Convex Optimization Methods for Model Reduction

by

Kin Cheong Sou

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2008

Author . . . . . .
Department of Electrical Engineering and Computer Science
August 29, 2008

Certified by . . . . . .
Luca Daniel
Associate Professor
Thesis Supervisor

Certified by . . . . . .
Alexandre Megretski
Professor
Thesis Supervisor

Accepted by . . . . . .
Terry P. Orlando
Chairman, Department Committee on Graduate Students

Convex Optimization Methods for Model Reduction

by

Kin Cheong Sou

Submitted to the Department of Electrical Engineering and Computer Science
on August 29, 2008, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Computer Science and Engineering

Abstract

Model reduction and convex optimization are prevalent in science and engineering applications. In this thesis, convex optimization solution techniques to three different model reduction problems are studied.

Parameterized reduced order modeling is important for rapid design and optimization of systems containing parameter dependent reducible sub-circuits such as interconnects and RF inductors. The first part of the thesis presents a quasi-convex optimization approach to solve the parameterized model order reduction problem for linear time-invariant systems. Formulation of the model reduction problem as a quasi-convex program allows the flexibility to enforce constraints such as stability and passivity in both non-parameterized and parameterized cases. Numerical results including the parameterized reduced modeling of a large RF inductor are given to demonstrate the practical value of the proposed algorithm.

A majority of nonlinear model reduction techniques can be regarded as a two step procedure as follows. First the state dimension is reduced through a projection, and then the vector field of the reduced state is approximated for improved computation efficiency. Neither of the above steps has been thoroughly studied. The second part of this thesis presents a solution to a particular problem in the second step above, namely, finding an upper bound of the system input/output error due to nonlinear vector field approximation. The system error upper bounding problem is formulated as an L2 gain upper bounding problem of some feedback interconnection, to which the small gain theorem can be applied. A numerical procedure based on integral quadratic constraint analysis and a theoretical statement based on L2 gain analysis are given to provide the solution to the error bounding problem. The numerical procedure is applied to analyze the vector field approximation quality of a transmission line with diodes.

The application of Volterra series to the reduced modeling of nonlinear systems is hampered by the rapidly increasing computation cost with respect to the degrees of the polynomials used. On the other hand, while it is less general than the Volterra series model, the Wiener-Hammerstein model has been shown to be useful for accurate and compact modeling of certain nonlinear sub-circuits such as power amplifiers. The third part of the thesis presents a convex optimization solution technique to the reduction/identification of the Wiener-Hammerstein system. The identification problem is formulated as a non-convex

quadratic program, which is solved by a semidefinite programming relaxation technique. It is demonstrated in the thesis that the formulation is robust with respect to noisy measurement, and the relaxation technique is oftentimes sufficient to provide good solutions. Simple examples are provided to demonstrate the use of the proposed identification algorithm.

Thesis Supervisor: Luca Daniel
Title: Associate Professor

Thesis Supervisor: Alexandre Megretski
Title: Professor

Acknowledgments

I would like to express my gratitude to my advisors Professor Luca Daniel and Professor Alexandre Megretski, who have shaped me into what I am like today. Luca has taught me a lot of electrical engineering aspects of my research as well as the attitude that I should take when I am faced with engineering challenges. In addition, Luca has also spent a lot of time helping me with my presentation skills, which I hope, will eventually match up to his standard. Alex, on the other hand, has profoundly influenced my way of conducting research. Through our discussions (and sometimes arguments), I finally started to learn the rope of conducting rigorous research. In addition, Alex has also shown to me that a lot of seemingly intractable problems can actually be solved, and I am oftentimes amazed by the creative solutions that he comes up with. In addition to my advisors, I would also like to thank Professor Munther Dahleh and Professor Jacob White for being my committee members and providing me with helpful suggestions. I would also like to acknowledge the support of the Interconnect Focus Center, one of the five research centers funded under the Focus Center Research Program, a Semiconductor Research Corporation and DARPA program.

Now it is time to mention my folks in the CP group. Sharing my office are Lei Zhang and Bo Kim. Lei is a good friend of mine and we hang out a lot. His personality changed the dynamics of the group – I witnessed the change of interactions between the people in the group before and after his arrival. Yan Li, who is Lei's wife, also happens to be my friend and we have had many enjoyable chats. Bo is a very charming lady who is also very nice to everybody that she comes across. I thank her for all those kind words and encouragements to the group members including me. Brad Bond is the buddy sitting initially behind me and then next door. We have had a lot of discussions of just about every topics, technical or non-technical. I also thank him for inviting me to his home in Tennessee during the wonderful Christmas in the year of 2006. Junghoon Lee also sits next door. He is one of the nicest guy I have ever met. Dmitry Vasilyev is also a good friend of mine. We have had lots of technical discussions, but we also have had lots of skiing trips together. Other members and former members of the group are also gratefully acknowledged. Tarek El Moselhy, Steve

Leibman, Laura Proctor, Homer Reid, Jaydeep Bardhan, Xin Hu, Shih-Hsien Kuo, Carlos Pinto Coelho, David Willis together made my stay in the CP group very enjoyable.

I thank my parents for their support and their constant updates of the home front. Without their sacrifice I could not have achieved what I have achieved today. I have been away from home for too long, and I think it must be very hard on them because of the separation. I sincerely appreciate their patience for waiting for me to complete my long journey to obtain my PhD.

The last semester at MIT has been particularly eventful for me. It has been laden with difficulties and setbacks for me, one after another. The trace of this half year will forever live in my mind. Especially memorable in this difficult semester is Qin, who proved to me, finally there is something I really care about.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivations

Model reduction is a widely accepted practice to facilitate system simulation and optimization. Different levels of success have been achieved depending on the specific model reduction applications. Algorithms for model reduction for linear time-invariant (LTI) system *analysis* have been successfully developed by many groups of researchers. For example, balanced truncation (or truncated balanced realization) [1, 2, 3] and the optimal Hankel norm model reduction [4] are expensive model reduction algorithms (by the standard of the electronic design automation community) but they are very accurate and possess nice theoretical guarantees such as reduced model stability and error bound. On the other hand, moment matching (Krylov subspace methods) [5, 6, 7, 8] and proper orthogonal decomposition [9] are relatively inexpensive model reduction algorithms, but they do not in general offer much guarantee in terms of ready-to-use accuracy measures (e.g., $\mathcal{H}_\infty$ norm error bound) or reduced model properties such as stability. Only in some special cases can the stability of the reduced models be assumed [8]. In addition, compromises between the two groups exist approximating the first group using the operations allowed in the second group [10, 11, 12]. All the aforementioned algorithms construct reduced models by operating on the state space representation (i.e., system matrices) of the full model and therefore are restricted to the model reduction problems of finite dimensional LTI systems. On the other hand, there are optimization/identification based model reduction algorithms which

directly find the coefficients of the reduced model without using the state space information of the full model. Rational transfer function fitting algorithms are well-known optimization based examples [13, 14]. In addition, rational transfer function fitting algorithms can enforce additional constraints such as stability and passivity. This will be shown in Chapter 2.

For the *design* and *optimization* of LTI systems, model reduction approaches have been less successful. One way to apply standard model reduction techniques to system design is to construct a reduced model for every full model ever considered by the design optimizer. This path tends to be time-consuming because typically a large number of full models have to be considered and reduced. Another way to apply model reduction techniques to system design is to construct *parameterized* reduced models. Once such reduced models have been constructed, the design optimization process can be greatly facilitated. Due to their popularity in the non-parameterized case, the moment matching techniques have been extended to the parameterized reduction case by many previous attempts [15, 16, 17, 18, 19, 20, 21, 22, 23]. One significant drawback of the moment matching based parameterized model reduction techniques is that to increase the accuracy of the reduced model, more moments need to be matched and this results in an increase in the order of the reduced model. The increase in order does not scales well with the number of parameters. On the other hand, optimization based techniques such as rational transfer function fitting can be generalized to the parameterized case, constructing reduced models with orders independent of the number of parameters, even if an increase in accuracy is desired. However, the challenge with rational transfer function fitting is that with constraints such as stability, the reduced model construction process can be very time-consuming (because the optimization problems are not convex in general). Therefore, the development of a stable reduced model generating rational transfer function fitting algorithm, which is efficient in both the model construction process and the simulation of the reduced models, would greatly benefit the design and optimization of LTI systems. The development of such an algorithm will be the main focus of Chapter 2.

The picture concerning the nonlinear model reduction problem is less clear simply because “nonlinear” is a very general collective term for systems other than LTI. First attempt approaches for nonlinear model reduction include trajectory piecewise linear/polynomial

based methods [24, 25, 26, 27, 28, 29, 30] and Volterra series based projection methods [31, 32, 33, 34, 35, 36]. Trajectory based methods can be considered as two step procedures as follows: first the dimension of the system state is reduced by a projection, then an approximation is made to the reduced vector field for efficient simulation. Volterra series based projection methods, on the other hand, first approximate the vector field using polynomials and then reduce the approximated model using projection schemes. To make a tradeoff between reduced model accuracy and complexity (time required for model simulation), it would be necessary to understand how to quantify the error in the two steps. While in some cases the projection error (e.g., trajectory piecewise linear method with balanced truncation [27]) can be quantified, the error due to vector field approximation (i.e., the second step in trajectory based methods and the first step in Volterra series based projection methods) is not very well-known. An attempt to solve the vector field approximation error estimation problem will be presented in Chapter 3.

Sometimes the only available information regarding the full model is its input and output measurements. On these occasions the projection based methods described above do not work. Instead, input/output based system identification techniques must be used to construct the reduced models. There is a very large body of input/output system identification techniques. See, for instance, [37, 38] for the descriptions of some of the techniques. The block diagram oriented identification technique based on the Wiener/Hammerstein/Wiener-Hammerstein structure is one of the most popular choices because of its simplicity, its ability to model complicated nonlinear effects, and its applicability to model realistic devices such as power amplifiers and RF amplifiers [39, 40, 41]. Being a classical problem, the identification of the Wiener and Hammerstein systems and their combinations has been considered in a large number of references [42, 43, 44, 45, 46, 47, 48]. However, very few of the aforementioned references actually consider the Wiener-Hammerstein identification problem itself (i.e., two LTI systems sandwiching a memoryless nonlinearity) because of the "non-separability" issue (i.e., the cascading of three blocks with unknown coefficients makes the identification task much more difficult than the Wiener or Hammerstein setup with only two unknown blocks). The non-separability issue is oftentimes addressed by making certain assumptions on one of the blocks (e.g., assuming the nonlinearity to be of

certain forms such as polynomial), which might make the approaches restrictive in some cases. On the other hand, if no assumptions are made, the resulting identification decision problem would be very difficult (e.g., non-convex), and in general it is solved by a general purpose solver which might not be efficient. The purpose of the third part of the thesis is to investigate whether the identification decision problem possesses any special properties due to the underlying Wiener-Hammerstein structure, and whether these properties can be exploited in facilitating the optimization solution process. Chapter 4 presents in detail the relevant results.

1.2 Dissertation Outline

The following three chapters contain the contributions of this thesis. In Chapter 2 a quasi-convex optimization based parameterized model reduction algorithm for LTI systems will be presented. In Chapter 3 the problem of bounding the system error due to an approximation to the nonlinear vector field will be considered. A convex optimization based numerical procedure and a theoretical statement will be given as solutions to the problem. In Chapter 4 a special case of the nonlinear model reduction problem, namely the Wiener-Hammerstein system identification problem, will be studied. A convex semidefinite programming based algorithm will be presented. Chapter 5 concludes the thesis.

Chapter 2

Model Order Reduction of Parameterized Linear Time-Invariant Systems via Quasi-Convex Optimization

Developing parameterized model order reduction (PMOR) algorithms would allow digital, mixed signal and RF analog designers to promptly instantiate field solver accurate small models for their parasitic dominated components (interconnect, RF inductors, MEM resonators etc.). The existing PMOR techniques are based either on statistical performance analysis [49, 50, 51, 52, 10] or on moment matching [15, 16, 17, 18, 19, 20, 21, 22, 23]. Some non-parameterized model order reduction or identification techniques based on an optimization approach are present in literature. References [53] and [54] identify systems from sampled data by essentially solving the Yule-Walker equation derived from a linear least squares problem. However, these methods might not be satisfactory since the objective of their minimization is not the norm of the difference between the original and reduced transfer functions, but rather the same quantity multiplied by the denominator of the reduced model. References [14] and [55] directly formulate the model reduction problem as a rational fit minimizing the $\mathcal{H}_2$ norm error, and therefore they solve a nonlinear least squares problem, which is not convex. To address the problem, those references propose solving linear least squares iteratively, but it is not clear whether the procedure will converge, and whether they can handle additional constraints such as positive real passiv-

ity. In order to reduce positive real systems, the authors of [13] propose using the KYP Lemma/semidefinite programming relationship [56], and show that the reduction problem can be cast into a semidefinite program, if the poles of the reduced models are given a priori. Reference [57] uses a different result derived from [58], to check positive realness. In that procedure, a set of scalar inequalities evaluated at some frequency points are checked. Reference [57] then suggests an iterative scheme that minimizes the $\mathcal{H}_2$ norm of the error system for the frequency points given in the previous iteration. However, this scheme does not necessarily generate optimal reduced models, since in order to do that, both the system model and the frequency points should be considered as decision variables. In short, the available methods lack one or more of the following desirable properties: rational fit, guaranteed stability and passivity, convexity, optimality or flexibility to impose constraints.

In principle, the method proposed in this thesis is a rational approximation based model reduction framework, but with the following three distinctions:

- Instead of solving the model reduction directly, the proposed methodology solves a relaxation of it.

- The objective function to be minimized is not the $\mathcal{H}_2$ norm, but rather the $\mathcal{H}_\infty$ norm. As it turns out, the resultant optimization problem, as described in Section 2.2, is equivalent to a quasi-convex program, i.e., an optimization of a quasi-convex function (all sub-level sets are convex sets) over a convex set. This property implies the following: 1) there exists a unique optimal solution to the problem; 2) the oftentimes efficient convex optimization solution techniques can be applied. Also, since the proposed method involves only a single optimization problem, it is near optimal with respect to the objective function used ($\mathcal{H}_\infty$ norm of error).

- In addition to the mentioned benefits, it will be demonstrated in the thesis that some commonly encountered constraints or additional objectives can be added to the proposed optimization setup *without* significantly increasing the complexity of the problem. Among these features are guaranteeing stability, positive realness (passivity of

impedance systems), bounded realness (passivity of scatter parameter systems), quality factor error minimality. Also, the optimization setup can be modified to generate an optimal parameterized reduced model that is stable for the range of parameters of interest.

The rest of this chapter is organized as follows. Section 2.1 provides some technical background. Section 2.2 describes the proposed relaxation and explains why it is quasi-convex after a change of decision variables. Section 2.3 gives an overview of the setup of the proposed method and some details of it. Section 2.4 demonstrates how to modify the basic optimization setup to incorporate various desirable constraints. Section 2.5 focuses on the extension of the optimization setup to the case of parameterized model order reduction. In Section 2.6 more design oriented modifications will be discussed. As a special case, the RF inductor design algorithm will be given. In Section 2.7 the complexity of the proposed algorithm is analyzed. In Section 2.8 several applications examples are shown to evaluate the practical value of the proposed method in terms of accuracy and complexity.

2.1 Technical Background

2.1.1 Tustin transform and continuous-time model reduction

In order to work with (rational) transfer functions in a numerically reliable way, the following standard procedure will be employed throughout the chapter: given a continuous-time (CT) system with transfer matrix $H_c(s)$, first apply a Tustin transform (e.g., [59]) to construct a discrete-time (DT) system $H(z) := H_c(s)|_{s=\lambda(z-1)/(z+1)}$ (with λ being a pre-specified real number, to be discussed), then construct a reduced DT system $\hat{H}(z)$ using the proposed model reduction technique, and finally apply the inverse Tustin transform to obtain the reduced CT system $\hat{H}_c(s) := \hat{H}(z)|_{z=(\lambda+s)/(\lambda-s)}$. The main benefit of the above procedure is that the transfer function coefficients of the optimally reduced DT model will be bounded, thus making the numerical procedure more robust. In addition, except for the somewhat arbitrary choice of the parameter λ in the Tustin transform, there is no obvious drawback for the model reduction procedure described above. Since the frequency

responses of the CT and DT systems are the frequency axis scaled versions of each other, there is an one-to-one correspondence between the ($\mathcal{H}_\infty$ norm) optimal reduced model in CT and DT with the same order. Consequently, model reduction settings for the rest of this chapter will be described in DT only.

The choice of the center frequency λ in the Tustin transform is somewhat arbitrary. While it is true that extreme choices (e.g., picking λ to be 1Hz, while the frequency range of interest is at 1GHz) can be harmful for the proposed model reduction framework, numerical experiments have shown that a broad choice of center frequencies would allow the proposed framework to work without suffering any CT/DT conversion problem. In fact, we have implemented, as part of the proposed model reduction algorithm, an automatic procedure that chooses the center frequency by minimizing the maximum slope of the magnitude of the frequency response, hence avoiding any possibly numerically harmful extreme situations.

2.1.2 $\mathcal{H}_\infty$ norm of a stable transfer function

For a stable transfer function $H(z) : \mathbb{C} \mapsto \mathbb{C}$, the $\mathcal{H}_\infty$ norm is defined as

$$\|H(z)\|_\infty := \sup_{\omega \in [0,2\pi)} \left|H(e^{j\omega})\right|. \tag{2.1}$$

The $\mathcal{H}_\infty$ norm for the multiple-input-multiple-output (MIMO) case with $H(z) : \mathbb{C} \mapsto \mathbb{C}^{p\times n}$ ($p \geq 1, n \geq 1$) is

$$\|H(z)\|_\infty := \sup_{\omega \in [0,2\pi)} \left\|H(e^{j\omega})\right\|_2. \tag{2.2}$$

The $\mathcal{H}_\infty$ norm can be thought of as the "amplification factor" of a system. In the context of model reduction, a reduced model $\hat{H}(z)$ is regarded as a good approximation to the full model $H(z)$ if the $\mathcal{H}_\infty$ norm of the difference $\left\|H(z) - \hat{H}(z)\right\|_\infty$ is small.

2.1.3 Optimal $\mathcal{H}_\infty$ norm model reduction problem

A reasonable model reduction problem formulation is the optimal $\mathcal{H}_\infty$ norm model reduction problem: given a stable transfer function $H(z)$ (possibly of large or even infinite order)

and an integer m (as the order of the reduced model), construct a stable rational transfer function with real coefficients

$$\hat{H}(z) = \frac{p(z)}{q(z)} := \frac{p_m z^m + p_{m-1} z^{m-1} + \ldots + p_0}{z^m + q_{m-1} z^{m-1} + \ldots + q_0}, \quad p_k \in \mathbb{R}, q_k \in \mathbb{R}, \forall k$$

such that order of $\hat{H}(z)$ is less than or equal to m, and the error $\left\| H(z) - \hat{H}(z) \right\|_\infty$ is minimized:

$$\begin{aligned} \underset{p,q}{\text{minimize}} \quad & \left\| H(z) - \tfrac{p(z)}{q(z)} \right\|_\infty \\ \text{subject to} \quad & \deg(q) = m, \quad \deg(p) \leq m, \\ & q(z) \neq 0, \quad \forall z \in \mathbb{C}, |z| \geq 1 \quad \text{(stability)}. \end{aligned} \tag{2.3}$$

Unfortunately, because of the stability constraint, program (2.3) is not a convex problem (see the next subsection for the definition). Up to now, no efficient algorithm for program (2.3) has been found.

2.1.4 Convex and quasi-convex optimization problems

This subsection will only describe the concepts necessary to the development of the thesis. For a more detailed description of the subject, consult, for example [60, 61].

A set $C \subset \mathbb{R}^n$ is said to be a convex set if

$$\alpha x + (1-\alpha) y \in C, \quad \forall x \in C, y \in C, \alpha \in [0,1].$$

In other words, a set C is convex if it contains the line segment connecting any two points in the set.

A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is said to be convex if

$$f(\alpha x_1 + (1-\alpha) x_2) \leq \alpha f(x_1) + (1-\alpha) f(x_2), \quad \forall x_1, x_2 \in \mathbb{R}^n, \alpha \in [0,1].$$

In other words, a function f is convex if the function value at any point along any line segment is below the corresponding linear interpolation between the function values at the

two end points. In addition, a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is concave is $-f$ is a convex function.

An optimization problem is said to be convex if it minimizes a convex objective function (or maximizes a concave objective function), and if the feasible set of the problem is convex. The nice property about a convex optimization problem is that any local optimum is also a global optimum. Convex optimization problems are oftentimes found to be efficiently solvable.

A relevant concept that will be explored in this chapter is the notion of a quasi-convex function. A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is quasi-convex if all its sub-level sets are convex sets. That is, the sets

$$\{x \in \mathbb{R}^n \,|\, f(x) \leq \gamma\} \text{ are convex}, \quad \forall \gamma \in \mathbb{R}.$$

The sub-level sets of a convex function are convex. Therefore, a convex function is automatically a quasi-convex function. However, the converse is not true. See Figure 2-1 for an illustration of a quasi-convex function which is not convex.

