly express the constraints that determine the convex set $S$. In this case, a convex optimization problem is formulated as

$$\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m, \\
& \quad c^T_i x = d_i, \quad i = 1, \ldots, p,
\end{align*}$$

(2.11)

where $f, f_1, \ldots, f_m$ are convex functions, i.e., for all $x, y \in \mathbb{R}^n$ and all $\alpha \geq 0$, $\beta \geq 0$ where $\alpha, \beta \in \mathbb{R}$ and $\alpha + \beta = 1$, they satisfy

$$\begin{align*}
f(\alpha x + \beta y) & \leq \alpha f(x) + \beta f(y) \\
f_i(\alpha x + \beta y) & \leq \alpha f_i(x) + \beta f_i(y).
\end{align*}$$

(2.12)
By comparing (2.12) with (2.2) we see that the inequality replaces the more
restrictive equality and that the inequality must hold for only certain values
of $\alpha$ and $\beta$. A linear or affine function is thus both convex and concave
and any linear, minimization or maximization, programming problem is ac-
cordingly also a convex optimization problem. Comparing (2.11) with the
general nonlinear form (2.9), the convex optimization problem has three
additional requirements: the objective function $f$ must be convex, the in-
equality constraint functions $f_1, \ldots, f_m$ must be convex, and the equality
constraint functions $h_i(x) = c_i^T x$ must be linear (affine). One can show that
the feasible set of the problem (2.11), given by

$$S = \{x \in \mathcal{D} \mid f_i(x) \leq b_i, \ i = 1, \ldots, m, \ c_i^T x = d_i, \ i = 1, \ldots, p\},$$

is in fact convex. Thus, in a convex optimization problem, we minimize a
convex objective function over a convex set described specifically by a set
of convex inequality constraints and linear equality constraints. It is not
difficult to show by applying the definition of a convex set and a convex
function that if $x^*$ is a locally optimal point of $f(x^*)$, then it must also be
a globally optimal point [4].

We have seen in §2.1.2 how a least-squares problem can be formulated as
an optimization (minimization) problem with no constraints. It turns out
that the quadratic objective function $f(x) = \|Ax - y\|$ is in fact a convex
function. In order to show this, we need to consider the definition of a norm
function. A function $f : \mathbb{R}^n \to \mathbb{R}$ with $\text{dom}(f) = \mathbb{R}^n$ is called a norm if and only if

- $f$ is nonnegative: $f(x) \geq 0$ for all $x \in \mathbb{R}^n$
- $f$ is definite: $f(x) = 0$ only if $x = 0$
- $f$ is homogenous: $f(\theta x) = \theta f(x)$ for all $\mathbb{R}^n$ and $\theta \in \mathbb{R}$
- $f$ satisfies the triangle inequality: $f(x + y) \leq f(x) + f(y)$ for all
  $x, y \in \mathbb{R}^n$.

The homogeneity property along with the triangle inequality can be com-

---

4A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is affine if it can be expressed as a sum of linear function and a constant, i.e., a function of the form $f(x) = Ax + b$, where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Any linear or constant function is by definition also affine.
bined into the single condition

\[ f(\alpha x + \beta y) \leq f(\alpha x) + f(\beta y) = \alpha f(x) + \beta f(y) \]

that in turn must hold for all \( \alpha, \beta \in \mathbb{R} \) and all \( x, y \in \mathbb{R}^n \). Since the \( \text{dom}(f) = \mathbb{R}^n \) is a convex set and we can always choose \( \alpha, \beta \geq 0 \) such that \( \alpha + \beta = 1 \), it follows that any norm function is also a convex function. Perhaps the most familiar norm is the Euclidean or \( l_2 \)-norm on \( \mathbb{R}^n \) defined as

\[ \|x\|_2 = (x^T x)^{\frac{1}{2}} = (x_1^2 + \cdots + x_n^2)^{\frac{1}{2}}. \]

The \( l_2 \)-norm is just a special case of the generalized \( l_p \)-norm given by

\[ \|x\|_p = (|x_1|^p + \cdots + |x_n|^p)^{\frac{1}{p}}, \quad p \geq 1. \]

Another important special case is the \( l_\infty \)-norm defined by the limit

\[ \lim_{p \to \infty} \|x\|_p = \max\{|x_1|, \ldots, |x_n|\}. \]

Thus, the least-squares problem (2.8) is an unconstrained convex optimization problem. More importantly, we can extend the definition by adding constraints. For our purposes we will be primarily concerned with the formulation

\[
\begin{align*}
\text{minimize} & \quad f(x) = \|Ax - y\|_2 \\
\text{subject to} & \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m \\
& \quad c_i^T x = d_i, \quad i = 1, \ldots, p
\end{align*}
\]  

(2.13)

where \( f_1, \ldots, f_m \) are convex functions.

There is in general no analytical formula for the solution of the convex optimization problem (2.11). There are however effective methods, or software implementations, designed for solving convex programs. One popular software implementation that we will utilize in order to solve the constrained least-squares problem (2.13) is CVX; a modeling system for constructing and solving disciplined convex programs (DCPs). CVX has the advantage of being implemented in Matlab, thereby allowing for the constraints and objectives to be specified using standard Matlab expression syntax. This combination makes it simple to perform the calculations needed to solve optimization problems and to further process the results obtained from their
solution. A complete survey of the CVX Research along with the CVX software download (Version 2.1, April 2014, Build 1079) can be found in [5].

2.2 Discrete-Time Linear Time-Invariant (LTI) Systems

The general relation for a single-input-single-output (SISO) discrete-time system can be formulated as

\[ y(n) = f[n, y(n-1), \ldots, y(n-N), x(n), \ldots, x(n-M)] \]  

where \( f[\cdot] \) denotes some function or a set of well-defined operations performed on the arguments. Here we assume that the exact internal structure of the system is either unknown or ignored and that the only way to interact with the system is by manipulating the associated input-output relation. A model for which we have no explicit access to the internal behavior of the system is referred to as a black box model. We usually say that the input signal \( x(n) \) is transformed by the system operator \( T \) into some output signal \( y(n) \) and use the notation \( y(n) \equiv T[x(n)] \) as to reflect this process. Another way to phrase this is that \( y(n) \) is the response of the system \( T \) to the excitation \( x(n) \). The given equation (2.14) represents actually a general \( N \)th-order difference equation in which the output sequence \( y(n) \) is recursively defined as a function of \( N \) past output values and the present and \( M \) past input values. Observe that in order to determine \( y(n) \) for \( n \geq n_0 \), we need the input signal \( x(n) \) for \( n \geq n_0 \) and the initial conditions \( y(n_0-1), \ldots, y(n_0-N) \). A system for which all initial conditions are zero is said to be initially relaxed\(^5\). Since the output at any instance of time \( n \) does not depend on future inputs nor future outputs, the system (2.14) is said to be causal.

When analyzing or designing discrete-time systems it is often desired to classify the systems according to the general properties that they satisfy. Moreover, the available techniques used to analyze discrete-time systems can only be applied when the system is beforehand assumed to satisfy certain properties. The standard classification of system properties includes concepts such as linearity, time invariance, causality, and stability. Here we consider an important subclass of the general discrete-time system (2.14), namely the class of linear time-invariant systems or simply LTI-systems. To

\(^5\)By convention we assume that every system is relaxed at \( n = -\infty \).
elaborate, a system $\mathcal{T}$ is said to be linear if and only if the relation
\[ \mathcal{T}[ax_1(n) + bx_2(n)] = a\mathcal{T}[x_1(n)] + b\mathcal{T}[x_2(n)] \]
holds for any arbitrary sequences $x_1(n)$ and $x_2(n)$ and any arbitrary constants $a$ and $b$. The given condition of linearity can be extended to any weighted linear combination of signals. The linearity property of the difference equation (2.14) can be reflected by expanding it as
\[ y(n) = -\sum_{k=1}^{N} a_k(n)y(n-k) + \sum_{k=0}^{M} b_k(n)x(n-k). \]  
(2.15)

The system $\mathcal{T}$ is said to be time-invariant if and only if the relation $y(n) = \mathcal{T}[x(n)]$ implies that
\[ \mathcal{T}[x(n-k)] = y(n-k) \]
for every input signal $x(n)$ and every time shift integer $k$. Clearly, the linear difference equation (2.15) is time-variant due to the time dependency of the parameters $a_k(n)$ and $b_k(n)$. However, if these parameters are assumed to be constant, then a system that is both linear and time-invariant can be described by the constant coefficient difference equation
\[ y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k), \]
or equivalently
\[ \sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k), \quad a_0 \equiv 1. \]  
(2.16)

The choice of the parameters $a_1, \ldots, a_N$ and $b_0, \ldots, b_M$ determines a specific LTI-system for which there exists a unique solution $y(n)$ for every distinct set of initial conditions. In general, the total response of the system can be decomposed as $y(n) = y_{zi}(n) + y_{zs}(n)$ where $y_{zi}(n)$ denotes the zero-input response obtained when the system is initially non relaxed and the input $x(n) = 0$ for all $n$, and $y_{zs}(n)$ denotes the zero-state response to the input signal $x(n)$ when the system is initially relaxed [6].

There are two basic time-domain descriptions of LTI-systems, wherein the input-output relationship described by the constant coefficient difference
equation (2.16) is one of them. The second method applies only when the LTI-system is assumed to be initially relaxed and relies on the fact that any arbitrary discrete-time signal \( x(n) \) can be resolved into a sum of weighted unit sample sequences, hence

\[
x(n) = \sum_{k=-\infty}^{\infty} x(k) \delta(n - k).
\]

(2.17)

Let us hypothetically denote the response \( y(n - k) \) of the relaxed LTI-system \( T \) to a unit sample input sequence \( \delta(n - k) \) by the function \( h(n - k) \). We can express this mathematically as

\[
y(n - k) \equiv h(n - k) = T[\delta(n - k)]
\]

for any integer \( k \in (-\infty, \infty) \). Then, the response \( y(n) \) to the input signal (2.17) can be expressed as

\[
y(n) = T[x(n)] = T \left[ \sum_{k=-\infty}^{\infty} x(k) \delta(n - k) \right]
\]

\[
= \sum_{k=-\infty}^{\infty} x(k) T[\delta(n - k)]
\]

\[
= \sum_{k=-\infty}^{\infty} x(k) h(n - k) = \sum_{k=-\infty}^{\infty} h(k) x(n - k).
\]

The formula

\[
y(n) = \sum_{k=-\infty}^{\infty} h(k) x(n - k)
\]