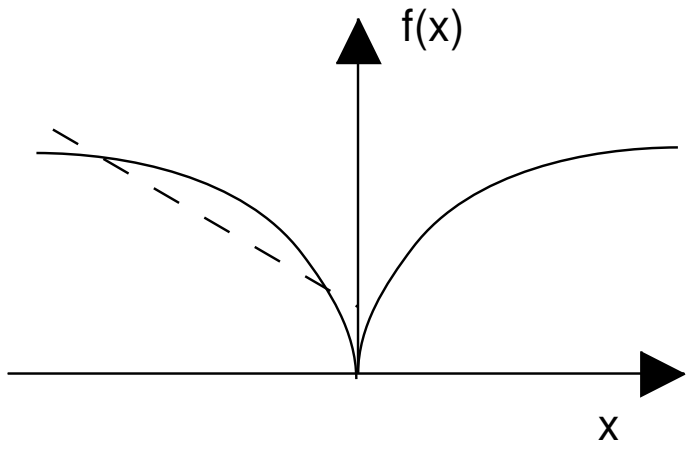


Figure 2-1: A one dimensional quasi-convex function which is not convex. All the sub-level sets of the function are (convex) intervals. However, the function values lie above the line segment (the dash line in the figure).

A quasi-convex optimization problem is a minimization problem of a quasi-convex function over a convex set. Quasi-convex optimization problems are not much more difficult to solve than convex problems. This is suggested by the fact that a local minimum of a quasi-convex problem is still a global minimum. In Sections 2.2 and 2.3 a specific class of

quasi-convex optimization problem will be identified, and an efficient algorithm to solve it will be detailed.

2.1.5 Relaxation of an optimization problem

While optimization provides a versatile framework for many model reduction decision problems, oftentimes the formulated optimization problems are difficult to solve (i.e., not convex). Formulating relaxations is a standard attempt to address the computation challenge above. A relaxation of an optimization problem is a related optimization problem such that an optimal solution to the original problem is a feasible solution to the relaxation. A relaxation can be introduced if it is much easier to solve, and the optimal solution to the relaxation is useful in constructing a reasonably good feasible solution to the original problem. However, note that such feasible solution might not be in general an optimal solution to the original problem. Typical ways for obtaining a relaxation include enlarging the feasible set and/or replacing the objective function with another (easier to optimize) function whose sub-level set contains the sub-level set of the original. It will be shown in Section 2.2 that the relaxation idea is useful in simplifying the proposed model reduction problem.

2.2 Relaxation Scheme Setup

This section describes the main theory of the proposed model reduction framework. The development of the framework is as follows: first a relaxation of (2.3) is proposed. Then a change of decision variables is introduced to the relaxation, and it can be shown that the relaxation is equivalent to a quasi-convex optimization problem, which happens to be readily solvable.

2.2.1 Relaxation of the $\mathcal{H}_\infty$ norm optimization

Motivated by the optimal Hankel norm model reduction [62], the following relaxation of the optimal $\mathcal{H}_\infty$ norm model reduction was proposed in [63]:

$$\begin{aligned} \underset{p,q,r}{\text{minimize}} \quad & \left\| H(z) - \frac{p(z)}{q(z)} - \frac{r(1/z)}{q(1/z)} \right\|_\infty \\ \text{subject to} \quad & \deg(q) = m, \quad \deg(p) \leq m, \quad \deg(r) < m \\ & q(z) \neq 0, \quad \forall z \in \mathbb{C}, |z| \geq 1 \quad \text{(stability)}. \end{aligned} \tag{2.4}$$

In program (2.4), an anti-stable rational part $\frac{r(1/z)}{q(1/z)}$, where r is a real coefficient polynomial of degree less than m, is added to the setup of (2.3). Because of the associated additional decision variables (i.e., the coefficients of polynomial r), program (2.4) is a relaxation of (2.3). After solving program (2.4), a (suboptimal) stable reduced model can simply be obtained as $\hat{H}(z) = \frac{p(z)}{q(z)}$. The following lemma, from [63], gives an error bound of the relaxation.

Lemma 2.2.1. *Let (p^*, q^*, r^*) be the optimal solution of program (2.4) with reduced order m,*

$$\gamma^* = \left\| H(z) - \frac{p^*(z)}{q^*(z)} - \frac{r^*(1/z)}{q^*(1/z)} \right\|_\infty$$

and

$$\hat{H}(z) := \frac{p^*(z)}{q^*(z)}$$

be a stable reduced model, then

$$\min_{D \in \mathbb{R}} \left\{ \left\| H(z) - \hat{H}(z) - D \right\|_\infty \right\} \leq (m+1)\gamma^*. \tag{2.5}$$

■

Remark 2.2.2. By definition γ^* is a lower bound of the error of the optimal $\mathcal{H}_\infty$ norm model reduction problem (2.3) and Lemma 2.2.1 states that the suboptimal reduced model provided by the proposed framework has an error upper bound $(m+1)$ times its error lower bound γ^*. In the lemma, $\hat{H}(z) := \frac{p^*(z)}{q^*(z)}$ is the outcome of the solving program (2.4) or

program (2.14), to be discussed in the next subsection. It should be noted that the scalar D in (2.5) can be incorporated into the reduced model $\hat{H}$, if $\hat{H}$ is not a strictly proper transfer function. Therefore the reduced model should really be understood as $\hat{H}(z)+D^*$ where D^* is chosen to be the optimizing D. In Section 2.3 procedure (2.26) will be discussed to construct a reduced model that always picks the optimizing D. ■

2.2.2 Change of decision variables in the relaxation scheme

The benefit of the relaxation (2.4) is not immediately obvious: program (2.4) still retains the non-convex stability constraint $q(z)\neq 0, \quad \forall z\in\mathbb{C}, |z|\geq 1$. More formally, it can be stated that the set of the coefficients of the polynomials,

$$\begin{aligned}\Omega^m_{qpr} \quad := \quad & \Big\{ (\vec{q},\vec{p},\vec{r})\in\mathbb{R}^m\times\mathbb{R}^{m+1}\times\mathbb{R}^m : \\ & q(z)=z^m+\vec{q}_{m-1}z^{m-1}+\ldots+\vec{q}_1 z+\vec{q}_0 \\ & p(z)=\vec{p}_m z^m+\vec{p}_{m-1}z^{m-1}+\ldots+\vec{p}_1 z+\vec{p}_0 \\ & r(z)=\vec{r}_{m-1}z^{m-1}+\vec{r}_{m-2}z^{m-2}+\ldots+\vec{r}_1 z+\vec{r}_0 \\ & \text{satisfying} \quad q(z)\neq 0, \quad \forall z\in\mathbb{C}:|z|\geq 1 \Big\}\end{aligned} \tag{2.6}$$

is not convex if $m>2$. As the first step to address the non-convexity difficulty, the following set of decision variables is proposed,

$$\begin{aligned}\Omega^m_{abc} \quad := \quad & \Big\{ \left(\vec{a},\vec{b},\vec{c}\right)\in\mathbb{R}^m\times\mathbb{R}^{m+1}\times\mathbb{R}^m : \\ & a(z)=\vec{a}_m(z^m+z^{-m})+\vec{a}_{m-1}(z^{m-1}+z^{-m+1})+\ldots+1 \\ & b(z)=\vec{b}_m(z^m+z^{-m})+\vec{b}_{m-1}(z^{m-1}+z^{-m+1})+\ldots+\vec{b}_0 \\ & c(z)=\tfrac{1}{j}\left(\vec{c}_m(z^m-z^{-m})+\ldots+\vec{c}_1(z-z^{-1})\right) \\ & \text{satisfying} \quad a(z)>0 \quad \forall z\in\mathbb{C}:|z|=1. \Big\}\end{aligned} \tag{2.7}$$

Note that the coefficient q_m in eq. (2.6) is normalized to one because stability condition (i.e., $\frac{p(z)}{q(z)}$ cannot have a pole at infinity) does not allow it to be zero. Likewise, the coefficient a_0 in eq. (2.7) is also normalized to one because positivity condition (i.e., $a_0=\int_0^{2\pi} a\left(e^{j\omega}\right)d\omega$) does not allow it to be zero. However, it should be pointed out that

in eq. (2.7), there is no normalization for a_m. In particular, it can be zero and the degree of $a(z)$ can be strictly less than m. The following lemma defines an one-to-one correspondence between the sets Ω^m_{qpr} and Ω^m_{abc}, and hence suggesting that both sets can be used to completely characterize the set of all reduced models in optimization problem (2.4).

Lemma 2.2.3. *Define $\tau_m : \Omega^m_{qpr} \mapsto \Omega^m_{abc}$ as follows:*

- *Given $(\vec{q},\vec{p},\vec{r}) \in \Omega^m_{qpr}$, $\left(\vec{a},\vec{b},\vec{c}\right) = \tau_m(\vec{q},\vec{p},\vec{r}) \in \Omega^m_{abc}$ is defined as follows: denote $D := \left(1+\sum_{k=0}^{m-1}(\vec{q}_k)^2\right)^{-1}$, then $\left(\vec{a},\vec{b},\vec{c}\right)$ are defined as the coefficients of the trigonometric polynomials*

$$\begin{aligned} a(z) &= Dq(z)q\left(z^{-1}\right) \\ b(z) &= \tfrac{D}{2}[p(z)q\left(z^{-1}\right)+q(z)r\left(z^{-1}\right)+p\left(z^{-1}\right)q(z)+q\left(z^{-1}\right)r(z)] \\ c(z) &= \tfrac{D}{2j}[p(z)q\left(z^{-1}\right)+q(z)r\left(z^{-1}\right)-p\left(z^{-1}\right)q(z)-q\left(z^{-1}\right)r(z)]. \end{aligned} \tag{2.8}$$

- *Given $\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{abc}$, $(\vec{q},\vec{p},\vec{r}) = {\tau_m}^{-1}\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{qpr}$ is defined as follows: let $\hat{m} \in \{0,1,\ldots,m\}$ be the degree of $a(z)$ in eq. (2.7), and let $z_k, k = 1,\ldots,\hat{m}$ be the (maybe repeated) roots of the ordinary polynomial $z^{\hat{m}}a(z)$ such that $|z_k| < 1$. Then $\vec{q}$ is defined as the coefficients of the polynomial*

$$q(z) := z^{m-\hat{m}}\prod_{k=1}^{\hat{m}}(z-z_k). \tag{2.9}$$

Denote $D := \left(1+\sum_{k=0}^{m-1}(\vec{q}_k)^2\right)^{-1}$, then $\vec{p},\vec{r}$ are uniquely defined by

$$D\left(p(z)q(z^{-1})+q(z)r(z^{-1})\right) = b(z)+jc(z). \tag{2.10}$$

Then

1. *The map τ_m is one-to-one with the inverse as ${\tau_m}^{-1}$.*
2. *The map τ_m satisfies the following frequency response matching property:*

$$H(e^{j\omega}) = \frac{p(e^{j\omega})}{q(e^{j\omega})}+\frac{r(e^{-j\omega})}{q(e^{-j\omega})} = \frac{b(e^{j\omega})+jc(e^{j\omega})}{a(e^{j\omega})}, 0 \le \omega < 2\pi. \tag{2.11}$$

■

Proof of Lemma 2.2.3. The proof of the lemma is divided into three steps:

Step 1 shows that the definitions of the maps τ_m and ${\tau_m}^{-1}$ "make sense". That is, given $(\vec{q},\vec{p},\vec{r}) \in \Omega^m_{qpr}$, the operation of applying τ_m is valid, and it should be true that $\left(\vec{a},\vec{b},\vec{c}\right) := \tau_m(\vec{q},\vec{p},\vec{r}) \in \Omega^m_{abc}$. Conversely, given $\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{abc}$, the operation of ${\tau_m}^{-1}$ is valid, and it should be true that $(\vec{q},\vec{p},\vec{r}) = {\tau_m}^{-1}\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{qpr}$.

The first statement can be verified simply by applying the definition in eq. (2.8).

For the second statement, suppose $\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{abc}$ is given. First show that the operation in eq. (2.9) is always valid, and $q(z)$ thus obtained satisfies the condition in eq. (2.6). Let $\hat{m}$ be the degree of $a(z)$, and define $\hat{a}(z)$ as

$$\hat{a}(z) := z^{\hat{m}}a(z) = \vec{a}_{\hat{m}}\left(z^{2\hat{m}}+1\right) + \vec{a}_{\hat{m}-1}\left(z^{2\hat{m}-1}+z\right) + \ldots + z^{\hat{m}}.$$

The following properties of the roots of $\hat{a}(z)$ can be concluded:

- Being an ordinary polynomial of degree $2\hat{m}$, $\hat{a}(z)$ has $2\hat{m}$ roots.
- Since $\vec{a}_{\hat{m}} \neq 0$, the origin (i.e., $0 \in \mathbb{C}$) cannot be a root of $\hat{a}(z)$. Therefore, $z_0 \in \mathbb{C}$ is a root of $\hat{a}(z)$ if and only if $a(z_0) = 0$.
- Since $\hat{a}(z)$ has real coefficients and $a(z) = a\left(z^{-1}\right)$, the following two cases are true: if $z_0 \in \mathbb{C}\setminus\mathbb{R}$ is a root of $\hat{a}(z)$, then so are ${z_0}'$, $\frac{1}{z_0}$ and $\frac{1}{{z_0}'}$, where $'$ is complex conjugate for $z_0 \in \mathbb{C}$. On the other hand, if $z_0 \in \mathbb{R}$ is a root of $\hat{a}(z)$, then so is $\frac{1}{z_0}$.
- Since $a(z) > 0, \forall\, |z| = 1$, there is no unit circle roots of $\hat{a}(z)$.

The four properties above imply that there are exactly $\hat{m}$ stable roots and $\hat{m}$ anti-stable roots of $\hat{a}(z)$ as the "unit circle mirror images" of the former (e.g., $1+2j$ and $0.2+0.4j$). Moreover, all roots with nonzero imaginary parts come in complex conjugate pairs. This concludes that the $\hat{m}$ roots described in eq. (2.9) can always be found, and $q(z)$ defined in eq. (2.9) has real coefficients polynomial of degree m, and all roots of $q(z)$ are stable (i.e., $q(z) \neq 0, \forall |z| \geq 1$).

To conclude the proof of **step 1**, it remains to be shown that when $\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{abc}$ is given and $\vec{q}$ has been found by eq. (2.9), $(\vec{p},\vec{r}) \in \mathbb{R}^{2m+1}$ can be found as the coefficients of $p(z)$ and $r(z)$ using eq. (2.10). First recognize that eq. (2.10) defines a linear function $M_{\vec{q}} : \mathbb{R}^{2m+1} \mapsto \mathbb{R}^{2m+1}$ such that

$$M_{\vec{q}}\left(\vec{p},\vec{r}\right) = \left(\vec{b},\vec{c}\right). \tag{2.12}$$

Then it is sufficient to prove that $M_{\vec{q}}$ is invertible. That is,

$$\mathrm{Ker}\left(M_{\vec{q}}\right) = 0. \tag{2.13}$$

To show eq. (2.13), consider $(\vec{p}^\star,\vec{r}^\star)$, corresponding to $p^\star(z)$ and $r^\star\left(z^{-1}\right)$ such that

$$p^\star(z)q\left(z^{-1}\right) \equiv -q(z)r^\star\left(z^{-1}\right).$$

The fact that $q\left(z^{-1}\right)$ in the LHS has m anti-stable roots and $q(z)$ in the RHS has no anti-stable roots implies that $r^\star\left(z^{-1}\right)$ should be m anti-stable roots. However, since the degree of $r^\star$ is strictly less than m, $r^\star$ should be zero and $p^\star$ should also be zero. This concludes that $(\vec{p}^\star,\vec{r}^\star) = 0 \in \mathbb{R}^{2m+1}$, showing that $M_{\vec{q}}$ is invertible and concluding **step 1**.

Step 2 shows that the map τ_m is one-to-one. For this purpose, it suffices to show the following: for every $(\vec{q},\vec{p},\vec{r}) \in \Omega^m_{qpr}$, if $\left(\hat{\vec{q}},\hat{\vec{p}},\hat{\vec{r}}\right) := {\tau_m}^{-1}\left(\tau_m\left(\vec{q},\vec{p},\vec{r}\right)\right)$, then $\left(\hat{\vec{q}},\hat{\vec{p}},\hat{\vec{r}}\right) = (\vec{q},\vec{p},\vec{r})$. First show that $\hat{\vec{q}} = \vec{q}$. Let $\hat{m} \in \{0,1,\ldots,m\}$ be the number of nonzero root of $q(z)$, then $q(z) = z^{m-\hat{m}}\prod_{k=1}^{\hat{m}}(z-z_k)$. Applying τ_m to $(\vec{q},\vec{p},\vec{r})$ results in a $a(z)$ with a *known* form. That is, $a(z) = D\prod_{k=1}^{\hat{m}}(z-z_k)\left(z^{-1}-z_k\right)$, with $D = \left(1+\sum_{k=0}^{m-1}\left(\vec{q}_k\right)^2\right)^{-1}$. Then the ordinary polynomial $z^{\hat{m}}a(z)$ in eq. (2.9) has exactly $\hat{m}$ stable roots (i.e., with magnitude less than one), and they are the roots of $p(z)$ (i.e., z_k for $k = 1,2,\ldots,\hat{m}$). Therefore, corresponding to $\hat{\vec{q}}$, the polynomial $\hat{q(z)} := z^{m-\hat{m}}\prod_{k=1}^{\hat{m}}(z-z_k)$, is exactly the same as $q(z)$, implying that $\hat{\vec{q}} = \vec{q}$. It remains to show that $\left(\hat{\vec{p}},\hat{\vec{r}}\right) = (\vec{p},\vec{r})$. This is true because, for any $\vec{q}$, the map $M_{\vec{q}}$

defined in eq. (2.12) is invertible. Then,

$$\left(\hat{\vec{p}},\hat{\vec{r}}\right) = \left(M_{\vec{q}}\right)^{-1} M_{\vec{q}}\left(\vec{p},\vec{r}\right) = \left(\vec{p},\vec{r}\right),$$

hence $\left(\hat{\vec{q}},\hat{\vec{p}},\hat{\vec{r}}\right) = \left(\vec{q},\vec{p},\vec{r}\right)$. This concludes **step 2**.

Finally, **step 3** shows that the frequency matching condition in eq. (2.11) holds. Given $\left(\vec{q},\vec{p},\vec{r}\right) \in \Omega^m_{qpr}$, then simply by checking the definition in eq. (2.8), it can be verified that $\left(\vec{a},\vec{b},\vec{c}\right) := \tau_m\left(\vec{q},\vec{p},\vec{r}\right)$ satisfies eq. (2.11).

Given, $\left(\vec{a},\vec{b},\vec{c}\right) \in \Omega^m_{abc}$, because of the matching (up to the constant multiplicative factor D) of the numerator of eq. (2.11) by the definition in eq. (2.10), it suffices to show that $\vec{q}$, as part of ${\tau_m}^{-1}\left(\vec{a},\vec{b},\vec{c}\right)$, satisfies the denominator matching of eq. (2.11) (i.e., $a(z) = Dq(z)q(z^{-1})$). To show this, notice that for $q(z)$ defined in eq. (2.9), $Dz^{\hat{m}}q(z)q\left(z^{-1}\right)$ is an ordinary polynomial with exactly the same (stable and anti-stable) roots of $z^{\hat{m}}a(z)$ because of the "unit circle mirror image" property of the roots of $z^{\hat{m}}a(z)$ shown in **step 1**. That means that the coefficients of $Dz^{\hat{m}}q(z)q\left(z^{-1}\right)$ and $z^{\hat{m}}a(z)$ can at worst be off by a constant multiplicative factor C. The coefficient of the monomial $z^{\hat{m}}$ of $z^{\hat{m}}a(z)$ is one by the definition in eq. (2.7). On the other hand, expressing $q(z)$ as $q(z) = z^m + \vec{q}_{m-1}z^{m-1} + \ldots + \vec{q}_0$, it can be seen that the coefficient of the monomial $z^{\hat{m}}$ in $Dz^{\hat{m}}q(z)q\left(z^{-1}\right)$ is also one, when $D := \left(1 + \sum_{k=0}^{m-1}\left(\vec{q}_k\right)^2\right)^{-1}$. Hence, the multiplicative factor C is one, and therefore $q(z)$, together with $a(z)$ satisfies the matching of $a(z) = Dq(z)q(z^{-1})$ in eq. (2.11). This concludes **step 3** and the proof of the lemma. ■

Remark 2.2.4. Lemma 2.2.3 states that both sets Ω^m_{qrp} in eq. (2.6) and Ω^m_{abc} in eq. (2.7) can completely characterize the relaxed model reduction problem in program (2.4). In addition, the stability constraint $q(z) \neq 0, \forall z \in \mathbb{C} : |z| \geq 1$ in (2.6), which makes the feasible set of (2.3) non-convex, can be replaced by the easier to handle (to be shown) positivity constraint $a(z) > 0, \forall z \in \mathbb{C} : |z| = 1$, and this paves way to the discovery of efficient algorithms for solving the relaxation problem. ■

Remark 2.2.5. Since the evaluation of z in the positivity constraint in eq. (2.7) is restricted to the unit circle only, for the model reduction problem in program (2.4), the evaluation of z can also be restricted to the unit circle because it is where the frequency response is

evaluated. Therefore, denoting $z = e^{j\omega} = \cos(\omega) + j\sin(\omega)$, program (2.4) is equivalent to

$$\begin{aligned} \underset{\tilde{a},\tilde{b},\tilde{c},\gamma}{\text{minimize}} \quad & \gamma \\ \text{subject to} \quad & |H(e^{j\omega})\tilde{a}(\omega) - \tilde{b}(\omega) - j\tilde{c}(\omega)| < \gamma\tilde{a}(\omega),\ 0 \le \omega < 2\pi, \\ & \tilde{a}(\omega) > 0, \quad 0 \le \omega < 2\pi, \\ & \deg(\tilde{a}) \le m, \deg(\tilde{b}) \le m, \deg(\tilde{c}) \le m, \end{aligned} \tag{2.14}$$

with

$$\begin{aligned} \tilde{a}(\omega) &= 1 + \tilde{a}_1\cos(\omega) + \ldots + \tilde{a}_m\cos(m\omega), \\ \tilde{b}(\omega) &= \tilde{b}_0 + \tilde{b}_1\cos(\omega) + \ldots + \tilde{b}_m\cos(m\omega) \\ \tilde{c}(\omega) &= \tilde{c}_1\sin(\omega) + \ldots + \tilde{c}_m\sin(m\omega). \end{aligned} \tag{2.15}$$

Because of the trigonometric terms, polynomials in eq. (2.15) (and in eq. (2.7)), are called trigonometric polynomials of degree m. The following lemma justifies the change of variables introduced by Lemma 2.2.3 in terms of possible computational efficiency gain. ■

Lemma 2.2.6. *Program (2.14) is quasi-convex (i.e., minimization of a quasi-convex function over a convex set).* ■

Proof of Lemma 2.2.6. First note that $\tilde{a}(\omega) > 0, \forall\, \omega \in [0, 2\pi)$ defines the intersection of infinitely many halfspaces (each defined by a particular $\omega \in [0, 2\pi)$) and therefore the feasible set is convex. Secondly, consider a sub-level set of the objective function (for any *fixed* γ). Since

$$|z| = \max_{|\theta|=1} \mathrm{Re}(\theta z), \quad \forall z \in \mathbb{C},$$

condition

$$|H(e^{j\omega})\tilde{a}(\omega) - \tilde{b}(\omega) - j\tilde{c}(\omega)| < \gamma\tilde{a}(\omega), \forall\, \omega \in [0, 2\pi)$$

is equivalent to

$$\mathrm{Re}\Big(\theta\big(H(e^{j\omega})\tilde{a}(\omega) - \tilde{b}(\omega) - j\tilde{c}(\omega)\big)\Big) < \gamma\tilde{a}(\omega), \quad \forall\, \omega \in [0, 2\pi),\ |\theta| = 1, \tag{2.16}$$

which is the intersection of halfspaces parameterized by θ and ω. Therefore, the sub-level sets of the objective function of program (2.14) is convex and the quasi-convexity of the program is established. ■

Remark 2.2.7. Quasi-convex program (2.14) happens to be polynomially solvable. A description of how to solve the relaxation, as well as how this fits in the general picture of the proposed model reduction algorithm, will be discussed in the next section. Finally, it should be emphasized that not all quasi-convex programs are efficiently solvable. This is the case for the parameterized model reduction problem to be discussed in Section 2.5. ■

2.3 Model Reduction Setup

This section deals with the solution procedure of the proposed model reduction framework. A summary of the procedure is given as follows.