(2.18)
represents a convolution sum, usually denote by \( y(n) = h(n) \ast x(n) \). In result, any relaxed LTI-system is completely characterizied by the single function \( h(n) \) referred to as the impulse response. Note that in the derivation of (2.18) we used the linearity property in the second step and the time-invariance property in the third step. The requirement for the system to be initially relaxed means that there is no energy accumulation present in the system when the impulse \( \delta(n) \) is applied. In fact, the given approach implicitly assumes that the system is relaxed. If either one of these properties fails to persist, the validity of (2.18) can no longer be justified. The causality requirement that the output \( y(n) \) at any time \( n = n_0 \) should not depend
on future values of $x(n)$ can easily be related to the necessary and sufficient condition $h(n) = 0$ for $n < 0$ of the impulse response. In this case, the summation limits of the convolution formula (2.18) may be modified as to reflect this system property. Accordingly, we write

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n - k).$$

(2.19)

At this point it is convenient to subdivide the class of LTI-systems characterized by the impulse response $h(n)$ into two types. Since equation (2.19) represent a system whose impulse response is of infinite-duration, we classify the system as an infinite impulse response (IIR) system. When the impulse response is zero outside some finite interval, say $h(n) = 0$ for $n < 0$ and $n \geq L$, the convolution sum reduces to

$$y(n) = \sum_{k=0}^{L-1} h(k)x(n - k)$$

(2.20)

and we say that the system is a finite impulse response (FIR) system of order $L - 1$ or length $L$. The convolution sum (2.20) reveals a practical way of implementing a FIR-system. Such a realization involves additions, multiplications, and a finite memory allocation. Clearly, such a realization is not possible for the IIR-system since it would require an infinite number of memory locations. Fortunately, a practical way of implementing a family of IIR-systems exists for those that can be recursively described by means of the constant coefficient difference equation (2.16). Such a realization requires a finite amount of delays and will always involve a feedback loop. It can be shown that any FIR-system can also be realized recursively. For this reason we think of the terms FIR and IIR as general characteristics that distinguish two different types of LTI-systems, while the terms recursive and nonrecursive refer to the specific implementation structure. In summary, we have presented two time-domain descriptions of causal LTI-systems: the recursive description given by the constant coefficient difference equation (2.16) and the nonrecursive impulse response description given by the convolution sum (2.18).
3 Scientific Research

3.1 Least-Squares System Identification

In many practical applications one is faced with the problem of predicting the output signal of a physical system given a known input signal. When the underlying characteristics of the system are unknown, meaning that a precise mathematical description of the system is not available, it is necessary to excite the system with a predefined input, observe the output, and by some method utilize this input-output (IO) relation as to determine these characteristics. From a narrow perspective, system identification refers to the process of determining a model or improving the mathematical representation of a physical system from experimental data. The available system identification techniques can generally be grouped into frequency-domain identification methods and time-domain identification methods. Further, we make the distinction between parametric models, in which we choose or assume a specific model structure and try to estimate the model parameters for best fit, and nonparametric models, where a model is not specified beforehand but is instead determined successively from the observed data.

In a broader sense, system identification deals with the task of designing a suitable model, choosing the input signal as to excite the dynamics of interest, implementing and interpreting the results from such an experiment [7]. We will exclusively deal with discrete-time single-input-single-output (SISO) models in which we assume that the system to be identified falls under the category of a linear time-invariant (LTI) system or at least behaves approximately as a LTI-system when operated under normal conditions. With this assumption we have implicitly narrowed us to parametric models. The material that follows will illustrate in particular how a FIR-model can be used to identify (approximate) an unknown system using experimental IO-data. The choice of the given parametric model is partly motivated by the nonrecursive linear form of FIR-system which makes them bounded-input-bounded-output (BIBO) stable. It is worth mentioning that the constant coefficient difference equation (2.16) can just as well be used as an suitable parametric model in which case we end up with an auto regressive moving average (ARMA) model [8].
3.1.1 System Identification Model

Recall from §2.2 that any relaxed causal LTI-system is completely characterized by its impulse response $h(n)$, $0 < n < \infty$. Thus, once the impulse response has been determined, the response $y(n)$ to any arbitrary excitation signal $x(n)$ can be obtained by evaluating $y(n) = h(n) \star x(n)$. Especially, for a causal FIR-system of length $L$, the response is given by the convolution sum

$$y(n) = \sum_{k=0}^{L-1} h(k)x(n-k).$$

In connection with the linear constant coefficient difference equation (2.16) we can deduce a FIR-system by setting the coefficients $a_0 = 1$ and $a_k = 0$ for $1 \leq k \leq M$. Hence,

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) = b_0 x(n) + \cdots + b_M x(n-M).$$

(3.1)

The output $y(n)$ at any instance of time $n_0$ can be viewed as a weighted linear combination of the input signal samples $x(n_0), \ldots, x(n_0-(M))$ in which the weighting coefficients are determined by the system parameters $b_k$ for $0 \leq k \leq M$. Since the system output is basically a weighted moving average of the input signal, the system is sometimes referred to as a moving average (MA) model. Note that in this case, the impulse response is simply given by

$$h(k) = \begin{cases} b_k, & 0 \leq k \leq M \\ 0, & \text{otherwise}. \end{cases}$$

Thus, the task of determining the impulse response $h(n)$ of a FIR-system is equivalent to finding the parameters $\{b_k\}$ that determine the MA-model. In practice it is not physically possible to determine the impulse response as purposed by the definition $h(n) = \mathcal{T}[\delta(n)]$. The best we can achieve with a predefined parametric model chosen to represent the unknown system is to estimate the parameters for best fit using available IO-data.