Algorithm 1: MOR
Input: $H(z)$
Output: $\hat{H}(z)$

i. Solve program (2.14) using a cutting plane algorithm (details in Subsection 2.3.1) to obtain the relaxation solution $(\tilde{a}, \tilde{b}, \tilde{c})$.

ii. Compute the denominator $q(z)$ using spectral factorization eq. (2.9).

iii. Solve a convex optimization problem to obtain the numerator $p(z)$. See Subsection 2.3.3.

iv. Synthesize a state space realization of the reduced model $\hat{H}(z) = p(z)/q(z)$. See [59] for details.

Step **i.** will be explained in Subsections 2.3.1 and 2.3.2. Step **iii.** will be explained in Subsection 2.3.3.

2.3.1 Cutting plane methods

Program (2.14) is a quasi-convex program with infinitely many constraints, and in general it can be solved by the cutting plane methods. This subsection will provide a general

description of the cutting plane methods, and their application to solving program (2.14) will be discussed in the subsequent parts of this chapter (Subsection 2.3.2 and Section 2.4).

Note that the cutting plane method is a standard optimization solution technique for quasi-convex problems, and it is given here for completeness. The cutting plane method solves the following problem: find a point in a target set X (e.g., the sub-optimal level set of a minimization problem), or verify that X is empty. The basic algorithm description is as follows.

a. Initialize the algorithm by finding an initial bounding set $\mathcal{P}_1$ such that $X \subset \mathcal{P}_1$.

b. At each step k, maintain a localization set $\mathcal{P}_k$ such that $X \subset \mathcal{P}_k$.

c. Compute a query point $x_k \in \mathcal{P}_k$. This is the current trial of the vector of the decision variables. Check if $x_k \in X$.

d. If $x_k \in X$, then terminate the algorithm and return x_k. Otherwise, return a "cut" (e.g., a hyperplane) such that all points in X must be in one side of the hyperplane (i.e., a halfspace). Denote the corresponding halfspace $\mathcal{H}$.

e. Update the localization set to $\mathcal{P}_{k+1}$ such that $\mathcal{P}_k \cap \mathcal{H} \subset \mathcal{P}_{k+1}$,

f. If Volume$(\mathcal{P}_{k+1}) < \varepsilon$, for some small ε (which, for instance, is determined by the desired sub-optimality level), then assert X is empty, and terminate the algorithm. Otherwise, go back to step **b**.

The choice of the localization set $\mathcal{P}_k$ and the query point x_k distinguishes one method from another. Reasonable choice of localization set/query point can be 1) a covering ellipsoid/center of the ellipsoid or 2) covering polytope/analytic center of the polytope. The former choice results in the ellipsoid algorithm (see [64] or [65] for detailed reference), while the latter choice results in the analytic center cutting plane method (ACCPM) (see [66] for reference). The finding of the initial bounding set $\mathcal{P}_1 : X \subset \mathcal{P}_1$ is problem dependent, and it will be discussed in the next subsection, in the context of program (2.14).

Step **a**. and step **d**. are the only steps in the cutting plane algorithm that are determined by the optimization problem to be solved. They will be discussed, in the context of program

(2.14), in Subsection 2.3.2 and Section 2.4, respectively. The subroutine implemented in step **d**. is typically referred to as an oracle. While the cutting plane algorithm is guaranteed to terminate in the number of iterations which scales polynomially to the problem size, the computation requirement of the oracle can range from light (e.g., the non-parameterized MOR case) to heavy (e.g., the parameterized MOR case).

Finally, it is noted that quasi-convex program (2.14) can also be solved as a semi-definite program (SDP) by interior point methods [67]. However, the discussion of this implementation will not be discussed in this thesis.

2.3.2 Solving the relaxation via the cutting plane method

In the context of solving the quasi-convex program (2.14) in Subsection 2.2.2, the description of the cutting plane method introduced in Subsection 2.3.1 can be more specific: the decision variables x in Subsection 2.3.1 are the coefficients of the trigonometric polynomials $\tilde{a}(\omega)$, $\tilde{b}(\omega)$ and $\tilde{c}(\omega)$. The target set X in Subsection 2.3.1 would be the set of trigonometric polynomial coefficients such that (2.14) is feasible (in particular, the stability constraint $\tilde{a}(\omega) > 0$ is satisfied) and the objective value γ can achieve its minimum (in practice, γ is allowed to be within a few percents above the minimum).

A simple strategy to obtain an initial bounding set (i.e., $\mathcal{P}_1$ in Subsection 2.3.1) is merely to assume it to be a "large enough" sphere. This is reasonable for most cases even though there is no real guarantee that it will work. However, for program (2.14), it is actually possible to find an initial bounding set which guarantees to contain the target set. The result is summarized in the following two statements.

Lemma 2.3.1. *Let $\tilde{a}_k$, $k = 1,2,\ldots,m$ be the coefficients of the trigonometric polynomial $\tilde{a}(\omega)$ in program (2.14), then the stability constraint $\tilde{a}(\omega) > 0, \forall\, \omega \in [0,2\pi)$ implies that $|a_k| \leq 2, \forall k = 1,2,\ldots,m$.* ■

Proof of Lemma 2.3.1. The stability constraint

$$\tilde{a}(\omega) = 1 + \tilde{a}_1 \cos(\omega) + \ldots + \tilde{a}_m \cos(m\omega) > 0, \quad \forall\, \omega \in [0,2\pi) \tag{2.17}$$

implies that

$$\int_0^{2\pi} \tilde{a}(\omega)\,(1+\cos(k\omega))\,d\omega \geq 0, \quad \forall k = 1,2,\ldots,m, \tag{2.18}$$

which (by the orthogonality of cosine) implies that

$$\tilde{a}_k \geq -2, \quad \forall k = 1,2,\ldots,m. \tag{2.19}$$

Similarly, eq. (2.17) also implies that

$$\int_0^{2\pi} \tilde{a}(\omega)\,(1-\cos(k\omega))\,d\omega \geq 0, \quad \forall k = 1,2,\ldots,m, \tag{2.20}$$

which in turns implies

$$\tilde{a}_k \leq 2, \quad \forall k = 1,2,\ldots,m. \tag{2.21}$$

Eq. (2.19) and (2.21) combined yields the desired result. ■

Lemma 2.3.2. *Let $\tilde{a}_k$, $\tilde{b}_k$ and $\tilde{c}_k$ be the trigonometric polynomial coefficients defined as in eq. (2.15) in program (2.14). Let $H(z)$ be any stable transfer function, and γ be any nonnegative number. Under the stability constraint $\tilde{a}(\omega) > 0, \forall\,\omega \in [0,2\pi)$, if it is true that*

$$\left\| \frac{\tilde{b}(\omega) + j\tilde{c}(\omega)}{\tilde{a}(\omega)} - H(e^{j\omega}) \right\|_\infty \leq \gamma. \tag{2.22}$$

Then

1. $\left|\tilde{b}_k\right| \leq 2\,(2m+1)\,(\|H(z)\|_\infty + \gamma), \quad \forall k = 0,1,\ldots,m.$
2. $|\tilde{c}_k| \leq 2\,(2m+1)\,(\|H(z)\|_\infty + \gamma), \quad \forall k = 1,2,\ldots,m.$

■

Proof of Lemma 2.3.2. First prove the first statement. Eq. (2.22) implies that

$$\left| \frac{\tilde{b}(\omega)}{\tilde{a}(\omega)} - \mathrm{Re}\left[H(e^{j\omega})\right] \right| \leq \gamma, \quad \forall\,\omega \in [0,2\pi) \tag{2.23}$$

because for any complex number $x \in \mathbb{C}$, $|\mathrm{Re}\,[x]| \leq |x|$. Eq. (2.23), together with the trian-

gular inequality, implies

$$\left|\tilde{b}(\omega)\right| - \left|\mathrm{Re}\left[H(e^{j\omega})\right]\right| |\tilde{a}(\omega)| \leq \gamma |\tilde{a}(\omega)|, \quad \forall \omega \in [0, 2\pi).$$

This in turns implies, as $\left|H(e^{j\omega})\right| \leq \|H(z)\|_\infty, \forall \omega \in [0, 2\pi)$, that

$$\left|\tilde{b}(\omega)\right| \leq |\tilde{a}(\omega)| \left(\|H(z)\|_\infty + \gamma\right), \quad \forall \omega \in [0, 2\pi).$$

Applying Lemma 2.3.1, it can be concluded from above that

$$\left|\tilde{b}(\omega)\right| \leq (2m+1) \left(\|H(z)\|_\infty + \gamma\right), \quad \forall \omega \in [0, 2\pi). \tag{2.24}$$

From eq. (2.24) it can be seen that

$$\begin{aligned} \left|\int_0^{2\pi} \tilde{b}(\omega) (1 + \cos(k\omega)) \, d\omega\right| &\leq 2\pi (2m+1) \left(\|H(z)\|_\infty + \gamma\right), \quad k = 0, 1, \ldots, m \\ \left|\int_0^{2\pi} \tilde{b}(\omega) (1 - \cos(k\omega)) \, d\omega\right| &\leq 2\pi (2m+1) \left(\|H(z)\|_\infty + \gamma\right), \quad k = 0, 1, \ldots, m \end{aligned} \tag{2.25}$$

Similar to the proof of Lemma 2.3.1, by applying the orthogonality of cosine, it can be concluded that

$$\begin{aligned} \left|\tilde{b}_0\right| &\leq (2m+1) \left(\|H(z)\|_\infty + \gamma\right) \\ \left|\tilde{b}_k\right| &\leq 2 (2m+1) \left(\|H(z)\|_\infty + \gamma\right) - 2, \quad \forall k = 1, 2, \ldots, m, \end{aligned}$$

which yields the desired result for the first statement in the Lemma.

The proof of the second statement is analogous to that of the first statement. Only the main steps are highlighted here. It can be concluded that

$$|\tilde{c}(\omega)| \leq |\tilde{a}(\omega)| \left(\|H(z)\|_\infty + \gamma\right), \quad \forall \omega \in [0, 2\pi).$$

Then using an approach analogous to eq. (2.25) with the "multipliers" $(1 \pm \sin(k\omega))$, the conclusion of the second statement can be made. ■

Remark 2.3.3. Lemma 2.3.1 can directly be applied to obtain a hypercube for bounding the

coefficients of $\tilde{a}_k$. To compute the bounds for the coefficients $\tilde{b}_k$ and $\tilde{c}_k$, Lemma 2.3.2 can be applied with $\gamma = \|H(z)\|_\infty$, corresponding to the objective value of a trial in which the coefficients $\tilde{b}_k$ and $\tilde{c}_k$ are set to zero. ■

2.3.3 Constructing the reduced model

Once the quasi-convex relaxation problem (2.14) has been solved, by for instance, the cutting plane method described in Subsections 2.3.1 and 2.3.2, the reduced model can be constructed: the denominator $q(z)$ and the numerator $p(z)$ of the reduced model could be found by applying eq. (2.9) and eq. (2.10) in Lemma 2.2.3. However, the following more practical procedure yields a reduced model whose approximation quality is no worse than the one obtained with (2.10): once $q(z)$ is found, calculate $p(z)$ as the optimal solution to the following program

$$\begin{aligned} \underset{p,\gamma}{\text{minimize}} \quad & \gamma \\ \text{subject to} \quad & \left| H(e^{j\omega}) - \tfrac{p(e^{j\omega})}{q(e^{j\omega})} \right| < \gamma, \forall\, \omega \in [0, 2\pi), \\ & \deg(p) \leq m. \end{aligned} \tag{2.26}$$

Note that program (2.26) is convex and can be solved by the same cutting plane method described in Subsections 2.3.1 and 2.3.2. Also note that since the degree of the numerator p can be m, the transfer function is not strictly proper, and the optimal constant term D in (2.5) is automatically chosen when program (2.26) is solved.

2.3.4 Obtaining models of increasing orders

In the proposed model reduction framework, the information from an order m model reduction can be reused to find the reduced models of order $m+k$ (with $k > 0$) relatively cheaply. The update procedure for order $m+1$ reduced model is described here (the procedure for higher order reduced models is the same). Suppose $(\tilde{a}_m^*, \tilde{b}_m^*, \tilde{c}_m^*)$ is the optimal trigonometric polynomials for order m reduction, and assume the corresponding error is

γ_m^*, then

$$\frac{\tilde{b}_m^*(\omega)+0\cdot\cos((m+1)\omega)+j\,(\tilde{c}_m^*(\omega)+0\cdot\sin((m+1)\omega))}{\tilde{a}_m^*(\omega)+0\cdot\cos((m+1)\omega)}$$

is automatically a valid (stable, passive, etc) candidate for the order $m+1$ reduction problem. Therefore it can be used as the initial center of the localization set (e.g., covering ellipsoid) for the $m+1$ order problem. The localization set for the $m+1$ order problem can also be inherited from that of the order m problem by appending the previous localization set in the following way. Let x_m be the vector of decision variables of the order m problem, x_m^* be coefficients of the optimal trigonometric polynomials $(\tilde{a}_m^*, \tilde{b}_m^*, \tilde{c}_m^*)$ of order m and P_m^* be the symmetric positive semi-definite matrix that defines the ellipsoid of the order m localization set, then

$$(x_m - x_m^*)' P_m^* (x_m - x_m^*) \leq 1$$

Now let $x_a^{m+1}, x_b^{m+1}, x_c^{m+1}$ be the coefficients of the $m+1$ degree terms in the $m+1$ degree trigonometric polynomials of the $m+1$ order reduction problem. If there exists some $M > 0$ s.t. $|x_a^{m+1}| < M$, $|x_b^{m+1}| < M$, $|x_c^{m+1}| < M$ then

$$(x_m - x_m^*)' P_m^* (x_m - x_m^*) + |x_a^{m+1}|^2 + |x_b^{m+1}|^2 + |x_c^{m+1}|^2 \leq 1 + 3M^2$$

can be used as the initial ellipsoid (i.e. localization set) for the $m+1$ model reduction problem. The order m optimal objective value γ_m^* can be used as the initial objective value when the $m+1$ order procedure starts. Using these initial iterates for the $m+1$ order problem, relatively few cuts will be required to obtain the $m+1$ order optimal trigonometric polynomials.

2.4 Constructing Oracles

The oracles, which defines the optimization problem in the cutting plane method described in Subsection 2.3.1, will be discussed in this section in detail.

2.4.1 Stability: Positivity constraint

From Lemma 2.2.3 it can be seen that the positivity constraint $\tilde{a}(\omega) > 0$ in program (2.14) is equivalent to the stability constraint in program (2.4) requiring $q(z)$ to be a Schur polynomial. Therefore, the positivity constraint must be strictly imposed for all ω ranging from 0 to 2π, and therefore the common engineering practice of enforcing such constraint on only a finite set of points in that interval will not suffice. In order to address this issue consider the positivity constraint (for convenience, assuming $\tilde{a}_m \neq 0$)

$$\tilde{a}(\omega) = 1 + a_1 \cos(\omega) + \ldots + a_m \cos(m\omega) > 0, \forall\, \omega \in [0, 2\pi). \tag{2.27}$$

It is sufficient (because $\tilde{a}(\omega)$ is an even function of ω) to check whether

$$\min_{\omega \in [0,\pi]} \tilde{a}(\omega) > 0.$$

Since $\tilde{a}(\omega)$ is continuous over $[0, \pi]$, the minimum is attained, and it can only be at the roots of

$$\frac{d\tilde{a}(\omega)}{d\omega} = -\tilde{a}_1 \sin(\omega) - \ldots - m\tilde{a}_m \sin(m\omega) = 0, \tag{2.28}$$

as the boundary points are included with

$$\frac{d\tilde{a}(0)}{d\omega} = \frac{d\tilde{a}(\pi)}{d\omega} = 0.$$

If there exists ω_0 among the roots of (2.28) s.t. $\tilde{a}(\omega_0) \leq 0$, then $\tilde{a}(\omega_0) > 0$ defines a cut, otherwise the positivity constraint is met.

In order to find the roots of (2.28), the identity $z = e^{j\omega} = \cos(\omega) + j\sin(\omega)$ can be applied to (2.28):

$$\begin{aligned} \frac{d\tilde{a}(\omega)}{d\omega} &= -\frac{1}{2j}\left(a_1\left(z - z^{-1}\right) + \ldots + m a_m \left(z^m - z^{-m}\right)\right) \\ &:= \partial\tilde{a}(z) \\ &= 0 \end{aligned}$$

Note that $z^m \partial\tilde{a}(z)$ is an ordinary polynomial of degree $2m$ and $e^{j\omega} \neq 0, \forall\, \omega \in \mathbb{R}$. Therefore, any ω_0 is a root of (2.28) if and only if it is a root of $\partial\tilde{a}(e^{j\omega})$ and the root finding task can be performed by finding (unit circle) roots of an ordinary polynomial $z^m \partial\tilde{a}(z)$ of degree $2m$.

2.4.2 Passivity for impedance systems: Positive real constraint

For some applications it is desirable that the reduced model transfer function has positive real part. In order to impose this constraint, it suffices to note that the real part of the relaxed transfer function in program (2.14) is $\tilde{b}(\omega)/\tilde{a}(\omega)$. Therefore, the only modification to (2.14) is to add the constraint

$$\tilde{b}(\omega) > 0, \quad \forall\, \omega \in [0, 2\pi)$$

and the treatment of this oracle is similar to that of the positivity constraint discussed in Subsection 2.4.1 because $\tilde{a}(\omega)$ and $\tilde{b}(\omega)$ are the same type of trigonometric polynomials.

However, it should be noted that program (2.26) should be modified accordingly to guarantee the positive realness of the final reduced model. That is, the following constraint should be added.

$$p(e^{j\omega})q(e^{-j\omega}) + p(e^{-j\omega})q(e^{j\omega}) > 0, \forall\, \omega \in [0, 2\pi)\,. \tag{2.29}$$

It is important to realize that the left side of constraint (2.29) is a trigonometric polynomial (with respect to ω) whose coefficients are linear functions of the decision variables $p(z)$.

2.4.3 Passivity for S-parameter systems: Bounded real constraint

For S-parameter models, the notion of dissipative system is given by the bounded real condition (i.e. $|H(z)| < 1, \forall z \in \mathbb{C}, |z| = 1$). To model this property, program (2.14) can be modified by adding the constraint

$$\tilde{a}(\omega) > |\,\tilde{b}(\omega) + j\,\tilde{c}(\omega)|, \quad \forall\, \omega \in [0, 2\pi)\,. \tag{2.30}$$

To construct the oracle, first check the positivity of the trigonometric polynomial

$$\tilde{a}(\omega)^2 - \tilde{b}(\omega)^2 - \tilde{c}(\omega)^2 > 0, \quad \forall\, \omega \in [0, 2\pi)\,.$$

If this condition is met, then bounded realness is satisfied at the current query point, otherwise there exists some $\omega_0 \in [0, 2\pi)$ at which the bounded real constraint in eq. (2.30) is violated. Then the constraint

$$\tilde{a}(\omega_0) > |\tilde{b}(\omega_0) + j\,\tilde{c}(\omega_0)|$$

defines a desired cut. It is noted that program (2.26) should be modified analogously to preserve the passivity of the final reduced model.

2.4.4 Multi-port positive real passivity

For a multi-port transfer matrix $H(z) \in \mathbb{C}^{n\times n}$ with real coefficients, positive real passivity means

$$H(e^{j\omega}) + H(e^{j\omega})' > 0, \quad \forall\, \omega \in [0, 2\pi)\,, \tag{2.31}$$

with $'$ denoting complex conjugate transpose of a matrix and the inequality in eq. (2.31) means that the matrix sum in the LHS has real and positive eigenvalues. Define the following notations.

Let

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k] + Du[k] \end{aligned} \tag{2.32}$$

be a state-space realization of $H(z)$ and define the 2×2 block matrix

$$\Sigma := \begin{bmatrix} 0 & C' \\ C & D + D' \end{bmatrix} := \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}. \tag{2.33}$$

The following generalized eigenvalue problem will be considered later.

$$z\begin{bmatrix} -\Sigma_{11}+\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} & A'+\Sigma_{12}\Sigma_{22}^{-1}B' \\ -I & 0 \end{bmatrix} - \begin{bmatrix} 0 & I \\ -A+B\Sigma_{22}^{-1}\Sigma_{21} & -B\Sigma_{22}^{-1}B' \end{bmatrix} = 0 \tag{2.34}$$

The following lemma describes the oracle construction procedure.

Lemma 2.4.1. *Assume $\Sigma_{22} > 0$. If generalized eigenvalue problem (2.34) does not have any eigenvalue on the unit circle, then (2.31) is satisfied. Otherwise, there exists $\omega_0 \in [0, 2\pi)$ such that $e^{j\omega_0}$ is an eigenvalue of (2.34), and $H(e^{j\omega_0}) + H(e^{j\omega_0})' \not> 0$. In this case if $v_0 \in \mathbb{C}^n$ is an eigenvector associated with a non-positive eigenvalue of $H(e^{j\omega_0}) + H(e^{j\omega_0})'$, then*

$$v_0{}'(H(e^{j\omega_0}) + H(e^{j\omega_0})')v_0 > 0 \tag{2.35}$$

defines a (real coefficient) linear cut with respect to the coefficients of the numerator of H. ■

Proof of Lemma 2.4.1. Note that (2.31) is the same as

$$u'H(e^{j\omega})u + u'H(e^{j\omega})'u > 0, \quad \forall u \in \mathbb{C}^n, u \neq 0, \omega \in [0, 2\pi), \tag{2.36}$$

and it is equivalent to (with Σ as defined in (2.33))

$$\begin{bmatrix} x \\ u \end{bmatrix} \Sigma \begin{bmatrix} x \\ u \end{bmatrix}' > 0, \tag{2.37}$$

subject to "system constraints"

$$zx = Ax + Bu \tag{2.38}$$

and $Hu = Cx + Du$ for $z \in \mathbb{C}$. According to KYP lemma [68], frequency dependent inequality (2.37) subject to "system constraint" (2.38) holds if and only if the system of equations

(with unknowns x, u and ψ)

$$\begin{aligned} zx &= Ax + Bu \\ \tfrac{\psi}{z} &= A'\psi - \Sigma_{11}x - \Sigma_{12}u \\ B'\psi &= \Sigma_{21}x + \Sigma_{22}u, \end{aligned} \tag{2.39}$$

does not have any nonzero solution for $|z| = 1$. Since Σ_{22} is assumed to be invertible, solving for u from the last equation of (2.39), it can be seen that the conditions in eq. (2.39) is equivalent to the condition that the generalized eigenvalue problem in eq. (2.34) does not have any eigenvalue on the unit circle. Therefore, if this condition is true, then condition (2.31) is met. Otherwise, let $e^{j\omega_0}$ be an eigenvalue of problem (2.34) and it needs to be shown that

$$H(e^{j\omega_0}) + H(e^{j\omega_0})' \not> 0. \tag{2.40}$$

Indeed, $e^{j\omega_0}$ being an eigenvalue of (2.34) implies that (2.39) is satisfied with $e^{j\omega_0}$ and the corresponding x, u and ψ, then quadratic form from (2.37) becomes

$$\begin{aligned} & x'\Sigma_{11}x + x'\Sigma_{12}u + u'\Sigma_{21}x + u'\Sigma_{22}u \\ = \;& x'\left(\Sigma_{11}x + \Sigma_{12}u\right) + u'\left(\Sigma_{21}x + \Sigma_{22}u\right) \\ = \;& x'\left(A'\psi - e^{-j\omega_0}\right) + u'B'\psi \\ = \;& \left(Ax + Bu - e^{j\omega}x\right)'\psi \\ = \;& 0 \end{aligned}$$

and (2.40) is resulted. In the derivation, the second and the fourth equalities are due to (2.39). The fact that (2.35) defines a linear cut should be obvious. ■

Remark 2.4.2. It should be noted that the assumption $\Sigma_{22} > 0$ is in fact necessary for positive real passivity condition eq. (2.31) to hold. This is because

$$H(e^{j\omega}) + H(e^{j\omega})' = C\left(e^{j\omega}I - A\right)^{-1}B + \left(C\left(e^{j\omega}I - A\right)^{-1}B\right)' + D + D', \tag{2.41}$$

where A, B, C, D are the state space matrices defined in eq. (2.32). Integrating eq. (2.41)

with respect to ω results in

$$\int_0^{2\pi} \left(C\left(e^{j\omega}I - A\right)^{-1} B + \left(C\left(e^{j\omega}I - A\right)^{-1} B \right)' + D + D' \right) d\omega = 2\pi \left(D + D'\right) = 2\pi\Sigma_{22} \tag{2.42}$$

as the first two terms in the integrand integrate to zero. Therefore, if eq. (2.41) is to be positive definite for all values of ω, then it integral (2.42) should also be positive definite, meaning that $\Sigma_{22} > 0$ is necessary for eq. (2.31) to hold. ■

2.4.5 Objective oracle

In the case where the transfer function H of the original system is fully specified explicitly (in terms of system matrices, numerator/denominator, or pole/zero/gain), and the exact $\mathcal{H}_\infty$ norm is to be minimized, one can use the following oracle: given the current iterates $(\tilde{a}, \tilde{b}, \tilde{c})$ and the desired level of optimality γ, an unstable transfer function

$$\hat{H}(e^{j\omega}) := \frac{\tilde{b}(\omega) + j\tilde{c}(\omega)}{\tilde{a}(\omega)}$$

can be realized. Then the difference system $H - \hat{H}$ can be formed to check if its $\mathcal{L}_\infty$ norm (same definition as $\mathcal{H}_\infty$ norm defined in eq. (2.1) and eq. (2.2), but not limited to stable systems) is less than γ. If the corresponding $\mathcal{L}_\infty$ norm is not smaller than γ, then a violating frequency ω_0 can be identified and the cut

$$|\tilde{b}(\omega_0) + j\tilde{c}(\omega_0) - \tilde{a}(\omega_0)H(\omega_0)| < \gamma\tilde{a}(\omega_0)$$

can be enforced.

In the case where the transfer function H of the original system is specified as sample data $(\omega_i, H(\omega_i)), \quad i = 1, 2, \ldots N$, the $\mathcal{L}_\infty$ norm check of the difference $H - \hat{H}$ can be simplified to checking N inequalities.

Finally, if the original transfer function H is again given explicitly (e.g., system matrices), but the $\mathcal{L}_\infty$ norm oracle mentioned above is deemed too expensive to compute, the frequency response of H can be sampled, and the proposed algorithm still applies (al-

though the $\mathcal{H}_\infty$ norm error is no longer guaranteed). Uniform sampling of the discrete-time frequency axis over the range of interest is generally a good choice for the proposed algorithm.

2.5 Extension to PMOR

This section discusses how the setup in (2.14) can be extended to solve the problem of the parameterized model order reduction.

2.5.1 Optimal $\mathcal{H}_\infty$ norm parameterized model order reduction problem and relaxation

The parameterized model order reduction problem is defined as follows: given a stable transfer function $H(z,\mathbf{p})$, where $\mathbf{p}$ is the vector of design parameters contained in a set $\mathcal{P} \subset \mathbb{R}^{n_\mathbf{p}}$, and a positive integer m (as the order of the reduced model), construct a stable parameterized rational transfer function with real coefficient functions

$$\hat{H}(z,\mathbf{p}) = \frac{p(z,\mathbf{p})}{q(z,\mathbf{p})} := \frac{p_m(\mathbf{p})z^m + p_{m-1}(\mathbf{p})z^{m-1} + \ldots + p_0(\mathbf{p})}{z^m + q_{m-1}(\mathbf{p})z^{m-1} + \ldots + q_0(\mathbf{p})}, \quad p_k, q_k : \mathbb{R}^{n_\mathbf{p}} \mapsto \mathbb{R}, \forall k$$

such that $\hat{H}(z,\mathbf{p})$ is the optimal solution of

$$\begin{aligned} \underset{p,q}{\text{minimize}} \quad & \max_{\mathbf{p}\in\mathcal{P}} \left\| H(z,\mathbf{p}) - \tfrac{p(z,\mathbf{p})}{q(z,\mathbf{p})} \right\|_\infty \\ \text{subject to} \quad & \deg(q) = m, \quad \deg(p) \le m, \\ & q(z,\mathbf{p}) \neq 0, \quad \forall z \in \mathbb{C}, |z| \ge 1, \forall \mathbf{p} \in \mathcal{P} \quad \text{(stability)}. \end{aligned} \tag{2.43}$$

Parallel to the development in the non-parameterized case in Section 2.2, quasi-convex program (2.14) is extended by introducing the following *parameterized univariate trigono-*

metric polynomials with real coefficients

$$\begin{aligned} a(z,\mathbf{p}) &= a_0(\mathbf{p}) + a_1(\mathbf{p})(z+z^{-1}) + \ldots + a_m(\mathbf{p})(z^m + z^{-m}), \\ b(z,\mathbf{p}) &= b_0(\mathbf{p}) + b_1(\mathbf{p})(z+z^{-1}) + \ldots + b_m(\mathbf{p})(z^m + z^{-m}), \\ c(z,\mathbf{p}) &= \tfrac{1}{j}\big(c_1(\mathbf{p})(z-z^{-1}) + \ldots + c_m(\mathbf{p})(z^m - z^{-m})\big). \end{aligned} \tag{2.44}$$

Then the parameterized version of program (2.14) becomes

$$\begin{aligned} \underset{\tilde{a},\tilde{b},\tilde{c},\gamma}{\text{minimize}} \quad & \gamma \\ \text{subject to} \quad & |H(e^{j\omega},\mathbf{p})\tilde{a}(\omega,\mathbf{p}) - \tilde{b}(\omega,\mathbf{p}) - j\tilde{c}(\omega,\mathbf{p})| < \gamma\tilde{a}(\omega,\mathbf{p}), \forall\,\omega \in [0,2\pi), \quad \forall\,\mathbf{p} \in \mathcal{P}, \\ & \tilde{a}(\omega,\mathbf{p}) > 0, \quad \forall\,\omega \in [0,2\pi), \forall\,\mathbf{p} \in \mathcal{P} \\ & \deg(\tilde{a}) \le m, \quad \deg(\tilde{b}) \le m, \quad \deg(\tilde{c}) \le m. \end{aligned} \tag{2.45}$$

Here the decision variables are γ, and the coefficients of $\tilde{a},\tilde{b},\tilde{c}$ as functions of the design parameter vector $\mathbf{p}$. By the same argument as in the proof of Lemma 2.2.6 in Subsection 2.2.2, program (2.45) can be shown to be quasi-convex. However, as it turns out, program (2.45) is difficult to solve. The subsequent part of this section will focus on approximately solving program (2.45) using the cutting method. The emphasis will be given to the construction of the parameterized stability oracle, as it is the main roadblock to the solution.

2.5.2 PMOR stability oracle – challenge and solution idea

2.5.2 A: PMOR stability check problem

In practice, in program (2.45) the frequency response matching constraint (i.e., the first set of the constraints) is enforced only at some finite number of frequencies and parameter values, and hence it can be handled by the same procedure for the non-parameterized case described in Subsection 2.4.5. The stability constraint (i.e., the second set of constraints in (2.45)), however, has to be enforced for all values of frequencies as well as design parameters. In the context of a solution procedure via the cutting plane method, constraint enforcement amounts to the following check in program (2.45): given functions

$a_0(\mathbf{p}), a_1(\mathbf{p}), \ldots, a_m(\mathbf{p})$, check if it is true that

$$\tilde{a}(\omega, \mathbf{p}) > 0, \quad \forall \omega, \forall \mathbf{p} \in \mathcal{P}, \tag{2.46}$$

2.5.2 B: Polynomially parameterized univariate trigonometric polynomial

In general, it is very difficult to solve the problem in eq. (2.46) if $a_0(\mathbf{p}), a_1(\mathbf{p}), \ldots, a_m(\mathbf{p})$ are arbitrary functions of $\mathbf{p}$. Therefore, the first step to solve the stability check challenge in eq. (2.46) is proposed in this thesis that these functions are restricted to be polynomials. Define (as the degree of $\tilde{a}(\omega, \mathbf{p})$)

$$\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}, \quad \mathbf{m} := \begin{bmatrix} \mathbf{m}_0 & \mathbf{m}_1 & \cdots & \mathbf{m}_{n_\mathbf{p}} \end{bmatrix}'$$

with $\mathbf{m}_0$ taking the place of m in program (2.45). Then

Definition 2.5.1. *A polynomially parameterized univariate trigonometric polynomial of degree* $\mathbf{m}$*, associated with* $\tilde{a}(\omega, \mathbf{p})$ *in eq. (2.46), is defined as* $\tilde{a} : [0, 2\pi) \times \mathcal{P} \mapsto \mathbb{R}$ *:*

$$\begin{aligned} \tilde{a}(\omega, \mathbf{p}) &= \sum_{\mathbf{i}_0=0}^{\mathbf{m}_0} \sum_{\mathbf{i}_1=0}^{\mathbf{m}_1} \cdots \sum_{\mathbf{i}_{n_\mathbf{p}}=0}^{\mathbf{m}_{n_\mathbf{p}}} \tilde{a}_{\mathbf{i}_0, \mathbf{i}_1, \ldots, \mathbf{i}_{n_\mathbf{p}}} \left(\mathbf{p}_1^{\mathbf{i}_1} \ldots \mathbf{p}_{n_\mathbf{p}}^{\mathbf{i}_{n_\mathbf{p}}} \right) \cos(\mathbf{i}_0 \omega) \\ &:= \sum_{\mathbf{i}=\mathbf{0}}^{\mathbf{m}} \tilde{a}_\mathbf{i} \mathbf{p}^{\mathbf{i}_{1:}} \cos(\mathbf{i}_0 \omega) \end{aligned} \tag{2.47}$$

with

$$\mathbf{i} \in \mathbb{Z}_+^{n_\mathbf{p}+1}, \quad \mathbf{i} := \begin{bmatrix} \mathbf{i}_0 & \mathbf{i}_1 & \cdots & \mathbf{i}_{n_\mathbf{p}} \end{bmatrix}',$$

and

$$\mathbf{p}^{\mathbf{i}_{1:}} := \mathbf{p}_1^{\mathbf{i}_1} \ldots \mathbf{p}_{n_\mathbf{p}}^{\mathbf{i}_{n_\mathbf{p}}},$$

and

$$\tilde{a}_\mathbf{i} \in \mathbb{R}, \quad \forall \mathbf{0} \leq \mathbf{i} \leq \mathbf{m},$$

with inequalities understood entry-wise. ■

Accordingly, the stability constraint in eq. (2.46) becomes

$$\tilde{a}(\omega, \mathbf{p}) = \sum_{\mathbf{i}=\mathbf{0}}^{\mathbf{m}} \tilde{a}_{\mathbf{i}} \mathbf{p}^{\mathbf{i}_{1:}} \cos(\mathbf{i}_0 \omega) > 0, \quad \forall \omega, \forall \mathbf{p}. \tag{2.48}$$

Unfortunately, even though constraint eq. (2.48) is linear (hence convex) with respect to the decision variables (i.e., coefficients $\tilde{a}_{\mathbf{i}}$), there is no known efficient algorithms to check whether it is satisfied or not. It will be clear that this difficulty is resulted from the fact that the set of positive multivariate trigonometric polynomials cannot be characterized in the same computationally tractable manner as in the univariate case. In addition, looking back at the non-parameterized stability oracle procedure described in Subsection 2.4.1 would provide some insight into why the parameterized case is more difficult. It was shown in Subsection 2.4.1 that the positivity check can be done by finding the roots of some univariate polynomial. However, for the parameterized case, the checking of constraint eq. (2.48) would analogously be finding the (infinitely many) roots of a *multivariate* polynomial. There is no efficient algorithm for such a problem.

2.5.2 C: Conversion to multivariate trigonometric polynomials

The next step to solve the challenge in eq. (2.48) is to transform the polynomially parameterized univariate trigonometric polynomial $\tilde{a}(\omega, \mathbf{p})$ in eq. (2.48) to a *multivariate* trigonometric polynomial. This transformation will be detailed in Subsection 2.5.3.

2.5.2 D: Sum-of-squares relaxation solution idea – overview

The benefit of transforming $\tilde{a}(\omega, \mathbf{p})$ in eq. (2.48) to a (to be defined) multivariate trigonometric polynomial is that it allows the use of sum-of-squares (SOS) relaxation. The main idea is that instead of checking the positivity of a multivariate trigonometric polynomial, it would be much more computationally tractable to check the SOS condition (to be defined in Subsection 2.5.4). In addition, it will also be shown that the relationship between the set of SOS and the set of positivity trigonometric polynomials are closely related, hence justifying the use of SOS. However, it should be forewarned that the SOS approach is not without its own limitations, which will further be explained in Subsection 2.5.4. Finally,

the parameterized stability oracle, based on the SOS relaxation idea, will be described in Subsection 2.5.5.

2.5.3 From polynomially parameterized univariate trigonometric polynomial to multivariate trigonometric polynomial

In a sense, $\tilde{a}(\omega, \mathbf{p})$ in eq. (2.48) is a "mixed" polynomial – if $\mathbf{p}$ is fixed, then $\tilde{a}$ is a trigonometric polynomial of ω. On the other hand, if ω is fixed, then $\tilde{a}$ is an ordinary polynomial of $\mathbf{p}$. There are SOS tools working with ordinary polynomials or trigonometric polynomials, but there is none for both. The solution strategy adopted by this thesis is to convert eq. (2.48) into a multivariate trigonometric polynomial positivity constraint. This adoption is for numerical robustness and convenience. A parallel procedure of working with ordinary polynomials is entirely possible. The development for the rest of this subsection will be divided into two parts. First, the multivariate trigonometric polynomial will formally be defined. Then the conversion bearing the title of this subsection will be detailed.

2.5.3 A: Multivariate trigonometric polynomials

We first recall that $n_{\mathbf{p}} \in \mathbb{N}$ is the number of design parameters, and the default dimension of many vector spaces to be discussed will be $n_{\mathbf{p}}+1$.