Let us assume that we have an unknown system, called a plant, characterized by the impulse response $h(n)$. We wish to identify (approximate) the plant by a FIR-system model of $M$ delays or $M+1$ adjustable coefficients. Accordingly, we excite the system with a known input signal $x(n)$ for $n = 0, \ldots, l$, and observe the output signal $y(n)$ of the plant. When
dealing with real-time measurements there will usually be noise present at the output signal that eventually corrupts the estimate of the system parameters. The presence of noise can be represented by including an additive noise source $w(n)$ at the output of the plant (see Figure 2). If we let $\hat{y}(n)$ denote the modeled or predicted output signal, then

$$\hat{y}(n) = \sum_{k=0}^{M} h(k) x(n - k).$$

Thus, in our simple identification model the noise is seen as being a part of the unknown system and is therefore being modeled as well. Unless the unknown system happens to be a FIR-system of order $M$ and $w(n) = 0$, there will always be a model prediction error associated with (3.2). Accordingly, we may form the error sequence $e(n)$ as the difference between the observed output signal $y(n)$ and the predicted output signal $\hat{y}(n)$. That is,

$$e(n) = y(n) - \hat{y}(n), \quad n = 0, \ldots, l.$$  

It is convenient to define the parameter vector $h \in \mathbb{R}^{M+1}$ and the input vector $x(n) \in \mathbb{R}^{M+1}$ as

$$h = \begin{bmatrix} h(0) \\ h(1) \\ \vdots \\ h(M) \end{bmatrix}, \quad x(n) = \begin{bmatrix} x(n) \\ x(n-1) \\ \vdots \\ x(n-M) \end{bmatrix}.$$
In this case, the system (3.2) can be expressed as
\[ \hat{y}(n) = h^T x(n), \quad n = 0, \ldots, l. \] (3.4)

Since we are usually dealing with causal signals, i.e., signals that are zero for \( n < 0 \), we can shift the above expression to the right by \( M \) samples. Hence,
\[ \hat{y}(n + M) = h^T x(n + M), \quad n = 0, \ldots, l - M. \] (3.5)

An alternative way to reflect this is to let \( n = M, \ldots, l \) in (3.2). Similarly, we define the error vector \( e \in \mathbb{R}^{l-M+1} \) as
\[ e = (y(M) - \hat{y}(M), \ldots, y(l) - \hat{y}(l)). \] (3.6)

Evaluating (3.5) for each value of \( n \) yields the matrix equation
\[
\begin{bmatrix}
\hat{y}(M) \\
\hat{y}(M + 1) \\
\vdots \\
\hat{y}(l)
\end{bmatrix} =
\begin{bmatrix}
x(M) & x(M-1) & \cdots & x(0) \\
x(M+1) & x(M) & \cdots & x(1) \\
\vdots & \vdots & \ddots & \vdots \\
x(l) & x(l-1) & \cdots & x(l-M)
\end{bmatrix}
\begin{bmatrix}
h(0) \\
h(1) \\
\vdots \\
h(M)
\end{bmatrix}.
\] (3.7)

With the vector notation \( \hat{y} \in \mathbb{R}^{l-M+1} \) and \( X \in \mathbb{R}^{(l-M+1) \times (M+1)} \), the system (3.7) can be expressed as
\[ \hat{y} = Xh. \] (3.8)

Note that the matrix \( X \) is formed in such a way that the product \( Xh \) is the convolution of \( h(n) \) and \( x(n + M) \). Consequently, we refer to the matrix \( X \) as the convolution matrix\(^6\). The whole idea behind the formulation of (3.7) relies on the assumption that the number of available measurement points exceeds the order of the system model. More specifically, we assume that \( l > 2M \) so that the linear system \( \hat{y} = Xh \) is overdetermined. In least-squares system identification we select the parameter vector \( h \) that minimizes the norm of the error vector \( e \), where
\[ \|e\| = \|y - Xh\| = \left[ \sum_{k=M}^{l} (y(k) - \hat{y}(k))^2 \right]^{\frac{1}{2}}. \]

\(^6\)The characteristic structure of \( X \in \mathbb{R}^{(l-M+1) \times (M+1)} \) resembles the shape of a non-symmetric Toeplitz matrix defined by the first row and the first column vector of \( X \).
With reference to equation (2.7) derived in §2.1.2, the least-squares approximate solution of (3.8) is given by

\[ h = (X^T X)^{-1} X^T \hat{y} = X^\dagger \hat{y}. \]  

(3.9)

Once the parameter vector has been estimated for a given set of IO-data, the predicted output \( \hat{y} \) can be directly simulated by a back substitution of \( h \) into (3.8).

When designing a system model it is instructive to compare the relative prediction error, defined as \( e_r = \|e\|/\|y\| \), of the approximation for different values of \( M \). If there were no noise present at the output of the plant, then obviously a larger value of \( M \) will lead to a smaller prediction error on the data used to form the model. Unfortunately, the presence of noise in real-time measurements is inevitable in which case we need to reflect over the choice of a larger model order as to avoid the possibility of modeling the noise to much. In addition, for large values of \( M \), the predictive ability of the model becomes worse for other IO-data not used to identify the system. Degeneration of the predictive ability when \( M \) is chosen to large is called overmodeling. As a result, we end up with a trade-off between the model order and the models predictive ability (robustness). In this context we will persistently refer to the IO-data used to identify the system as the modeling-data and the IO-data used to test the model as the validation-data. Earlier we mentioned that noise introduces uncertainties in the measurements and eventually corrupts the estimate of the system parameter vector \( h \). For illustration purposes, suppose that the predicted output can be expressed as

\[ \hat{y} = Xh + w \]

where \( w \in \mathbb{R}^{l-M+1} \) represents a unknown noise source or some other small measurement error. Here we assume that \( h = X^\dagger \hat{y} \) gives no estimation error (\( \|e\| = 0 \)) whenever \( w = 0 \). Another way to phrase this is to say that \( h \) yields a perfect prediction of \( y \) when \( w = 0 \). The least-squares estimate of \( h \) is then given by

\[ \hat{h} = X^\dagger (Xh + w) = h + X^\dagger w. \]