Definition 2.5.2. *A halfspace $\tilde{\mathcal{H}} \subset \mathbb{Z}^{n_{\mathbf{p}}+1}$ is a set such that $\tilde{\mathcal{H}} \cap \left(-\tilde{\mathcal{H}}\right) = \{0\}$, $\tilde{\mathcal{H}} \cup \left(-\tilde{\mathcal{H}}\right) = \mathbb{Z}^{n_{\mathbf{p}}+1}$, and $\tilde{\mathcal{H}} + \tilde{\mathcal{H}} \subset \tilde{\mathcal{H}}$ (i.e., closed under addition).* ■

To explicitly denote the dimension of a halfspace, $\tilde{\mathcal{H}}$ can be written as $\tilde{\mathcal{H}}_d \subset \mathbb{Z}^d$ for any $d \in \mathbb{N}$ with the default value of d as $n_{\mathbf{p}}+1$. It can be verified that the following procedure defines a halfspace $\check{\mathcal{H}}_d \subset \mathbb{Z}^d$. It is defined that $\mathbf{k} \in \check{\mathcal{H}}_d$ if one of the following is true

1. $\mathbf{k}_{d-1} > 0$,
2. $\mathbf{k}_{d-1} = 0$ and $(\mathbf{k}_0, \ldots, \mathbf{k}_{d-2}) \in \check{\mathcal{H}}_{d-1}$,

with $\check{\mathcal{H}}_1 := \{0, 1, 2, \ldots\}$. The symbol $\mathcal{H}$ will be reserved for the halfspace thus constructed in $\mathbb{Z}^{n_{\mathbf{p}}+1}$. That is,

$$\mathcal{H} := \check{\mathcal{H}}_{n_{\mathbf{p}}+1}. \tag{2.49}$$

Notation. For any $\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$, $\mathbf{B}_\mathbf{m} \subset \mathbb{Z}^{n_\mathbf{p}+1}$ is defined as

$$\mathbf{B}_\mathbf{m} = \left\{ \mathbf{k} \in \mathbb{Z}^{n_\mathbf{p}+1} | -\mathbf{m} \leq \mathbf{k} \leq \mathbf{m} \right\}. \tag{2.50}$$

Here the inequalities are understood entry-wise (i.e., $|\mathbf{k}_i| \leq \mathbf{m}_i, \forall\, i = 0, \ldots, n_\mathbf{p}$). ■

Notation. Denote

$$\mathbf{z} := \begin{bmatrix} z_0 & z_1 & \cdots & z_{n_\mathbf{p}} \end{bmatrix}^T \in \mathbb{C}^{n_\mathbf{p}+1}, \tag{2.51}$$

and

$$\mathbf{k} := \begin{bmatrix} k_0 & k_1 & \cdots & k_{n_\mathbf{p}} \end{bmatrix}' \in \mathbb{Z}^{n_\mathbf{p}+1}. \tag{2.52}$$

Then the "multivariate power" is defined as

$$\mathbf{z}^\mathbf{k} := z_0^{k_0} {z_1}^{k_1} \cdots {z_{n_\mathbf{p}}}^{k_{n_\mathbf{p}}}. \tag{2.53}$$

■

Definition 2.5.3. *A multivariate trigonometric polynomial of degree* $\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$ *is defined as a function* $a(\mathbf{z}) : \mathbb{C}^{n_\mathbf{p}+1} \mapsto \mathbb{C}$ *such that*

$$a(\mathbf{z}) := \sum_\mathbf{k} a_\mathbf{k} \left(\mathbf{z}^\mathbf{k} + \mathbf{z}^{-\mathbf{k}} \right), \quad \mathbf{k} \in \mathcal{H} \cap \mathbf{B}_\mathbf{m}, \quad a_\mathbf{k} \in \mathbb{R}, \forall \mathbf{k}, \tag{2.54}$$

where $\mathbf{z}^\mathbf{k}$, $\mathcal{H}$ *and* $\mathbf{B}_\mathbf{m}$ *are defined in eq. (2.53), eq. (2.49) and (2.50), respectively.* ■

Define the $n_\mathbf{p}+1$ dimensional unit sphere as

$$\mathbb{T} := \left\{ \mathbf{z} \in \mathbb{C}^{n_\mathbf{p}+1} | \, |z_0| = |z_1| = \ldots = \left|z_{n_\mathbf{p}}\right| = 1 \right\}. \tag{2.55}$$

Then it can be seen that

$$a(\mathbf{z}) \in \mathbb{R}, \quad \forall \mathbf{z} \in \mathbb{T}$$

because for all $\mathbf{k}$,

$$\frac{1}{2}\left(\mathbf{z}^{\mathbf{k}}+\mathbf{z}^{-\mathbf{k}}\right)=\cos\left(\mathbf{k}'\omega\right) \quad \text{with} \quad \omega:=-j\left[\log\left(\mathbf{z}_0\right) \quad \log\left(\mathbf{z}_1\right) \quad \cdots \quad \log\left(\mathbf{z}_{n_{\mathbf{p}}}\right)\right]^T \in \mathbb{R}^{n_{\mathbf{p}}+1}, \tag{2.56}$$

which gives rise to the name "trigonometric polynomial".

Definition 2.5.4. *A trigonometric polynomial is said to be positive if it is positive on the unit sphere.*

$$a\left(\mathbf{z}\right)>0, \quad \forall \mathbf{z}\in\mathbb{T}, \tag{2.57}$$

and a trigonometric polynomial is said to be nonnegative if it is nonnegative on the unit sphere.

$$a\left(\mathbf{z}\right)\geq 0, \quad \forall \mathbf{z}\in\mathbb{T}, \tag{2.58}$$

where $\mathbb{T}$ is defined in eq. (2.55). ■

2.5.3 B: The conversion

The first step towards the conversion is to re-define the indeterminates $(\omega,\mathbf{p})$ in $\tilde{a}\left(\omega,\mathbf{p}\right)$ in eq. (2.47). This is achieved with an additional assumption, which will remain throughout the chapter.

Assumption. *It is assumed that $\mathcal{P}$ is a bounded set. That is, there exist $\underline{\mathbf{p}}\in\mathbb{R}^{n_{\mathbf{p}}}$ and $\overline{\mathbf{p}}\in\mathbb{R}^{n_{\mathbf{p}}}$ such that*

$$\mathcal{P}=\left\{\mathbf{p}\in\mathbb{R}^{n_{\mathbf{p}}}\,\middle|\,\underline{\mathbf{p}}_i\leq\mathbf{p}_i\leq\overline{\mathbf{p}}_i,\,\forall i=1,2,\ldots,n_{\mathbf{p}}\right\}. \tag{2.59}$$

■

Denote $\mathbf{z}$ as in eq. (2.51) as a new set of indeterminates that will be used in eq. (2.47), and recall the definition of the unit sphere $\mathbb{T}$ in eq. (2.55). Then following lemma defines an one-to-one correspondence between the sets $[0,2\pi)\times\mathcal{P}$ and $\mathbb{T}$ (corresponding to variables $(\omega,\mathbf{p})$ and $\mathbf{z}$).

Lemma 2.5.5. *The function* $f : \mathbb{T} \mapsto [0, 2\pi) \times \mathcal{P}, f(\mathbf{z}) = (\omega, \mathbf{p})$ *is one-to-one, when it is defined as*

$$f(\mathbf{z}) := \begin{bmatrix} f_0(\mathbf{z}_0) \\ f_1(\mathbf{z}_1) \\ \vdots \\ f_{n_\mathbf{p}}(\mathbf{z}_{n_\mathbf{p}}) \end{bmatrix} = \begin{bmatrix} \omega \\ \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_{n_\mathbf{p}} \end{bmatrix}$$

with

$$\begin{aligned} f_0(\mathbf{z}_0) &= -j \log(\mathbf{z}_0) \\ f_i(\mathbf{z}_i) &= \tfrac{\overline{\mathbf{p}}_i + \underline{\mathbf{p}}_i}{2} + \left(\tfrac{\overline{\mathbf{p}}_i - \underline{\mathbf{p}}_i}{4}\right)\left(\mathbf{z}_i + \mathbf{z}_i^{-1}\right), \quad \forall i = 1, 2, \ldots, n_\mathbf{p} \\ &= \tfrac{\overline{\mathbf{p}}_i + \underline{\mathbf{p}}_i}{2} + \left(\tfrac{\overline{\mathbf{p}}_i - \underline{\mathbf{p}}_i}{2}\right)\cos(-j \log(\mathbf{z}_i)). \end{aligned} \tag{2.60}$$

■

Proof of Lemma 2.5.5. First, by inspection, $[0, 2\pi) \times \mathcal{P} = f(\mathbb{T})$, which shows that f is surjective. Then, since $\log(\cdot)$ and $\cos(\cdot)$ are injective on their respective domains (i.e., $\mathbb{T}$ and $[0, 2\pi)$), f is injective. Therefore, f is one-to-one. ■

The fact that f is one-to-one means that the positivity check in eq. (2.48) is the same as the check of

$$\tilde{a}(f(\mathbf{z})) > 0, \quad \forall \mathbf{z} \in \mathbb{T}. \tag{2.61}$$

The real benefit of introducing f in eq. (2.60), though, is that $\tilde{a}(f(\mathbf{z}))$ is a multivariate trigonometric polynomial, as stated by the following lemma.

Lemma 2.5.6. *Let* $\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$ *and* $\tilde{a}(\omega, \mathbf{p})$ *be a polynomially parameterized univariate trigonometric polynomial of degree* $\mathbf{m}$*, defined as in eq. (2.47). Let* $f(\mathbf{z}) = (\omega, \mathbf{p})$ *be the change of indeterminates defined as in Lemma 2.5.5. Define* $a(\mathbf{z}) := \tilde{a}(f(\mathbf{z}))$*, then it is a multivariate trigonometric polynomial (with respect to* $\mathbf{z}$*) of degree* $\mathbf{m}$*. That is,* $a(\mathbf{z})$ *has the form in eq. (2.54)* ■

Proof of Lemma 2.5.6. **Step 1** is to show that the set of (degree unspecified) trigonometric

polynomials is closed under addition, scalar multiplication and multiplication. Let

$$\begin{aligned} b(\mathbf{z}) &:= \sum_{\mathbf{k}} b_{\mathbf{k}} \left(\mathbf{z}^{\mathbf{k}} + \mathbf{z}^{-\mathbf{k}}\right), \quad \mathbf{k} \in \mathcal{H}, \quad b_{\mathbf{k}} \in \mathbb{R}, \forall \mathbf{k} \\ c(\mathbf{z}) &:= \sum_{\mathbf{i}} c_{\mathbf{i}} \left(\mathbf{z}^{\mathbf{i}} + \mathbf{z}^{-\mathbf{i}}\right), \quad \mathbf{i} \in \mathcal{H}, \quad c_{\mathbf{i}} \in \mathbb{R}, \forall \mathbf{i} \end{aligned}$$

be two (degree unspecified) trigonometric polynomials. Then $(b+c)(\mathbf{z}) := b(\mathbf{z}) + c(\mathbf{z})$ and $(\alpha b)(\mathbf{z}) := \alpha b(\mathbf{z})$ are trigonometric polynomials by inspection. Furthermore, since

$$b(\mathbf{z})\, c(\mathbf{z}) = \sum_{\mathbf{k}} \sum_{\mathbf{i}} b_{\mathbf{k}} c_{\mathbf{i}} \left(\mathbf{z}^{\mathbf{k}} + \mathbf{z}^{-\mathbf{k}}\right) \left(\mathbf{z}^{\mathbf{i}} + \mathbf{z}^{-\mathbf{i}}\right).$$

The fact that

$$\left(\mathbf{z}^{\mathbf{k}} + \mathbf{z}^{-\mathbf{k}}\right) \left(\mathbf{z}^{\mathbf{i}} + \mathbf{z}^{-\mathbf{i}}\right) = \mathbf{z}^{\mathbf{k}+\mathbf{i}} + \mathbf{z}^{-(\mathbf{k}+\mathbf{i})} + \mathbf{z}^{\mathbf{k}-\mathbf{i}} + \mathbf{z}^{-\mathbf{k}+\mathbf{i}}, \quad \forall \mathbf{k}, \mathbf{i} \in \mathcal{H}$$

is a trigonometric polynomial shows that the product $b(\mathbf{z})\, c(\mathbf{z})$ is a trigonometric polynomial. Hence **step 1** is shown. **Step 1**, in particular, implies that a polynomial of trigonometric polynomials is still a trigonometric polynomial.

Step 2 of the proof is to recognize that $a(\mathbf{z}) = \tilde{a}(f(\mathbf{z}))$ as in the statement of the Lemma is indeed a polynomial of trigonometric polynomials with respect to $\mathbf{z}$. Applying eq. (2.60) to $a(\omega, \mathbf{p})$ in eq. (2.47) yields

$$a(f(\mathbf{z})) = \frac{1}{2} \sum_{\mathbf{i}=0}^{\mathbf{m}} \tilde{a}_{\mathbf{i}} \prod_{t=1}^{n_{\mathbf{p}}} \left(\frac{\overline{\mathbf{p}}_t + \underline{\mathbf{p}}_t}{2} + \left(\frac{\overline{\mathbf{p}}_t - \underline{\mathbf{p}}_t}{4} \right) \left(\mathbf{z}_t + \mathbf{z}_t^{-1}\right) \right)^{\mathbf{i}_t} \left(\mathbf{z}_0^{\mathbf{i}_0} + \mathbf{z}_0^{-\mathbf{i}_0}\right). \tag{2.62}$$

It is then to recognize that

$$\mathbf{z}_t = \mathbf{z}^{\delta_t},$$

with δ_t having only a single non-zero value of 1 in the t^{th} entry. Therefore, factors in eq. (2.62) such as

$$\frac{\overline{\mathbf{p}}_t + \underline{\mathbf{p}}_t}{2} + \left(\frac{\overline{\mathbf{p}}_t - \underline{\mathbf{p}}_t}{4} \right) \left(\mathbf{z}_t + \mathbf{z}_t^{-1}\right)$$

and

$$\mathbf{z}_0^{\mathbf{i}_0} + \mathbf{z}_0^{-\mathbf{i}_0}$$

are trigonometric polynomials of $\mathbf{z}$, and consequently by **step 1**, eq. (2.62) is a polynomial of trigonometric polynomial of $\mathbf{z}$ with the form

$$a(\mathbf{z}) = \sum_{\mathbf{k}} a_{\mathbf{k}} \left(\mathbf{z}^{\mathbf{k}} + \mathbf{z}^{-\mathbf{k}} \right), \quad \mathbf{k} \in \mathcal{H}, \quad a_{\mathbf{k}} \in \mathbb{R}, \forall \mathbf{k} \tag{2.63}$$

Finally, **step 3** of the proof is to verify that the degree of eq. (2.63) is indeed $\mathbf{m}$. This can be shown simply by checking the monomials in eq. (2.62). ■

Remark 2.5.7. Lemma 2.5.5 asserts that the parameterized stability check can be performed, equivalently, by the positivity check in eq. (2.48) and eq. (2.61). Both are equally hard, but Lemma 2.5.6 states that the latter is a positivity check of a multivariate trigonometric polynomial, which can be checked in a restricted sense by using the SOS relaxation idea to be described in Subsection 2.5.4. ■

Remark 2.5.8. In the conversion to eq. (2.54) given in Lemma 2.5.6, the coefficients $a_{\mathbf{k}}$ are not independent. This can be seen as follows: by the trigonometric identity

$$\cos(nx) = T_n(\cos(x)), \quad \forall x \in [0, 2\pi)$$

where $T_n(\cdot)$ is the Chebyshev polynomial of degree n, it can be seen that eq. (2.62) is actually an ordinary polynomial of the terms $\cos(-j\log(\mathbf{z}_i))$, whereas mixed terms such as $\cos(-j\log(\mathbf{z}_i))\sin(-j\log(\mathbf{z}_k))$ are allowed in eq. (2.54). For example,

$$\mathbf{z}_i\mathbf{z}_k + \mathbf{z}_i^{-1}\mathbf{z}_k^{-1} = 2\left(\cos(-j\log(\mathbf{z}_i))\cos(-j\log(\mathbf{z}_k)) - \sin(-j\log(\mathbf{z}_i))\sin(-j\log(\mathbf{z}_k))\right).$$

The "over-parameterizations" of $a(\mathbf{z})$ in eq. (2.54) when dealing with $\tilde{a}(\omega, \mathbf{p})$ in eq. (2.47) can also be seen by looking at the lengths of the respective vector of coefficients. Denote

Notation.

$$\tilde{\mathbf{a}} \in \mathbb{R}^{|\tilde{\mathbf{a}}|}, \ \tilde{\mathbf{a}} := \begin{bmatrix} \vdots \\ \tilde{a}_{\mathbf{k}} \\ \vdots \end{bmatrix}, \quad \mathbf{0} \leq \mathbf{k} \leq \mathbf{m}, \quad |\tilde{\mathbf{a}}| := \prod_{i=0}^{n_{\mathbf{p}}} (\mathbf{m}_i + 1) \tag{2.64}$$

$$\mathbf{a} \in \mathbb{R}^{|\mathbf{a}|},\ \mathbf{a} := \begin{bmatrix} \vdots \\ a_{\mathbf{k}} \\ \vdots \end{bmatrix}, \quad \mathbf{k} \in \mathcal{H} \cap \mathbf{B}_{\mathbf{m}}, \quad |\mathbf{a}| := \frac{1}{2}\left(\prod_{i=0}^{n_{\mathbf{p}}} (2\mathbf{m}_i + 1) + 1\right). \tag{2.65}$$

Here $\tilde{a}_{\mathbf{k}}$ and $a_{\mathbf{k}}$ are coefficients of the trigonometric polynomials in eq. (2.47) and eq. (2.54), respectively. ■

Then it is generally true that $|\mathbf{a}| > |\tilde{\mathbf{a}}|$. The observation of the coefficient redundancy in the general multivariate trigonometric polynomial representation might lead to a speedup in the implementation of the parameterized stability check. Unfortunately, improvement in this direction has not been pursued in this thesis. ■

The final result in this subsection concerns about the relationship between the vectors of coefficients in eq. (2.64) and eq. (2.65). It can easily be argued that $\mathbf{a}$ is the image of $\tilde{\mathbf{a}}$ under some *linear* function.

Lemma 2.5.9. *Let* $\mathbf{a}$ *in eq. (2.65) be the vector of coefficients of* $a(\mathbf{z})$ *as in eq. (2.54). Let* $\tilde{\mathbf{a}}$ *in eq. (2.64) be the vector of coefficients of* $\tilde{a}(\omega, \mathbf{p})$ *as in eq. (2.47). If* a *and* $\tilde{a}$ *are related by Lemma 2.5.6, then there exists* $M \in \mathbb{R}^{|\mathbf{a}| \times |\tilde{\mathbf{a}}|}$ *such that*

$$\mathbf{a} = M\tilde{\mathbf{a}}.$$

■

Proof of Lemma 2.5.9. By expanding the terms in eq. (2.62), it can be seen that eq. (2.62) has exactly the same monomials as in eq. (2.54) (i.e., $\mathbf{z}^{\mathbf{k}}$, $\mathbf{k} \in \mathcal{H} \cap \mathbf{B}_{\mathbf{m}}$). In addition, the coefficients of the monomials in eq. (2.62) and eq. (2.54) are linear functions of $\tilde{\mathbf{a}}$ and $\mathbf{a}$, respectively. Therefore, equating the monomial coefficients term by term concludes the proof. ■

Remark 2.5.10. It should be noted, however, that showing the existence of the matrix M is very different from actually obtaining a formula for M. The latter task is much more cumbersome. In general, this is a task in which a parser based on a computer algebraic system can help significantly (e.g., the SOSTOOL [69] for the ordinary polynomial case).

Nevertheless, a formula will be obtained for a special case in which $n_{\mathbf{p}} = 2$ in Subsection 2.5.6. ■

To summarize, this subsection concludes with the equivalence of two positivity checks for the parameterized stability check problem. That is,

$$\tilde{a}(\omega, \mathbf{p}) > 0, \qquad \forall\, (\omega, \mathbf{p}) \in [0, 2\pi) \times \mathcal{P} \tag{2.66a}$$

$$\iff \quad a(\mathbf{z}) > 0, \qquad \forall \mathbf{z} \in \mathbb{T}, \tag{2.66b}$$

with $\tilde{a}(\omega, \mathbf{p})$ defined in eq. (2.47) and $a(\mathbf{z})$ defined in eq. (2.54), and they are connected by Lemma 2.5.6. The second check is a positivity check of a multivariate trigonometric polynomial, which will be subject of Subsection 2.5.4.

2.5.4 Multivariate trigonometric sum-of-squares relaxation

It should be emphasized that the material in this subsection is standard, and only the most relevant topics are discussed here. See [70] for an excellent description of the full list of topics.

In Subsections 2.5.2 and 2.5.3 it was established that the parameterized stability check is the positivity check of a multivariate trigonometric polynomial (see eq. (2.66b)). This computation, in a limited sense, can be performed by the use of SOS idea to be described.

This subsection first defines SOS, and then it will proceed to describe two properties of SOS – one with its computationally tractable characterization (i.e., Gram matrix representation), and the other with its relationship to positive trigonometric polynomials. Finally, the combination of these two properties will lead to the idea of SOS relaxation.

2.5.4 A: Definition of sum-of-squares

Definition 2.5.11. *A multivariate positive orthant polynomial of degree* $\mathbf{m} \in \mathbb{Z}_+^{n_{\mathbf{p}}+1}$ *is defined as*

$$h(\mathbf{z}) \;:=\; \sum_{\mathbf{k}} h_{\mathbf{k}} \mathbf{z}^{-\mathbf{k}}, \quad \mathbf{0} \le \mathbf{k} \le \mathbf{m}, \quad h_{\mathbf{k}} \in \mathbb{R}, \forall \mathbf{k}. \tag{2.67}$$

Here the inequalities are understood entry-wise. That is, $0 \le \mathbf{k}_i \le \mathbf{m}_i$, $\forall i$. ■

Definition 2.5.12. *A trigonometric polynomial $a(\mathbf{z})$ is called a sum-of-squares (SOS) if*

$$a(\mathbf{z}) = \sum_{l=1}^{\nu} h_l(\mathbf{z})\, h_l\left(\mathbf{z}^{-1}\right), \tag{2.68}$$

where $h_l(\mathbf{z})$ are positive orthant polynomials defined in eq. (2.67), and ν is a positive integer. ■

Note that the degrees of the positive orthant polynomials can actually be higher than the degree of the trigonometric polynomial. See [70, 71] for an example.

2.5.4 B: Gram matrix representation of sum-of-squares

First, it is reminded that $n_{\mathbf{p}}$ is the number of design parameters. Therefore, the (trigonometric) polynomials involved will be $n_{\mathbf{p}}+1$ variate (trigonometric) polynomials. Now, the notion of Gram matrix trigonometric polynomial characterization will be defined.