Clearly, the term \( X^\dagger w \) reflects a disturbance in the estimation of \( \hat{h} \) caused by the source \( w \). We note that the error \( X^\dagger w \) is completely independent of
3.2 Contribution

3.2.1 Inverse System Modeling

In some situations the primary purpose of designing a suitable model for the unknown system is not to predict the response but rather to determine a corrective system that compensates for the unwanted effects caused by the channel in which the transmitted signal propagates. In the general context of LTI-system theory the corrective system is referred to as the inverse system while from a communication theory perspective one usually prefers the abbreviation channel equalizer. To elaborate, consider the system function

\[ H(z) = \sum_{n=0}^{M} h(n)z^{-n} = z^{-M}(z - z_1) \cdots (z - z_M) \]  

(3.10)

associated with the estimated sequence \( h(n) \), \( 0 \leq n \leq M \). Here, the region of convergence (ROC) is determined by the set \( \{ z \in \mathbb{C} | z \neq 0 \} \). Clearly, \( H(z) \) represents a system that is both causal and stable. The corresponding inverse system is simply obtained by solving \( H(z)G(z) = 1 \) for \( G(z) \). Hence,

\[ G(z) = H^{-1}(z) = z^M \frac{1}{(z - z_1) \cdots (z - z_M)}. \]

(3.11)

If we now let \( g(n) = \mathcal{Z}^{-1}\{G(z)\} \), the reconstructed input signal \( \hat{x}(n) \) can be expressed as

\[ \hat{x}(n) = \sum_{k=0}^{n} g(k)\hat{y}(n-k), \quad n = M, \ldots, l. \]

(3.12)

\footnote{It can be shown that the smallest left inverse of a full rank skinny matrix \( A \in \mathbb{R}^{m \times n} \) is the matrix \( A^\dagger = (A^T A)^{-1} A^T \in \mathbb{R}^{n \times m} \). The word smallest is interpreted in the sense that for any other matrix \( B \), such that \( BA = I \), we have that \( \sum_{i,j} b_{ij}^2 \geq \sum_{i,j} a_{ij}^1 \).}
The complete channel equalization process is depicted in Figure 3 using a
time-domain description. Observe that in order to reconstruct the input
signal as suggested by (3.12) we have to assume that the system $T$ is invert-
ible, i.e., that there exists a one-to-one correspondence between its input and
output signals. Another way to phrase this is to say that $g(n) = Z^{-1}\{G(z)\}$
is not unique unless the region of convergence for the inverse system $G(z)$
is also specified. If we require for the inverse system to be causal, the
ROC of $G(z)$ must be chosen as $|z| > \text{max}(|z_1|, \ldots, |z_M|)$. Unless this ROC
contains the unit circle, which only occurs when $\text{max}(|z_1|, \ldots, |z_M|) < 1$,
the inverse system function will be unstable. Whenever all of the zeros
are contained inside the unit circle, $H(z)$ is said to be a minimum phase
system. Only then can we determine a stable, causal inverse system $G(z)$
that compensates for the effects caused by $H(z)$. Thus, the minimum phase
property of $H(z)$ ensures the stability of the inverse system [6]. The issue
of estimating or tracking the input sequence from a known non-minimum
phase FIR-system having many zeros located near or on the unit circle is
a challenging issue that cannot be solved by conventional techniques [9].

From a theoretical point of view, the non-minimum phase system $H(z)$
posses a stable non-causal inverse counterpart when the ROC is selected as
$|z| < \text{min}(|z_1|, \ldots, |z_M|)$. Consequently, methods used to track the input of
a non-minimum phase system rely either on a off-line implementation of the
noncausal inverse system or on a stable approximation of the causal inverse
system.

A simple approach that can be used to directly approximate the inverse
sequence $g(n)$ from experimental IO-data is illustrated in Figure 4. If we sup-
pose that the modeled or reconstructed input signal $\hat{x}(n)$ can be expressed
as $\hat{x}(n) = g(n) \ast y(n)$, $n = M, \ldots, l$, the procedure of estimating $g(n)$ is
completely analogous to the method used to determine $h(n)$ (compare with
Eq. 3.2). Hence, if we express the input-output relation as $\hat{x} = Yg$ and
define the error vector

$$
e = (x(M) - \hat{x}(M), \ldots, x(l) - \hat{x}(l)),$$

(3.13)

then a vector $g$ that minimizes $e$ with respect to the norm $\|e\| = \|x - Yg\|$ is
given by the least-squares estimate

$$g = (Y^T Y)^{-1} Y^T \hat{x} = Y^\dagger \hat{x}.$$

(3.14)
3 SCIENTIFIC RESEARCH

Figure 4: FIR-system modeling of the inverse $g(n)$.

The main advantage in estimating $g(n)$ directly is that the method bypasses the need of inverting the system and thereby avoids the stability issue that emerge when dealing with a non-minimum phase system. It is important to emphasize that the method is much more sensitive to noise and has the disadvantage of loss in flexibility when identifying high-order systems.