Definition 2.5.13. *A vector of ($n_{\mathbf{p}}+1$ variate) monomials θ of degree $\mathbf{m}$ is defined as*

$$\theta(\mathbf{z}) := \theta_{n_{\mathbf{p}}}\left(\mathbf{z}_{n_{\mathbf{p}}}\right) \otimes \ldots \otimes \theta_0(\mathbf{z}_0), \quad \forall \mathbf{z} \in \mathbb{C}^{n_{\mathbf{p}}+1} \tag{2.69}$$

with

$$\theta_i(\mathbf{z}_i) := \begin{bmatrix} 1 & \mathbf{z}_i & \cdots & \mathbf{z}_i^{\mathbf{m}_i} \end{bmatrix}^T \in \mathbb{C}^{\mathbf{m}_i+1}, \quad i = 0,1,\ldots,n_{\mathbf{p}}.$$

Also, denote

$$M := \prod_{i=0}^{n_{\mathbf{p}}} (\mathbf{m}_i + 1) \tag{2.70}$$

as the length of vector θ. ■

Definition 2.5.14. *A symmetric matrix $Q \in \mathbb{R}^{M\times M}$ is called a Gram matrix associated with trigonometric polynomial $a(\mathbf{z})$ of degree $\mathbf{m}$ defined in eq. (2.54) if*

$$a(\mathbf{z}) = \theta\left(\mathbf{z}^{-\mathbf{1}}\right)^T Q\, \theta(\mathbf{z}), \quad \forall \mathbf{z} \in \mathbb{Z}^{n_{\mathbf{p}}+1}, \tag{2.71}$$

where θ and M are defined in eq. (2.69) and in eq. (2.70), respectively. ■

In addition to the definition in eq. (2.54), the Gram matrix provides alternative way to characterize a trigonometric polynomial. Given a trigonometric polynomial $a(\mathbf{z})$ as in eq. (2.54), one (of the many) Gram matrix associated with it can be

$$Q = \begin{bmatrix} \mathbf{a}[1] & \mathbf{a}[2] & \cdots & \mathbf{a}[|\mathbf{a}|] \\ \mathbf{a}[2] & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{a}[|\mathbf{a}|] & 0 & \cdots & 0 \end{bmatrix}, \tag{2.72}$$

where $\mathbf{a} \in \mathbb{R}^{|\mathbf{a}|}$ is defined in eq. (2.65) and $\mathbf{a}[i]$ denotes its i^{th} entry, assuming that the ordering of the entries of $\mathbf{a}$ in Q in eq. (2.72) are consistent with that of the monomials in θ in eq. (2.69). On the other hand, given a Gram matrix, the trigonometric polynomial coefficients can be obtained by the following theorem from [70].

Theorem 2.5.15. *Let $a_{\mathbf{k}}$ be the coefficients of a trigonometric polynomial $a(\mathbf{z})$ in eq. (2.54), and let Q be a Gram matrix associated with $a(\mathbf{z})$ satisfying eq. (2.71). Then it holds that:*

$$a_{\mathbf{k}} = \mathrm{Tr}\left(T_{\mathbf{k}} Q\right), \tag{2.73}$$

where

$$T_{\mathbf{k}} = T_{\mathbf{k}_{n_{\mathbf{p}}}} \otimes \ldots \otimes T_{\mathbf{k}_0} \tag{2.74}$$

with $T_{\mathbf{k}_i} \in \mathbb{R}^{M \times M}$ being Toeplitz matrices with 1 on the $+\mathbf{k}_i$ diagonal, for all $i = 0, 1, \ldots, n_{\mathbf{p}}$.

■

For example, for $M = 4$,

$$T_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

The main benefits of using the Gram matrix representation of trigonometric polynomials is that it provides a computationally tractable way to characterize the SOS. This is summarized by the following theorem from [70].

Theorem 2.5.16. *A trigonometric polynomial $a(\mathbf{z})$ is a sum-of-squares, with the degree of h_l in eq. (2.68) less than or equal to $\mathbf{m} \in \mathbb{Z}^{n_\mathbf{p}+1}$, if and only if there exists a* **positive semi-definite** *Gram matrix $Q \in \mathbb{R}^{M \times M}$ with $M := \prod_{i=0}^{n_\mathbf{p}} (\mathbf{m}_i + 1)$ (defined in eq. (2.71)) associated with the trigonometric polynomial $a(\mathbf{z})$.* ■

Remark 2.5.17. Theorem 2.5.16 allows the linear matrix inequality (LMI) [56] characterization of SOS in terms of a positive semi-definite Gram matrix. In the event of optimization with SOS decision variables, the LMI characterization allows the optimization problem to be formulated as a SDP, which can be solved in polynomial time by interior point algorithms [67]. This is the main advantage of the Gram matrix characterization of SOS, as well as one of the two reasons of why the SOS relaxation (to be described) is utilized. ■

2.5.4 C: Sum-of-squares and positive trigonometric polynomials

The other benefit of working with SOS is its intimate relationship with positive and nonnegative trigonometric polynomials (see eq. (2.57) and eq. (2.58) for definitions), which are the objects of concerned for parameterized stability checking. Evaluated on the unit sphere, a SOS (as its name suggests) becomes

$$a(\mathbf{z}) = \sum_{l=1}^{\nu} |h_l(\mathbf{z})|^2 \geq 0, \quad \mathbf{z} \in \mathbb{T}. \tag{2.75}$$

As it is indicated by eq. (2.75), if a trigonometric polynomial is a SOS, then immediately it is nonnegative. However, it is not known whether the converse is true or not. Nevertheless, a "partial converse" turns out to be true, as stated by the following theorem from [72].

Theorem 2.5.18. *If a trigonometric polynomial is positive, then it is also a sum-of-squares.* ■

Remark 2.5.19. Intuitively, Theorem 2.5.18, together with the preceding discussion, suggests that, for any $\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$, the set of SOS of degree $\mathbf{m}$ is "sandwiched" between the set of positive trigonometric polynomials and its closure (i.e., the set of nonnegative trigonometric polynomials). This relationship can be summarized in the following schematic.

$$\{\text{positive}\} \subset \{\text{SOS}\} \subset \{\text{nonnegative}\}. \tag{2.76}$$

Set inclusion relationship in eq. (2.76) ensures that the set of SOS and the set of positive trigonometric polynomials cannot be too different. ■

2.5.4 D: Sum-of-squares relaxation

To recap, Subsection 2.5.3 establishes that the parameterized stability constraint checking problem can be formulated into two equivalent positivity checking problems in eq. (2.66a) and (2.66b). Both checks are equally hard, but the latter is a positivity check of a multivariate trigonometric polynomial. Then the set inclusion relationship in eq. (2.76) suggests that eq. (2.76) can be replaced by a check of SOS which, according to Theorem 2.5.16, can be formulated as a SDP which admits efficient solution algorithms such as interior point methods. This chain of ideas is referred to as the SOS relaxation in this chapter. The following is the schematics of the SOS relaxation.

$$\text{(hard)} \qquad \tilde{a}(\omega, \mathbf{p}) > 0, \qquad \forall\, (\omega, \mathbf{p}) \in [0, 2\pi) \times \mathcal{P} \tag{2.77a}$$

$$\text{(hard)} \qquad \Longleftrightarrow \quad a(\mathbf{z}) > 0, \qquad \forall \mathbf{z} \in \mathbb{T} \tag{2.77b}$$

$$\text{(easy)} \qquad \Longrightarrow \quad a(\mathbf{z}) \in \{\text{SOS}\}, \tag{2.77c}$$

where $\tilde{a}(\omega, \mathbf{p})$ (from eq. (2.47)) is a polynomially parameterized univariate trigonometric polynomial, and $a(\mathbf{z})$ (from eq. (2.54)) is a multivariate trigonometric polynomial.

More details should be pointed out regarding the SOS relaxation idea.

Remark 2.5.20. The right arrow in eq. (2.77c) conforms with the set inclusion relationship in eq. (2.76), and also explains the name "relaxation". However, it should be noted that the right arrow does not come trivially – it is the consequence of Theorem 2.5.18, a result that is not so obvious, and not so trivial to show. ■

Remark 2.5.21. It is obvious that not all SOS are positive trigonometric polynomials (e.g., the zero polynomial). To make sure that positivity is really enforced, the check in eq. (2.77c) can be modified to be $a(\mathbf{z}) - \varepsilon$ is SOS, for some small $\varepsilon > 0$. The real problem of SOS relaxation, however, lies in the fact that the statement in Theorem 2.5.16 does *not* completely characterize the set of SOS for any degree $\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$. This is explained in the subsequent remarks. ■

Remark 2.5.22. The positive semi-definite Gram matrix $Q \in \mathbb{R}^{M\times M}$ in Theorem 2.5.16 is insufficient to fully characterize the set of all SOS's of degree $\mathbf{m}$ because the latter set also contains SOS with positive orthant polynomials of degree higher than $\mathbf{m}$. Therefore the set $\{\text{SOS}\}$ in eq. (2.77c) (i.e., SOS relaxation) should accordingly be understood as the set of degree $\mathbf{m}$ SOS's which is representable by a positive semi-definite Gram matrix $Q \in \mathbb{R}^{M\times M}$. The limitation of the representability of the Gram matrix characterization leads to a restriction in SOS relaxation. In particular, the right arrow implication in eq. (2.77c) is no longer true – there can be positive trigonometric polynomials of degree $\mathbf{m}$ which does not belong to the $\{\text{SOS}\}$ in eq. (2.77c). ■

Remark 2.5.23. To allow a less restrictive Gram matrix characterization of the set of SOS's of degree $\mathbf{m}$, Theorem 2.5.16 can be applied to the case for $\mathbf{n} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$ such that $\mathbf{n} \geq \mathbf{m}$. In order to exclude the choices that lead to a trigonometric polynomial of degree higher than $\mathbf{m}$, additional constraints are needed. That is, for the Gram matrix $Q \in \mathbb{R}^{N\times N}$ with $N := \prod_{i=0}^{n_\mathbf{p}} (\mathbf{n}_i + 1)$, constraints such as

$$\text{Tr}\,(T_\mathbf{k} Q) = 0, \quad \forall \mathbf{k} \not\leq \mathbf{m}$$

should be enforced. ■

Remark 2.5.24. There is a price for using $\mathbf{n} \geq \mathbf{m}$ in Remark 2.5.23 because the the complexity of a SDP involved will be $O\left(N^4\right)$, which grows rather quickly with N. In practice, this means that the set of SOS's of degree $\mathbf{m}$ cannot be completely characterized using Gram matrix representation because $\mathbf{n}$ (and hence N) cannot be too large. Therefore, the SOS relaxation is not really a relaxation. Nevertheless, experimental results seem to suggest that the limitation is not crippling. ■

Remark 2.5.25. There is no analogy to Theorem 2.5.18 in the multivariate ordinary polynomial case, with the closest results pertaining only to the SOS of *rational* functions (see, Chapter 3 of [70]). The restriction in ordinary polynomial SOS adds to the list of justifications for the choice of working with trigonometric SOS instead of ordinary SOS. Nevertheless, there is a rather large body of literature regarding ordinary SOS, see, for example, [73, 74, 75]. ■

2.5.5 PMOR stability oracle – a SDP based algorithm

In this subsection, a SDP based parameterized stability oracle will be presented. As it was explained in Subsection 2.5.4, rather than checking positivity constraints such as eq. (2.77a) or eq. (2.77b) which truly corresponds to the parameterized stability constraint, it is the SOS constraint in eq. (2.77c) that is being checked in this subsection. In addition, Remark 2.5.22 in Subsection 2.5.4 concludes that the set $\{\text{SOS}\}$ in eq. (2.77c) should be restrictive – let $\mathbf{m} \in \mathbb{Z}_+^{n_\mathbf{p}+1}$ be the degree of the trigonometric polynomial considered, then the set $\{\text{SOS}\}$ in eq. (2.77c) refers to the set of SOS's of degree $\mathbf{m}$ with positive orthant polynomial degree $\mathbf{m}$ (see eq. (2.68) for definition). It is a *subset* of the set of all SOS's of degree $\mathbf{m}$. Now the SOS oracle will be presented.

Algorithm 2: PMOR SOS ORACLE

Input: query point – a vector of coefficients $\tilde{\mathbf{a}} \in \mathbb{R}^{|\tilde{a}|}$ (see eq. (2.64)). This vector defines the polynomial $\tilde{a}(\omega, \mathbf{p})$ of degree $\mathbf{m}$ in eq. (2.47).

Output: declaration of SOS constraint met, or a cut $(\alpha, \beta) : \alpha' x > \beta$, for all vector of coefficients $x \in \mathbb{R}^{|\tilde{a}|}$ corresponding to SOS's with positive orthant polynomial degree less than or equal to $\mathbf{m}$.

i. With the coefficient $\tilde{\mathbf{a}}$ for $\tilde{a}(\omega, \mathbf{p})$ of degree $\mathbf{m}$ in eq. (2.47), obtain trigonometric polynomial $a(\mathbf{z})$ in eq. (2.54) using Lemma 2.5.6 in Subsection 2.5.3.

ii. Solve the semidefinite program with decision variables $y \in \mathbb{R}$ and $Q \in \mathbb{R}^{M \times M}$,

$$\begin{aligned} \underset{y,Q}{\text{minimize}} \quad & y \\ \text{subject to} \quad & \theta\left(\mathbf{z}^{-\mathbf{1}}\right)^T Q\, \theta(\mathbf{z}) = a(\mathbf{z}) + y, \quad \forall \mathbf{z} \in \mathbb{C}^{n_\mathbf{p}+1} \\ & Q = Q' \geq 0, \end{aligned} \tag{2.78}$$

where the vector of monomials θ is of degree $\mathbf{m}$ is defined in eq. (2.69), M is defined in eq. (2.70), and $\mathbb{T}$ is defined in eq. (2.55).

iii. **if** program (2.78) is feasible and optimal $y^* < 0$,

return SOS constraint is met

else

return Cut $(\alpha,\beta) \in \mathbb{R}^{|\tilde{a}|} \times \mathbb{R}$ constructed using from the dual solution to (2.78).

The following lemma certifies the correctness of the oracle and gives a constructive proof of the existence of (α,β).

Lemma 2.5.26. *Let $\tilde{\mathbf{a}} \in \mathbb{R}^{|\tilde{\mathbf{a}}|}$ (defined in eq. (2.64)) as the coefficient vector of $\tilde{a}(\omega,\mathbf{p})$ be given. If program (2.78) is feasible and the optimal value $y^* < 0$, then $\tilde{a}(\omega,\mathbf{p}) > 0$, $\forall \omega \in [0,2\pi), \mathbf{p} \in \mathcal{P}$. Otherwise, a cut $(\alpha,\beta) \in \mathbb{R}^{|\tilde{\mathbf{a}}|} \times \mathbb{R}$ can be returned. The cut has the following property: $\alpha' x > \beta$ for all $x \in \mathbb{R}^{|\tilde{\mathbf{a}}|}$ such that the optimal objective value of program (2.78) is negative.* ■

Proof of Lemma 2.5.26. First consider the case when program (2.78) is feasible. Since $\theta\left(\mathbf{z}^{-1}\right)^T Q\,\theta(\mathbf{z}) > -\infty$, $\forall \mathbf{z} \in \mathbb{T}$ and $|a(\mathbf{z})| < \infty$, an optimal solution exists. Let it be y^*. If $y^* < 0$, then $\tilde{a}(\omega,\mathbf{p}) = a(\mathbf{z}) = \theta\left(\mathbf{z}^{-1}\right)^T Q\,\theta(\mathbf{z}) - y^* > \theta\left(\mathbf{z}^{-1}\right)^T Q\theta(\mathbf{z}) \geq 0$, $\forall \mathbf{z} \in \mathbb{T}$. Next consider the case when program (2.78) is feasible but $y^* \leq 0$. Express the polynomial equality $\theta\left(\mathbf{z}^{-1}\right)^T Q\theta(\mathbf{z}) = a(\mathbf{z})$ as equalities with the corresponding coefficients using eq. (2.73), program (2.78) can be rewritten as

$$\begin{array}{ll} \underset{y,Q}{\text{minimize}} & y \\ \text{subject to} & \text{Tr}(Q) = a_{\mathbf{0}} + y, \\ & \text{Tr}(T_{\mathbf{k}} Q) = a_{\mathbf{k}}, \quad \forall \mathbf{k} \in (\mathcal{H} \cap \mathbf{B}_{\mathbf{m}}) \setminus \{\mathbf{0}\} \\ & Q = Q' \geq 0, \end{array} \tag{2.79}$$

where $T_{\mathbf{k}}$, $\mathcal{H}$ and $\mathbf{B}_{\mathbf{m}}$ are defined in eq. (2.74), eq. (2.49) and eq. (2.50). Now consider the Lagrangian of (2.79)

$$\begin{aligned} L(\lambda) &= \underset{y,Q=Q'>0}{\text{minimize}} \{y + \lambda_{\mathbf{0}}(\text{Tr}(Q) - y - a_{\mathbf{0}}) \\ &\quad + \sum_{\mathbf{k}} \lambda_{\mathbf{k}}(\text{Tr}(T_{\mathbf{k}} Q) - a_{\mathbf{k}})\} \\ &= \underset{y,Q=Q'>0}{\text{minimize}} \{y(1-\lambda_0) + \text{Tr}(Q(\sum_{\mathbf{k}} \lambda_{\mathbf{k}} T_{\mathbf{k}})) - \sum_{\mathbf{k}} \lambda_{\mathbf{k}} a_{\mathbf{k}}\}, \end{aligned}$$

with the summation over the set $\mathcal{H} \cap \mathbf{B_m}$, and $T_{\mathbf{0}}$ being the identity matrix. It is true that

$$L(\lambda) = \begin{cases} -\sum_{\mathbf{k}} \lambda_{\mathbf{k}} a_{\mathbf{k}} & \text{if } \lambda_{\mathbf{0}} = 1, \sum_{\mathbf{k}} \lambda_{\mathbf{k}} T_{\mathbf{k}} \geq 0 \\ -\infty & \text{otherwise} \end{cases}$$

At the optimum, the optimal primal/dual pair (y^*, λ^*) has the following property

$$-\sum_{\mathbf{k}} \lambda^*_{\mathbf{k}} a_{\mathbf{k}} = y^*. \tag{2.80}$$

Recall, in Subsection 2.5.3, the definition of $\mathbf{a}$ in eq. (2.65) and the linear relationship $\mathbf{a} = M\tilde{\mathbf{a}}$ for some matrix M. Under the condition that $y^* \geq 0$, eq. (2.80) implies that $\lambda^{*\prime}\tilde{\mathbf{a}} \leq 0$. Therefore, all coefficient vectors x (of $\tilde{a}(\omega, \mathbf{p})$) that make $y^* < 0$ should satisfy

$$\lambda^{*\prime} M x > 0, \tag{2.81}$$

and therefore $(M'\lambda^*, 0)$ is the desired cut.

Finally, consider the case when (2.78) is infeasible. By argument of the statements of alternatives, infeasibility of (2.78) implies the existence of feasible dual solution λ s.t.

$$\lambda_{\mathbf{0}} = 1, \quad \sum_{\mathbf{k}} \lambda_{\mathbf{k}} T_{\mathbf{k}} \geq 0, \quad \text{and} \quad \sum_{\mathbf{k}} \lambda_{\mathbf{k}} a_{\mathbf{k}} \leq 0.$$

Therefore, $\sum_{\mathbf{k}} \lambda_{\mathbf{k}} a_{\mathbf{k}} > 0$ will lead to the same type of cut as in (2.81). ■

Remark 2.5.27. Once again it is reiterated that the SOS constraint is a restrictive version of the positivity constraint which is desirable to check, as the former check is the only tractable problem to solve. ■

Remark 2.5.28. While the specific construction of the SOS constraint oracle in Lemma 2.5.26 requires the dependence of $\tilde{a}$ on the design parameter to be polynomial, there is no restriction in the dependence of $\tilde{b}$ and $\tilde{c}$, and they can be chosen to best fit the problem at hand. ■

Remark 2.5.29. It is program (2.79) used in the proof of Lemma 2.5.26, instead of program (2.78), that is actually formulated and solved because the former is readily formulated as

the SDP "standard" form, which can be solved by solvers such as SeDuMi [76]. The details of how to construct program (2.79) will be illustrated in the following subsection through the special case in which two design parameters are allowed. ■

2.5.6 PMOR positivity oracle with two design parameters

General SOS programming problems can be formulated using available parsers such as SOSTOOLS [69]. However, this tool requires the use of computer algebraic system (e.g., MATLAB Symbolic Toolbox), which is slow in the context of cutting plane oracle application, as oracles must be called thousands of times to solve a single instance of the optimization problem. Therefore, dedicated codes for formulating (2.79) are preferred.

Consider the case in which only two design parameters are allowed. Denote the parameters as D and W (i.e., wire separation and wire width for RF inductor design). Let m be the reduced order, M and N be the highest degrees of D and W. Then in this subsection, the polynomially parameterized univariate trigonometric polynomial $\tilde{a}(\omega,\mathbf{p})$ in eq. (2.47) will be denoted as

$$\tilde{a}(\omega,D,W)=\sum_{\tilde{k}=0}^{m}\sum_{\tilde{i}=0}^{M}\sum_{\tilde{j}=0}^{N}\tilde{a}_{\tilde{i}\tilde{j}\tilde{k}}D^{\tilde{i}}W^{\tilde{j}}\cos\left(\tilde{k}\omega\right), \tag{2.82}$$

where indices $\tilde{i}$ and $\tilde{j}$ are associated with design parameter D and W and index $\tilde{k}$ is with the frequency variable ω. The triplet $\left(\tilde{k},\tilde{i},\tilde{j}\right)$ takes the role of the multi-index $\mathbf{i}$ in the definition of eq. (2.47).