3.2.2 Extending the FIR Model

In §3.1.1 we presented a practical way of identifying or modeling a unknown system by means of the FIR parametric model. The analysis so far has treated the unknown system as an black box, meaning that the internal behavior of the system has been entirely ignored. Consequently, any outside unaccounted effect or disturbance that manifests itself at the output signal is seen as being a part of the unknown system. From a system identification point of view the disturbance will be embedded into the model, giving a distorted image of the actual channel. The following material illustrates a method that can be used to improve the estimation of the FIR-system model parameters provided that we have some prior knowledge about the system behavior. This is achieved by treating the system identification problem as an convex optimization problem and restricting the set of permissible solutions or estimations by appropriate constraint functions.

An important result derived in §2.1.3 shows that any norm function satisfies the convexity property. Consequently, a system identification problem in which we want to determine a parameter vector $h$ that minimizes the $l_2$-norm of the error vector $e = y - Xh$ can be considered as an unconstrained convex optimization problem. More importantly, we can extend
this formulation by including constraint functions. In accordance with the result (2.13), the extended FIR-system model can be expressed as

\[
\begin{align*}
\text{minimize} & \quad e(h) = \|y - Xh\|_2 \\
\text{subject to} & \quad e_i(h) \leq b_i, \quad i = 1, \ldots, m \\
& \quad c_i^T h = d_i, \quad i = 1, \ldots, p.
\end{align*}
\]

Here \(e : \mathbb{R}^{M+1} \rightarrow \mathbb{R}\) is the objective function, \(e_i : \mathbb{R}^{M+1} \rightarrow \mathbb{R}\) are the convex inequality constraint functions bounded by the constant \(b_i\) and \(g_i : \mathbb{R}^{M+1} \rightarrow \mathbb{R}\), where \(g_i(h) = c_i^T h\), are the linear equality constraint functions bounded by the constants \(d_i\).

The task of determining appropriate constraint functions is in general not trivial. The principle however is to restrict the domain in which we are searching for the parameter vector \(h\) that minimizes \(e(h)\). Even though the characteristic of the system are not entirely known, we can still construct a model based on both a certain insight to the system behavior and experimental IO-data. In the simplest case one can pose upper and lower boundaries for the model parameters by the constraint \(l_i \leq h_i \leq u_i\), \(i = 0, \ldots, M\), and allow the algorithm find a solution that satisfy the given constraint. We can even imagine the case that some of the parameters \(h_i\) are already known. In this case we only need to estimate the remaining unknown free parameters from experimental data. For illustration purposes we consider an example of how inequality constraints functions can be used to influence the frequency behavior of the magnitude function \(|H(k)|\), \(k = 0, \ldots, M\).

Recall that the system function \(H(z)\) associated with causal sequence \(h(n)\) of length \(L = M + 1\) is given by

\[
H(z) = \sum_{n=0}^{L-1} h(n)z^{-n}.
\]

Since the ROC of \(H(z)\) includes the unit circle, the corresponding \(N\)-point discrete Fourier transform (DFT) can be obtained by making the substitution \(z = e^{i2\pi k/N}, k = 0, \ldots, N - 1\). Hence,

\[
H(k) = \sum_{n=0}^{N-1} h(n)e^{-i2\pi kn}, \quad k = 0, \ldots, N - 1.
\]

Here we assume that \(L \leq N\) so that \(h(n)\) can be completely recovered from
$H(k)$ (see Appendix A.1). Observe that $H(k)$ is a linear function of $h(n)$ and that $|H(k)|$ is real valued. Consequently, we can define the requirement $|H(k)| \leq \varepsilon(k)$ where $\varepsilon(k)$ is some desired tolerance curve such that $\varepsilon(k) > 0$, $k = 0, \ldots, N - 1$. This is equivalent to the convex constraint

$$\max_k \frac{1}{\varepsilon(k)} |H(k)| \leq 1$$

$$\Leftrightarrow \left\| \frac{1}{\varepsilon(k)} H(k) \right\|_\infty \leq 1.$$

The constrained optimization problem can now be expressed as

$$\begin{align*}
\text{minimize} & \quad \| y - Xh \|_2 \\
\text{subject to} & \quad \left\| \frac{1}{\varepsilon(k)} H(k) \right\|_\infty \leq 1, \quad k = 0, \ldots, N - 1.
\end{align*}$$

(3.17)
4 Result and Analysis

4.1 Simulated Channel

As a particular example we study the system described by

\[ H(z) = \frac{B(z)}{A(z)} = K \frac{z^2 - 0.9z + 0.8}{z^2 - 0.9z} \]  \hspace{1cm} (4.1)

Accordingly, in order to identify \( h(n) \) using the FIR-model (3.2), we excite the unknown system with a predefined input signal \( x(n) \) and observe the output signal \( y(n) \). The input-output modeling data used in this example is shown in Figure 5. It is often instructive to plot the relative prediction error \( e_r = \|e\|/\|y\| \) as a function of the of the modeling order \( M \) or the number of adjustable coefficients \( M + 1 \). This is illustrated in Figure 6 for different values of the signal-to-noise (SNR) ratio. Due to the characteristic shape of these curves, we will refer to them as \( L \)-curves. Obviously when \( w(n) = 0 \), the relative prediction error can be reduced to zero with a model order of 50, meaning that a perfect prediction of the output signal is attained. It is however not immediately apparent how to choose an appropriate model order when there is noise present at the output of the plant. Simply increasing the model order further (overmodeling) as to obtain a smaller prediction error leads to a corrupted spectrum estimation (see Figure 7). Also, the predictive ability will at some point start to degrade for other IO-data not used to form the model. This fact is indicated by Figure 8 where we test the model on the validation data shown in Figure 5. As expected, the relative prediction error of the validation data starts to increase after a certain point, in this case when the number of coefficients is greater than 30. This point may often serve as an aid on how to choose an appropriate model order. Finally, shown in Figure 9 is the reconstructed input modeling signal and input validation signal. Observe that the reconstructed input validation signal has been determined without any prior knowledge of the actual input signal, but rather using the previously estimated model and the observed output signal. A straight implementation of the expression (3.12) will correctly track the input signal since the system \( H(z) \) is a minimum-phase system. It has been observed by gradually reducing the SNR that the zeros of \( H(z) \) will eventually shift outside the unit circle. Using a model order of \( M = 29 \), this tends to occur around a SNR = 2 (see Figure 10).
4 RESULT AND ANALYSIS

Figure 5: Input (blue) and output (black) experimental data used for modeling and validation of the system (4.1).

Figure 6: Illustrating the relative prediction error $\epsilon_r$ as a function of the model order or the number of adjustable coefficients. Shown in black is the $L$-curve when there is no noise source present. Shown in blue, red, and dashed red are the corresponding $L$-curves when the SNR equals a factor of 20, 10, and 5 respectively.
Figure 7: Illustrating the magnitude spectrum $20\log|H(\Omega)|$ and the phase spectrum $\angle H(\Omega)$ of the estimated function $H(k)$ when the SNR = 10. The actual spectrum is shown in black while the red and purple curves correspond to the estimations when using 30 and 50 adjustable coefficients respectively.

Figure 8: Illustration of the $L$-curve for modeling data (red) and validation data (dashed black) for a SNR of 10 at the modeling output signal.
4 RESULT AND ANALYSIS

Figure 9: Reconstruction of the input modeling signal and the input validation signal using a model of 30 adjustable coefficients. The SNR has been set to a value of 10 at the output of the plant.

Figure 10: Pole-zero plot of the estimated FIR-system function $H(z)$ of order $M = 29$. As indicated by the left plot, when there is no noise present all of the zeros are located symmetrically inside the unit circle. The right plot shows how the zeros are affected when the SNR = 2.
4.2 Constrained Optimization Problem

Next, we present an example on how the estimation of $h(n)$ can be improved by implementing the constrained optimization problem (3.17). The first illustration is recapitulated in Figure (11) where we identify the unknown system, having a SNR of 10 at the output, by a FIR-model of 50 adjustable coefficients. Here, the tolerance curve $\varepsilon(k)$ has been constructed as to resemble the shape of the actual magnitude spectrum. By posing the given constraint we can reject the solutions that cause the magnitude curve to exceed this upper limit and thereby achieve an overall improvement in the estimation. The curious reader may wonder how this curve is chosen in practice when we have no prior knowledge of the underlying system. One approach that can be used is to repeat the identification process using several independent measurements and averaging the estimated parameters. Once $h_{\text{avg}}(k)$ has been determined we can define the tolerance curve as $\varepsilon(k) = |H_{\text{avg}}(k)|$. Another illustration is provided in Figure 12 were we essentially perform the same analysis for a SNR = 2. As a result of the corrupted estimation, the magnitude spectrum contains a downward spike. This strange behavior can be explained by considering the associated pole-zero plot (see Figure 13) where we notice that two zeros have aligned at the corresponding frequency. Since in addition two of the zeros are located outside the unit circle, the system function $H(z)$ is no longer a minimum phase system. It is interesting to observe the effect of the posed constraint by studying the related pole-zero plot. Namely, a majority of the zeros have now retained their original symmetry. This has the implication of reducing fluctuations in both the magnitude and phase spectrum while at the same time restores the minimum phase property of $H(z)$. 
Figure 11: Illustrating the magnitude spectrum $20\log|H(k)|$ and the phase spectrum $\angle H(k)$ for the estimated function $H(k)$ of order $M = 49$ when the SNR = 10. The first graph from the top shows the result of the unconstrained estimation (red) while the second graph illustrates the corresponding estimation (blue) obtained by constraining the problem with a tolerance curve $\varepsilon(k)$ (dashed black).
4 RESULT AND ANALYSIS

Figure 12: Illustrating the magnitude spectrum $20\log|H(k)|$ for the estimated function $H(k)$ of order $M = 29$ when the SNR = 2.

Figure 13: Pole-zero plot of the estimated FIR-system function $H(z)$ of order $M = 29$. As indicated by the right plot, the constraint function has the effect of resetting the symmetry of the zeros and forcing them back inside the unit circle.
4.3 Conclusion

In this thesis we address the potential advantages that can be achieved by treating the parametric FIR-system identification model (3.2) as a convex optimization problem (3.15). This extension allows us to integrate prior knowledge about the system parameters in terms of convex constraint functions and thereby restrict the domain of possible estimations. Phrased in the language of convex optimization, this means restricting the set $S$ of all feasible points in which a solution to the minimization problem $e(h) = \|y - Xh\|_2$ is contained. Consequently, some of the false estimations that are believed to occur due to external disturbances can be automatically rejected or ignored in the estimation process. The task of determining appropriate constraint functions generally requires a certain prior insight of the unknown system. This makes the method highly application dependent. Even when this information is available it is not guaranteed that one can formulate a system property in terms of convex inequality constraint or linear equality constraint functions. For a more extensive survey of the current progress and limitations of convex optimization techniques in the field of system identification see [10] and [11].