Similar to the treatment in Subsection 2.5.3, the parameter set $\mathcal{P}$ will be assumed to be bounded. That is, there exist $D\in[\underline{\mathrm{D}},\bar{D}]$ and $W\in[\underline{\mathrm{W}},\bar{W}]$ such that

$$\begin{aligned} D &= D_0+D_1\left(z_D+{z_d}^{-1}\right)\\ W &= W_0+W_1\left(z_W+{z_d}^{-1}\right), \end{aligned} \tag{2.83}$$

where

$$\begin{aligned} D_0 &= 0.5(\underline{\mathrm{D}}+\bar{D})\\ W_0 &= 0.5(\underline{\mathrm{W}}+\bar{W})\\ D_1 &= 0.25(\bar{D}-\underline{\mathrm{D}})\\ W_1 &= 0.25(\bar{W}-\underline{\mathrm{W}}) \end{aligned}$$

and $z_D \in \mathbb{C}, |z_D| = 1$, $z_W \in \mathbb{C}, |z_W| = 1$. Also, a new variable z will be defined such that

$$\omega := -\sqrt{-1}\log(z). \tag{2.84}$$

With the redefinition of the indeterminates (i.e., z, z_D and z_W), the multivariate trigonometric polynomial $a(\mathbf{z})$, as in eq. (2.54), will be denoted as

$$a(\mathbf{z}) = \sum_{k=-m}^{m} \sum_{i=-M}^{M} \sum_{j=-N}^{N} a_{ijk} z_D^i z_W^j z^k, \tag{2.85}$$

with the hidden assumptions that the coefficients a_{ijk} do conform to the rule of a trigonometric polynomial. For example, $a_{ijk} = a_{-i-j-k}$. Also, it is pointed out here that in this subsection the symbol j is treated as an index, and the unit imaginary number will be denoted explicitly as $\sqrt{-1}$.

As stated in Lemma 2.5.9 in Subsection 2.5.3, the coefficients of multivariate trigonometric polynomial in eq. (2.85) are linearly related to the coefficients of the polynomial parameterized univariate trigonometric polynomial in eq. (2.82). Here, an explicit formula for the relation will be given: substituting eq. (2.83) and eq. (2.84) into $\tilde{a}(\omega, W, D)$ in eq. (2.82) leads to

$$\begin{aligned} & \tfrac{1}{2}\left(\sum_{\tilde{k}=0}^{m} \sum_{\tilde{i}=0}^{M} \sum_{\tilde{j}=0}^{N} \tilde{a}_{\tilde{i}\tilde{j}\tilde{k}} \left(D_0 + D_1\left(z_D + z_D^{-1}\right)\right)^{\tilde{i}} \left(W_0 + W_1\left(z_W + z_W^{-1}\right)\right)^{\tilde{j}} (z^{\tilde{k}} + z^{-\tilde{k}}) \right) \\ := & \sum_{k=-m}^{m} \sum_{i=-M}^{M} \sum_{j=-N}^{N} b_{ijk} z_D^i z_W^j z^k. \end{aligned} \tag{2.86}$$

Equating the coefficients of the monomials yields

$$\begin{aligned} & a_{ijk} := \\ & \tfrac{1}{2} \sum_{p=|i|}^{M} \sum_{q=|j|}^{N} \left(\sum_{s=0}^{\lfloor \frac{p-|i|}{2} \rfloor} \binom{p}{|i|+2s} D_0^{p-|i|-2s} D_1^{|i|+2s} \binom{|i|+2s}{s} \right) \cdot \\ & \left(\sum_{t=0}^{\lfloor \frac{q-|j|}{2} \rfloor} \binom{q}{|j|+2t} W_0^{q-|j|-2t} W_1^{|j|+2t} \binom{|j|+2t}{t} \right) \tilde{a}_{p,q,|k|}, \end{aligned} \tag{2.87}$$

where

$$\begin{pmatrix} p \\ q \end{pmatrix} := \frac{p!}{(p-q)!q!}$$

Note that in eq. (2.87) the indices i, j and k only appear in absolute value. This is explained by the constraint that $a(\mathbf{z})$ is a trigonometric polynomial (in fact, an ordinary polynomial of $\cos\left(-\sqrt{-1}\log(z)\right)$, $\cos\left(-\sqrt{-1}\log(z_W)\right)$, and $\cos\left(-\sqrt{-1}\log(z_D)\right)$ only). Furthermore, eq. (2.87) indicates that there can be at most $(m+1)(M+1)(N+1)$ unique coefficients in $a(\mathbf{z})$ – this is the exactly the same number of coefficients in $\tilde{a}(\omega,\mathbf{p})$.

With the multivariate trigonometric polynomial coefficients a_{ijk} clearly defined in eq. (2.87), the optimization problem in (2.79) can be set up and solved using a standard SDP solver such as SeDuMi.

2.6 Additional modifications based on designers' need

It will be shown here that the proposed Algorithm 1 (MOR) given in Section 2.3 and Algorithm 2 (PMOR) given in Section 2.5 are quite flexible, and they can serve as a basic framework which can easily be modified to account for several additional desirable constraints devised for instance from a designer's knowledge about the specific system to be modelled.

2.6.1 Explicit approximation of quality factor

When the transfer function H is for instance the impedance of an RF inductor, the accurate representation of the quality factor

$$Q(\omega) := \frac{\mathrm{Im}(H(e^{j\omega}))}{\mathrm{Re}(H(e^{j\omega}))}, \quad \omega \in [0, 2\pi)$$

is of critical importance for the designers in order to evaluate the system performance. In this case, the basic problem in (2.14) can be modified to guarantee a very good quality factor accuracy.

$$
\begin{aligned}
\underset{\tilde{a},\tilde{b},\tilde{c},\gamma}{\text{minimize}} \quad & \gamma \\
\text{subject to} \quad & \left|H(e^{j\omega})\tilde{a}(\omega) - \tilde{b}(\omega) - j\tilde{c}(\omega)\right| < \gamma\tilde{a}(\omega), \\
& \left|\frac{\text{Im}(H(e^{j\omega}))}{\text{Re}(H(e^{j\omega}))}\tilde{b}(\omega) - \tilde{c}(\omega)\right| < \rho\gamma\tilde{b}(\omega), \\
& \tilde{a}(\omega) > 0, \tilde{b}(\omega) > 0, \quad \forall\,\omega \in [0, 2\pi), \\
& \deg(\tilde{a}) = m, \deg(\tilde{b}) \leq m, \deg(\tilde{c}) \leq m.
\end{aligned}
\tag{2.88}
$$

ρ in the second set of constraint is a tuning parameter of the relative accuracy between match on frequency response and on quality factor. The oracles for program (2.88) are similar to those for program (2.14). The positive real part constraint and the reduced model should be constructed using

$$
\begin{aligned}
\underset{p,\gamma}{\text{minimize}} \quad & \gamma \\
\text{subject to} \quad & \left|H(e^{j\omega}) - \frac{p(e^{j\omega})}{q(e^{j\omega})}\right| < \gamma, \forall\,\omega \in [0, 2\pi), \\
& \left|\frac{\text{Im}(H(e^{j\omega}))}{\text{Re}(H(e^{j\omega}))} - \frac{p(e^{j\omega})q(e^{-j\omega}) - p(e^{-j\omega})q(e^{j\omega})}{p(e^{j\omega})q(e^{-j\omega}) + p(e^{-j\omega})q(e^{j\omega})}\right| < \rho\gamma \\
& p(e^{j\omega})q(e^{-j\omega}) + p(e^{-j\omega})q(e^{j\omega}) > 0, \forall\,\omega \in [0, 2\pi). \\
& \deg(p) \leq m,
\end{aligned}
\tag{2.89}
$$

Again, this program is quasi-convex, and the oracle procedure with constraint (2.29) can be applied here as well.

2.6.2 Weighted frequency response setup

In some applications the desired approximation accuracy is different in different frequency ranges. For those applications the objective function of program (2.14) can be replaced by

$$
\left\|W(z)\left(H(z) - \hat{H}(z)\right)\right\|_\infty,
$$

where $W(z)$ are weights that can be chosen to be larger for the "more important" frequency range.

2.6.3 Matching of frequency samples

Program (2.14) can be modified so that the reduced transfer function matches exactly the original transfer function at some particular frequencies ω_k between 0 and π. In order to do this, equality constraints such as

$$H\left(e^{j\omega_k}\right)\tilde{a}(\omega_k) - \tilde{b}(\omega_k) - j\tilde{c}(\omega_k) = 0, \quad \forall k$$

can be imposed. Similarly, the program (2.26) can be modified to make sure the final reduced model matches the full model at those frequencies. Besides the intended use of exact sample matching, this modification has the practical meaning of reducing the number of optimization decision variables in programs (2.14) and (2.26), hence reducing the runtime significantly.

2.6.4 System with obvious dominant poles

Algorithm 3 implements a PMOR procedure, and it is specialized in the case where the full model has a pair of "dominant poles". It is given because it can take advantage of the problem specific insight common, for instance, in RF inductor design. Note that the reduced model $\hat{H}(z,\mathbf{p})$ is stable because, as described in Algorithm 3, $|\hat{z}^*(\mathbf{p})| < 1$, and $\hat{H}(z,\mathbf{p})$ is stable $\forall \mathbf{p} \in \mathcal{P}$.

Algorithm 3: PMOR: RF INDUCTOR DESIGN
Input: $H(z,\mathbf{p})$
Output: $\hat{H}(z,\mathbf{p})$

i. Construct reduced models $\tilde{H}_{\mathbf{p}}(z)$ for each $\mathbf{p} \in \mathcal{P}_1 \subset \mathcal{P}$, where $\mathcal{P}_1$ is a finite (training) set

ii. Identify the dominant poles $z^*_{\mathbf{p}}$ of models $\tilde{H}_{\mathbf{p}}(z)$

iii. For each model $\tilde{H}_{\mathbf{p}}(z)$, construct proper "non-dominant" systems $H^1_{\mathbf{p}}(z)$ s.t.

$$\tilde{H}_{\mathbf{p}}(z) = \frac{K_{\mathbf{p}} z^2}{(z - z^*_{\mathbf{p}})(z - \overline{z^*_{\mathbf{p}}})} H^1_{\mathbf{p}}(z), \tag{2.90}$$

where $K_{\mathbf{p}} \in \mathbb{R}$.

iv. Construct global interpolation model $\hat{K}(\mathbf{p})$ and $\hat{z^*}(\mathbf{p})$. Special attention should be paid to the model $\hat{z}^*(\mathbf{p})$ to make sure that $|\hat{z}^*(\mathbf{p})| < 1, \forall \mathbf{p} \in \mathcal{P}$

v. Solve program (2.45) to find a parameterized model $\hat{H}^1(z,\mathbf{p})$ with non-dominant systems $H^1_{\mathbf{p}}(z)$ as inputs.

vi. Construct reduced model of the original system using eq. (2.90). That is,

$$\hat{H}(z,\mathbf{p}) = \frac{\hat{K}(\mathbf{p})z^2}{(z-\hat{z^*}(\mathbf{p}))(z-\overline{\hat{z^*}(\mathbf{p})})}\hat{H}^1(z,\mathbf{p}).$$

Note that in order to make sure the final model $\hat{H}(z,\mathbf{p})$ is passive, pole and zero information of the "dominant" system can be taken into account to form the numerator of the overall system when parameterized "non-dominant" system $\hat{H}^1(z,\mathbf{p})$ is being computed.

2.7 Computational complexity

There are two sources that contribute to the complexity. The first part is the computation of the frequency samples, which, when using accelerated solvers [77, 78, 79], is $O(n\log(n))$ for each frequency point, with n being the order of the full model. The examples in Section 4.8 usually required from 20 to 200 frequency samples. The second part is the cost of running the optimization algorithm. The complexity analysis here is based on the specific method of ellipsoid algorithm (which is implemented as a test code). If q and n_v are the order of the reduced model and the number of decision variables in the optimization respectively, then $n_v = O(q)$. Based on the fact that the volume of the bounding ellipsoid is reduced by at least a factor of $1-\frac{1}{n_v}$, it can be concluded that it takes $O(n_v^2) = O(q^2)$ iterations to terminate the algorithm. At each iteration of the ellipsoid algorithm, the cost is $O(q^2)$ (matrix vector product performed when updating the bounding ellipsoid). Therefore, the cost of the second part is $O(q^4)$. The overall complexity of the algorithm is summarized as

$$O(n\log(n)n_s) + O(q^4),$$

with n_s being the number of frequency samples computed. Similarly, for the parameterized case, $n_v = O(q\prod q_{pk})$ where q_{pk} is the degree of the polynomial with each parameter p_k as in (2.47) and the complexity is

$$O(n\log(n)n_s) + O(q\prod q_{pk})^4). \tag{2.91}$$

Based on our experience in running the examples in Section 4.8, the bottleneck for non-parameterized model reduction is represented by the computation of the frequency response samples, i.e. the first term in (2.91), unless the samples are available as measured data. For parameterized applications, on the contrary, the bottleneck is solving the relaxation as there are many more decision variables. Therefore, the second term of (2.91) becomes the dominating factor.

2.8 Applications and Examples

In this section several application examples are shown to illustrate how the proposed optimization based model reduction algorithm works and performs in practice. All the examples in this section were implemented in MATLAB and run on a Pentium IV laptop with 1GHz clock, 1GB of RAM and running Windows XP. A basic, stability constrained version of the proposed algorithm can be found at

`http://www.rle.mit.edu/cpg/research_codes.htm`

2.8.1 MOR: Comparison with PRIMA

In this subsection the proposed algorithm is compared with the commonly used model reduction method of moment matching. The first two examples are non-parameterized comparison. The last example is a *parameterized* modelling problem for a 2 turn RF inductor as described in [18].

RF inductor example. The first example is a comparison between multi-point moment matching (PRIMA) [8] and the proposed algorithm for reducing a 7 turn spiral RF

inductor model generated by an electro-magneto-quasi-static (EMQS) mixed potential integral equation (MPIE) solver [79]. The original model has order 1576. PRIMA is set to match 2 moments at DC, 6 moments at each of the following frequencies: 4GHz, 8GHz 12GHz. The resulting model has order 20. On the other hand, two models are constructed using the proposed method. One has order 14 using 20 frequency samples (same computational cost as PRIMA), and the other has order 20 using 40 frequency samples (same order as PRIMA). When using the proposed method, both stability and positive-real passivity oracles are checked in this example. The following error metric is computed: $\max(\frac{|H(f)-\hat{H}(f)|}{|H(f)|}), f \in [0, 14\text{GHz}]$. Comparison results are shown in Table 2.1, with QCO being the shorthand for the proposed quasi-convex optimization method.

Table 2.1: Reduction of RF inductor from field solver data using QCO and PRIMA

	QCO	QCO	PRIMA
order	14	20	20
cost (# of solves)	20	40	20
error (%) : H	6.9×10^{-3}	7.1×10^{-4}	1.8×10^{-3}

RLC line example. This is a cooked-up example in which the full model is not quite reducible. The example is presented here in order to examine how PRIMA and the proposed method perform in a poorly defined setup. In this example we reduce an RLC line segmented into 10 sections (full model order 20) with an open circuit termination. The transfer function is the admittance. The model is obtained as follows: inductor currents and capacitor voltages are the state variables. KCL is imposed at each capacitor node, and the branch equation is used between adjacent nodes. The reduced models of both methods have order 10, and PRIMA is set to match 4 moments at 10^4 rad/s, 4 moments at 5×10^4 rad/s, and 2 moments at 10^5 rad/s respectively. Figures 2-2 and 2-3 compare the magnitudes of the admittance of the full model, and the reduced models by PRIMA, and by the proposed method, respectively. The difficulties encountered when modelling this example with PRIMA are discussed in [80]. As expected, in this example PRIMA performs better locally, but the proposed method does better for the whole frequency range of interest.

PMOR of 2 turn RF inductor. In this example, the two turn RF inductor in [18] is analyzed. In [18], an 12th order parameterized reduced model was constructed using a

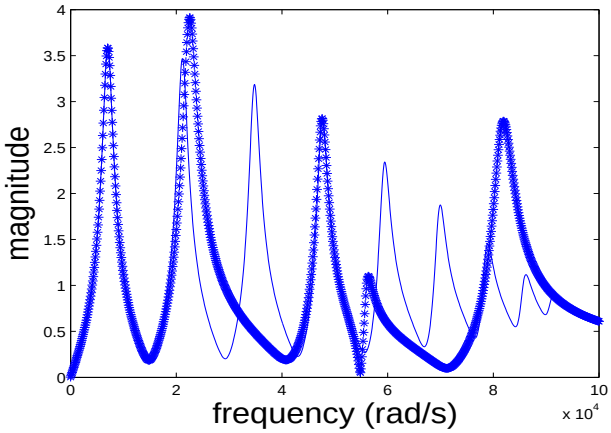


Figure 2-2: Magnitude of admittance of an RLC line. Solid line: full model. Solid with Stars: PRIMA 10th order ROM.

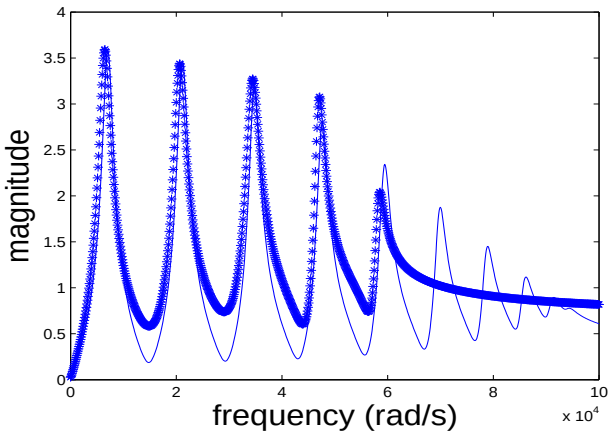


Figure 2-3: Magnitude of admittance of an RLC line. Solid line: full model. Solid with Stars: QCO 10th order ROM.

moment matching method. On the other hand, we have constructed an 8th order PROM using the proposed method. Figures 2-4 show the comparison results in [18] for the case of wire width $D = 1\mu m$ and wire separation $W = 1,\ldots,5\mu m$, with the additional result of the proposed method superimposed.

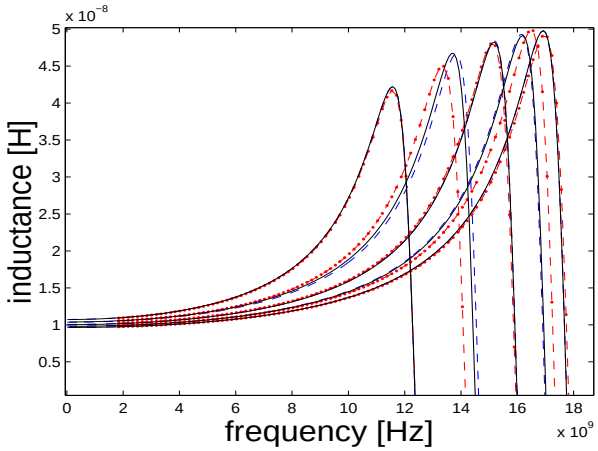


Figure 2-4: Inductance of RF inductor for different wire separations. Dash: full model. Dash-dot: moment matching 12th order. Solid: QCO 8th order.

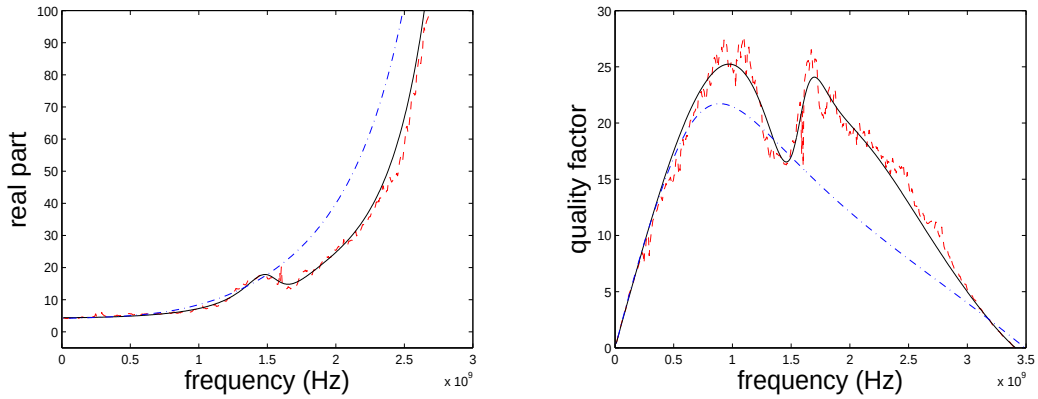
2.8.2 MOR: Comparison with a rational fit algorithm

In the third example we compare the proposed method with an existing optimization based rational fit [55, 14, 57] by constructing a reduced model from measured frequency response of a fabricated spiral RF inductor [81]. In this example, the order of the reduced model is 10, and the positive real part constraint is imposed. Frequency weights (preferring samples of up to 3GHz) are used, and the quality factor is explicitly minimized. In particular, program (2.88) is solved with tuning parameter $\rho = 10^{-4}$. Runtime for the proposed method was 60 seconds. On the other hand, rational fit [55], vector fitting [14] and passivity enforcement [57] were used in combination to construct another passive model for comparison. The runtime for running the mentioned algorithms was 30 seconds.