As a particular example we illustrate how the effect of an additive noise source present at the output of the plant can be suppressed by implementing the constrained optimization problem (3.17). This system identification model was actually tested on real-time measurements where the unknown channel consisted of a transmitting and a receiving antenna separated a small distance apart. As the purpose of this measurement was to test how a non-modulated signal radiates, the measured output signal quality was extremely bad; the main limiting factor being a signal-to-noise ratio below 1. For this reason, the results of this experimental setup have been left out.
Appendix A

A.1 Discrete Fourier Transform (DFT)

Let $x(n)$ be a finite duration sequence of length $N$, i.e., $x(n) = 0$ for $n < 0$ and $n \geq N$. Then the $N$-point *discrete Fourier transform* (DFT) of $x(n)$ is defined as

$$X(k) \equiv \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}, \quad k = 0, \ldots, N - 1. \quad (A.1)$$

Let us adopt the notation $W_N = e^{-j\frac{2\pi}{N}}$. By evaluating (A.1) for each value of $k$ we obtain the matrix equation

$$\begin{bmatrix}
X(0) \\
X(1) \\
\vdots \\
X(N-1)
\end{bmatrix} =
\begin{bmatrix}
1 & 1 & \cdots & 1 \\
1 & W_N & \cdots & W_N^{N-1} \\
\vdots & \vdots & \ddots & \vdots \\
1 & W_N^{N-1} & \cdots & W_N^{(N-1)(N-1)}
\end{bmatrix}
\begin{bmatrix}
x(0) \\
x(1) \\
\vdots \\
x(N-1)
\end{bmatrix}. \quad (A.2)$$

It is convenient to define the vectors $X \in \mathbb{R}^N$, $x \in \mathbb{R}^N$, and the transformation matrix $W_N \in \mathbb{R}^{N \times N}$. In this case, the matrix form representation of the $N$-point DFT may be simply expressed as

$$X = W_N x. \quad (A.3)$$

An important property of the DFT is that the matrix $W_N$ of the linear transformation (A.3) is in fact symmetric. If we let $W_N^\star$ denote the complex conjugate of the matrix $W_N$, the *inverse discrete Fourier transform* (IDFT) can be expressed in matrix form as

$$x = \frac{1}{N} W_N^\star X \quad (A.4)$$

or equivalently using the notation

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi}{N}nk}, \quad n = 0, \ldots, N - 1. \quad (A.5)$$

Now let us assume that $x(n)$ for $n < 0$ and $n \geq L$, where $L < N$. In this case we can still compute the DFT using the matrix form (A.3) by zero padding the sequence $x(n)$ with $N - L$ zeros in the interval $L \leq n \leq N - 1$. 
A.2 IIR-system model

Consider the linear constant coefficient difference equation given by

\[ y(n) = - \sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k). \]  \hspace{1cm} (A.6)

We will show how (A.6) can be used as an parametric model for system identification. In contrast to the moving average (MA) model (see Eq. 3.1), obtained by setting the coefficients \( a_1 = \ldots = a_N = 0 \), the output signal \( y(n) \) now also depends on the \( N \) previous values of the output signal. For this reason, the IIR-system model described by the constant coefficient difference equation (A.6) is sometimes referred to as an auto regressive moving average (ARMA) model.

Assume that the output signal \( y(n), n = 0, \ldots, l \), from an unknown LTI-system can be expressed as

\[ y(n) = - \sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) + w(n) \]  \hspace{1cm} (A.7)

where \( w(n) \) represent additive white Gaussian noise (AWGN) with zero mean. It is convenient to introduce the column vectors

\[
p = [a_1 \ldots a_N \ b_0 \ldots b_M]^T
\]

and

\[
q(n) = [-y(n-1) \ldots -y(n-N) \ x(n) \ldots x(n-M)]^T
\]

so that (A.7) can be written as

\[ y(n) = p^T q(n) + w(n), \quad n = 0, \ldots, l. \]  \hspace{1cm} (A.8)

The objective here is to estimate the parameter vector \( p \) using available input-output (IO) data. In the simplest treatment, the predicted output signal \( \hat{y}(n) \) is obtained by setting \( w(n) = 0 \), i.e.,

\[ \hat{y}(n) = p^T q(n), \quad n = 0, \ldots, l. \]  \hspace{1cm} (A.9)
Evaluating (A.9) for each value of \( n \) yields the matrix equation

\[
\begin{bmatrix}
\hat{y}(0) \\
\hat{y}(1) \\
\vdots \\
\hat{y}(l)
\end{bmatrix} =
\begin{bmatrix}
-y(-1) & \cdots & -y(-N) & x(0) & \cdots & x(-M) \\
-y(0) & \cdots & -y(1-N) & x(1) & \cdots & x(1-M) \\
\vdots \\
\vdots \\
-y(l-1) & \cdots & -y(l-N) & x(l) & \cdots & x(l-M)
\end{bmatrix}
\begin{bmatrix}
a_1 \\
\vdots \\
a_N \\
b_0 \\
\vdots \\
b_M
\end{bmatrix}.
\]

With the vector notation \( \hat{y} \in \mathbb{R}^{l+1} \) and \( Q \in \mathbb{R}^{(l+1)(M+N+1)} \), the above linear system can be expressed as

\[
\hat{y} = Qp. \tag{A.10}
\]

In least-squares estimation we choose the parameter vector \( p \) as to minimize the Euclidean norm of the error vector \( e \), where

\[
\|e\| = \|y - Qp\| = \left[ \sum_{k=0}^{l} (y(k) - \hat{y}(k))^2 \right]^{\frac{1}{2}}.
\]

The least-squares (approximate) solution of (A.10) is then given by

\[
p = (Q^TQ)^{-1}Q^T\hat{y} = Q^T\hat{y}. \tag{A.11}
\]
References