Fig 2-5.a and 2-5.b show the real part of the impedance, and the quality factor of the model produced by the proposed approach comparing to measured data and to a model of the same order generated using the optimization based approaches in combination.

2.8.3 MOR: Comparison to measured S-parameters from an industry provided example

In the fourth example we identify a reduced model from measured multi-port S-parameter data. 390 frequency response samples have been measured on a commercial graphic card.



(a) real part of impedance (b) quality factor

Figure 2-5: Identification of RF inductor. Dash line: measurement. Solid line: QCO 10th order reduced model. Dash-dot line: 10th order reduced model using methods from [14, 55, 57].

The internal architecture and implementation details are not available. Although the original data is multi-input-multi-output, data from only one port is used to construct the reduced model. Figure 2-6 shows the comparison result for the corresponding ports. The reduced model is order 20. The model was identified in 30 seconds.

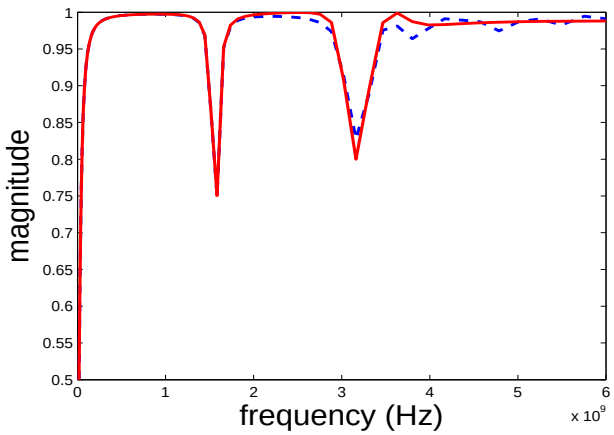


Figure 2-6: Magnitude of one of the port S-parameters for an industry provided example. Solid line: reduced model (order 20). Dash line: measured data (almost overlapping).

2.8.4 MOR: Frequency dependent matrices example

In the fifth example we apply the proposed method to reduce a model of an RF inductor generated by a full wave MPIE solver accounting for the substrate effect using layered Green's functions [82, 79]. Since the system matrices are frequency dependent, the order of the full model is infinite. The order of the reduced model is 6 and the positive real part constraint is imposed. Computation time was 2 seconds. Figure 2-7 shows the result of the quality factor.

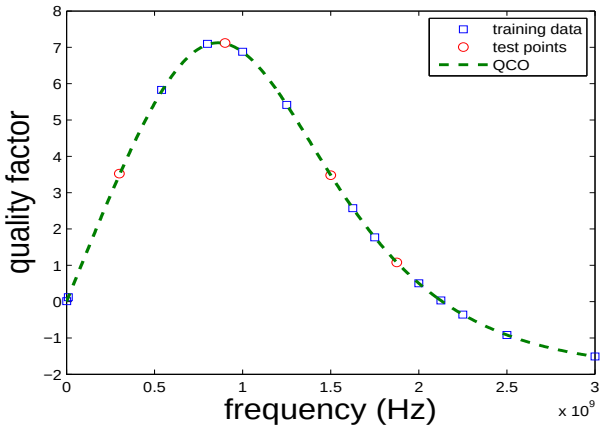


Figure 2-7: Quality factor of an RF inductor with substrate captured by layered Green's function. Full model is infinite order and QCO reduced model order is 6.

2.8.5 MOR: Two coupled RF inductors

A 10^{th} order passive reduced model of two coupled 4 turn RF inductors (identical, side by side) was constructed. It took about 120 seconds to build the reduced model. Figure 2-8 shows the result for the magnitude and phase of S12.

2.8.6 PMOR of fullwave RF inductor with substrate

In this example an 8^{th} order passive parameterized reduced model is constructed for an RF inductor with substrate. The full model has more than 2000 states (quasi-static). The design parameters are wire width (W) and wire separation (D). The parameter space is a square

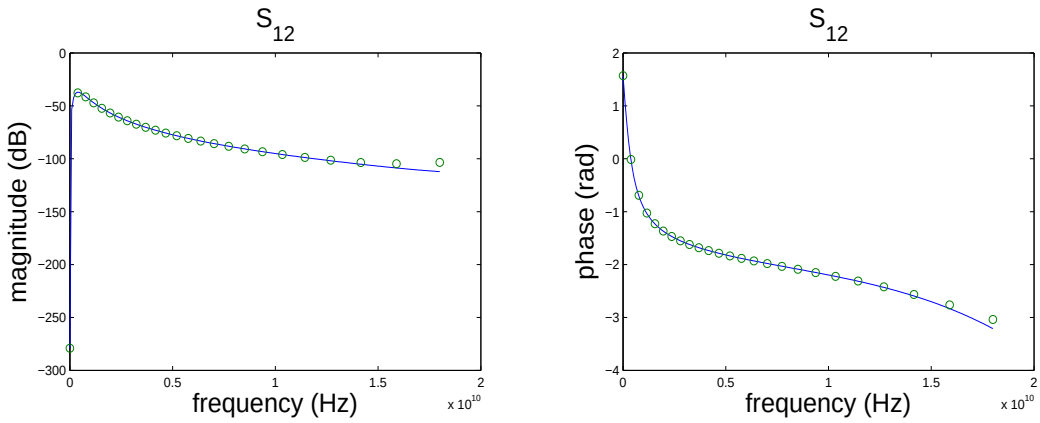


Figure 2-8: S12 of the coupled inductors. Circle: Full model. Solid line: QCO reduced model.

from (1,1) to (5,5) microns. In constructing the reduced model, 25 (W,D) pairs forming a grid of $(1:5)\times(1:5)$ were used as training data. The reduced model is tested with simulation results from field solver on a $\big((1.5:1:4.5)\times(1.5:1:4.5)\big)$ grid, and Figure 2-9 shows the result. Construction of reduced model took overnight.

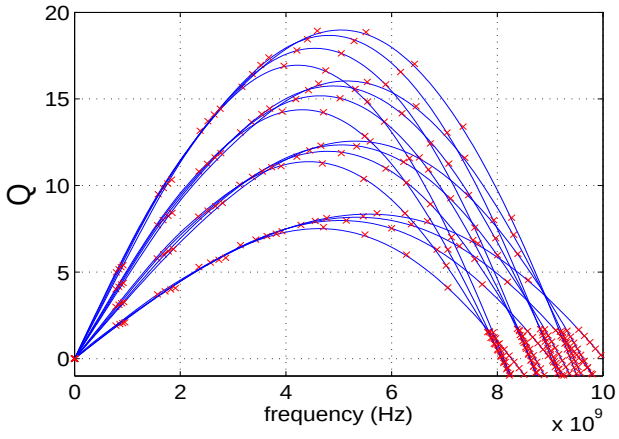


Figure 2-9: Quality factor of parameterized RF inductor with substrate. Cross: Full model from field solver. Solid line: QCO reduced model.

2.8.7 PMOR of a large power distribution grid

In this example a passive parameterized reduced model of a power distribution grid is built using the techniques in Subsection 2.6.3, and those similar to Algorithm 3. The design

parameters are die size $D \in [7,9]$mm, and wire width $W \in [2,20]\mu$m. 25 full models distributed uniformly in the design space are used as training points for the reduced model of order 32. To test the parameterized reduced model, comparison of full model and reduced model is done at parameters $D \in \{8.25, 8.75\}$mm and $W \in \{4,8,12,14,18\}\mu$m. Figures 2-10 and 2-11 show the result at $D = 8.25$ mm and $D = 8.75$ mm, respectively.

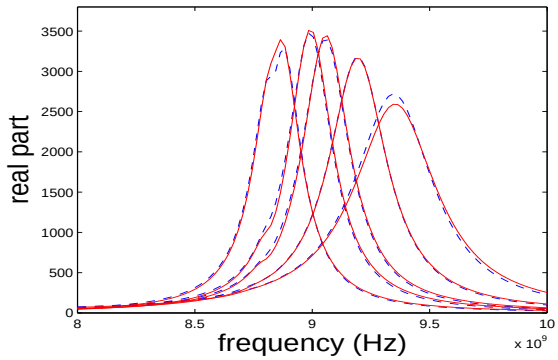


Figure 2-10: Real part of power distribution grid at $D = 8.25$ mm and $W = 4,8,12,14,18$ um. Dash: Full model. Solid: QCO reduced model.

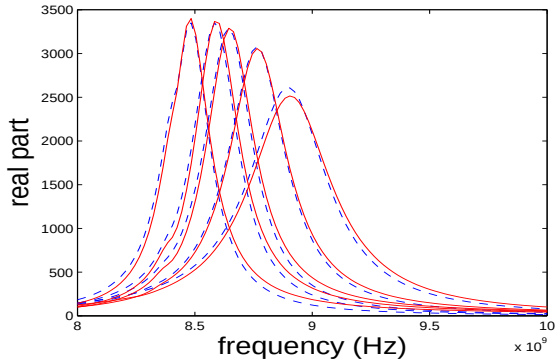


Figure 2-11: Real part of power distribution grid at $D = 8.75$ mm and $W = 4,8,12,14,18$ um. Dash: Full model. Solid: QCO reduced model.

2.9 Conclusion

In this chapter a relaxation framework for the optimal $\mathcal{H}_\infty$ norm MOR problem is proposed. The framework has been demonstrated to perform approximately as well as PRIMA when

reducing large systems, and better than PRIMA for examples that require a more global accuracy in frequency response. Unlike PRIMA, the proposed method has a guaranteed error bound, and it can reduce models with frequency dependent system matrices, hence it can capture for instance substrate and fullwave effects. Unlike other optimization based methods, the proposed method has been shown to be very flexible in preserving stability and passivity. Finally, the proposed optimization setup has also been extended to solve parameterized MOR problems. Several examples have been presented validating both the MOR and PMOR approaches against field solvers and measured data on large RF inductors, IC power distribution grids and industrial provided package examples.

Chapter 3

Bounding L2 Gain System Error Generated by Approximations of the Nonlinear Vector Field

A growing number of results can be found in the literature addressing the problem of nonlinear model order reduction. For example, [31, 32, 33, 34, 35, 36] employ Volterra series and moment matching techniques to solve the "weakly nonlinear" model order reduction problem. Another class of methods based on piecewise approximations address strongly nonlinear problems [24, 25, 26, 27, 28, 29, 30]. Both of the weakly and strongly nonlinear methods involve the following two steps: a state projection to a lower dimensional subspace and the approximation of the reduced nonlinear vector field to facilitate simulation. However, to the best of our knowledge, there has not been any published result in the field of electronic design automation regarding the approximation quality of the approximation step above. The work in this chapter presents an effort in this direction for a practical dynamical system settings for applications in integrated circuit design as follows.

$$\begin{aligned} \dot{x}(t) &= Ax(t) + \Phi(x(t)) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3.1}$$

where $A \in \mathbb{R}^{q\times q}$, $B \in \mathbb{R}^{q\times 1}$, $C \in \mathbb{R}^{1\times q}$. $\Phi : \mathbb{R}^q \mapsto \mathbb{R}^q$ is a general *reduced* vector field. For example, $\Phi(\cdot) = V'\Phi_f(V\cdot)$ for some projection matrix $V \in \mathbb{R}^{n\times q}$ (e.g., see [83]). Typically,

q is a small positive integer (e.g., $q = 10$). On the other hand, $\Phi_f : \mathbb{R}^n \mapsto \mathbb{R}^n$ is the full order nonlinear vector field with $n \gg q$. When the reduced nonlinear vector field Φ is approximated by $\tilde{\Phi}$, system (3.1) becomes

$$\begin{aligned} \dot{x}(t) &= Ax(t) + \tilde{\Phi}(x(t)) + Bu(t) \\ y(t) &= Cx(t). \end{aligned} \tag{3.2}$$

To reiterate, two *reduced* systems have been defined – the original system in eq. (3.1) and the approximated system in eq. (3.2). The two systems are of the same order. The objective of this chapter is to relate the error between nonlinear functions Φ and $\tilde{\Phi}$ to the error between systems (3.1) and (3.2) described in Figure 3-1.

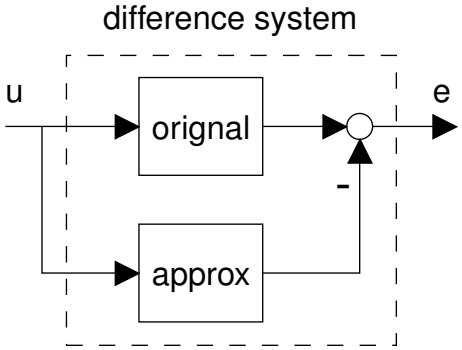


Figure 3-1: The difference system setup. The original system in eq. (3.1) and the approximated system in eq. (3.2) are driven by the same input u, and the difference between the corresponding outputs is taken to be the difference system output denoted as e. The L2 gain (to be defined in Subsection 3.2.2) from u to e for the difference system is a reasonable metric for the approximation quality between the systems in eq. (3.1) and eq. (3.2).

The rest of the chapter is organized as follows: Section 3.1 presents a motivating application example explaining why the error bounding problem should be considered. Section 3.2 summarizes background materials such as the small gain theorem which forms the basis of the development of this chapter. In Section 3.3 the system error will formally be introduced as the L2 gain of a difference system, which will be analyzed by the robustness analysis technique (i.e., the small gain theorem). Section 3.4 presents the main theoretical contribution: under some assumptions, the L2 gain of the difference system is upper bounded by the L2 gain of $\Phi(\cdot) - \tilde{\Phi}(\cdot)$ with a positive multiplicative constant. Based again

on the small gain theorem, a numerical procedure is presented in Section 3.5 to compute a more convenient upper bound of the L2 gain of the difference system using the L2 gain information of $\Phi(\cdot) - \tilde{\Phi}(\cdot)$. Finally, in Section 3.6, the numerical procedure from Section 3.5 is applied to some nonlinear system model reduction problem to validate the statements.

3.1 A motivating application

This subsection presents a specific (but more restrictive) application to illustrate why an approximation such as (3.2) is useful, and why it would be interesting to provide a bound for the induced system error. Consider the more specific setup

$$\begin{aligned} \dot{x}(t) &= Ax(t) - V'\Phi_f(Vx(t)) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3.3}$$

where $A \in \mathbb{R}^{q\times q}$, $V \in \mathbb{R}^{n\times q}$, $B \in \mathbb{R}^{q\times 1}$, $C \in \mathbb{R}^{1\times q}$. $\Phi_f : \mathbb{R}^n \mapsto \mathbb{R}^n$:

$$\Phi_f(v) = \begin{bmatrix} \phi_f(v_1) & \phi_f(v_2) & \cdots & \phi_f(v_n) \end{bmatrix}',$$

where $\phi_f : \mathbb{R} \mapsto \mathbb{R}$ is any nonlinear function. Note that system (3.3) has repeated nonlinearities, and it can model for instance any circuit with repeated nonlinear elements, such as the diode transmission line to be discussed in Section 3.6. Furthermore, the method in this example can be modified by appending the nonlinear function Φ_f with different nonlinearities, at the expense of a more complicated derivation and computation. However, it should be emphasized that the mentioned restriction in system (3.3) pertains only to this example, and not to the main result of this chapter.

System (3.3) can be considered as the result of applying for instance a congruence transformation on a model of order n using a projection matrix V, where n and q (with $n \gg q$) are the orders of the full and reduced models respectively. A common complaint about the applicability of system (3.3) is that when using the model in simulation, the nonlinear function ϕ_f must be evaluated n times for every reduced vector field evaluation. Therefore finding an approximation function $g : \mathbb{R}^q \mapsto \mathbb{R}^q$, such that $g(w) \approx V'\Phi_f(Vw), \forall w \in \mathbb{R}^q$,

with an evaluation cost much cheaper than $O(n)$, would be of great interest for most nonlinear model order reduction techniques. A few results can be found about this topic. For example, [84] investigated the possibility of using Kernel methods for such a construction, while [85, 86] proposed methods based on polynomial (Taylor series) approximation of $V'\Phi_f(V\cdot)$.

However, when considering the special case (3.3), it would be much more convenient to find an approximation to the scalar nonlinear function ϕ_f, instead of the entire vector field. For example, if ϕ_f is approximated by a scalar polynomial of degree d,

$$\phi_f(z) \approx \tilde{\phi}_f(z) = \sum_{k=0}^{d} p_k z^k, \tag{3.4}$$

and accordingly

$$\Phi_f(v) \approx \tilde{\Phi}_f(v) := \begin{bmatrix} \tilde{\phi}_f(v_1) \\ \tilde{\phi}_f(v_2) \\ \vdots \\ \tilde{\phi}_f(v_n) \end{bmatrix}, \tag{3.5}$$

then the corresponding vector field approximation is a q vector of q-variate polynomials of degree d

$$V'\Phi_f(Vx) \approx V'\tilde{\Phi}_f(Vx) = \sum_{\beta} c_\beta x^\beta, \tag{3.6}$$

where $\beta \in \mathbb{Z}_+^q, \beta = (\beta_1, \beta_2, \ldots, \beta_q), \sum_j \beta_j \leq d$, $c_\beta \in \mathbb{R}^q$ and x^β is shorthand for $\prod_j x_j^{\beta_j}$. The approximated system becomes

$$\begin{aligned} \dot{x}(t) &= Ax(t) - V'\tilde{\Phi}_f(Vx(t)) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3.7}$$

The above polynomial approximation scheme has the following benefits:

1. Approximating a scalar nonlinear function ϕ_f is much easier than approximating the vector-valued nonlinear function $V'\Phi_f(V\cdot)$.

2. It can be verified that the coefficient vectors c_β can be computed efficiently.

3. The Jacobian of the approximated vector field is

$$A - V' \mathrm{diag}\left(\frac{d\tilde{\phi}_f}{dz}, \ldots, \frac{d\tilde{\phi}_f}{dz}\right) V. \tag{3.8}$$

If A is symmetric and Hurwitz, the Jacobian can be constrained to be Hurwitz simply by constraining the univariate polynomial $\frac{d\tilde{\phi}_f}{dz}$ to be nonnegative, which is true if and only if it is a sum of squares of polynomials, and this condition can in turn be efficiently enforced using linear matrix inequalities (LMI) [73].

However, there are two issues that are worth considering:

- Estimating and controlling the cost of evaluating the polynomial approximated vector field.
- Providing precise statements about the accuracy of the approximation quality in terms of quantifiable system measures such as the L2 gain (to be defined in Subsection 3.2.2) of the difference system of (3.3) and (3.7).

The answer to the first question depends on the specific application. The computation cost for evaluating nonlinear vector field $V'\tilde{\Phi}_f(V\cdot)$ is $O\left(q\begin{pmatrix} q+d \\ d \end{pmatrix}\right)$. Since such cost is independent of n, and since typically $n \gg \max\{q,d\}$, computation efficiency is greatly improved. However, as also pointed out in [84], $\begin{pmatrix} q+d \\ d \end{pmatrix}$ is admittedly still a large number even for not excessively large q and d. Measures should be taken to control computational complexity, but this will not be discussed here, as it is not the main focus.

Instead, this chapter presents results that address the second issue: providing statements about the accuracy of the approximation. In particular, under the assumptions that system (3.3) has finite incremental L2 gain (to be defined in Subsection 3.2.3) and stability, it will be shown that the L2 gain from input u to the difference of output y of systems (3.3) and (3.7) is bounded by a linear function of the L2 gain of the difference of the scalar nonlinear functions $\phi_f(\cdot) - \tilde{\phi}_f(\cdot)$, if the latter difference is small enough. In addition, this chapter presents a framework for numerically calculating an a priori (i.e., before simulation) error

bound of the L2 gain of the difference system, again based on the L2 gain of $\phi_f(\cdot) - \tilde{\phi}_f(\cdot)$. Finally, it should be noted that the results of this chapter are valid for a more general framework (3.1) than what is discussed in this motivating application subsection. Namely, the system error is presented in terms of (3.1) and (3.2), and the vector field approximation error is between general nonlinearities Φ and $\tilde{\Phi}$.

3.2 Technical Background

3.2.1 L2 gain of a memoryless nonlinearity

Let $u \in \mathbb{R}^m$ and $y \in \mathbb{R}^p$ be the input and output of a memoryless nonlinearity F (i.e., $y = F(u)$). Then the L2 gain γ_F of the memoryless nonlinearity F is defined as

$$\gamma_F := \sup_{u \neq 0} \frac{\|F(u)\|_2}{\|u\|_2} \tag{3.9}$$

3.2.2 L2 gain of a dynamical system

Let $u : \mathbb{R}_+ \mapsto \mathbb{R}^m$ and $y : \mathbb{R}_+ \mapsto \mathbb{R}^p$ denote the (finitely L2 integrable) input and output signals of a dynamical system. The L2 gain γ of a system is defined as

$$\gamma := \inf_{r \geq 0} r : \inf_{T \geq 0} \int_0^T \left(r^2 \|u(\tau)\|_2^2 - \|y(\tau)\|_2^2 \right) d\tau > -\infty. \tag{3.10}$$

for all valid input/output pairs (u, y). For the rest of the chapter, unless noted otherwise, L2 gain related integrals inequalities are assumed to hold for *all* valid input/output pairs.

Intuitively, finiteness of the L2 gain of a system means that the output energy is no more than a constant times the input energy, and hence the L2 gain can serve as a notion for stability. In addition, if the L2 gain is small, then the system can be considered "small", in the sense that it needs a very strong input to excite any non-negligible output. In particular, it is desirable that the difference system in Figure 3-1 has very small L2 gain.

3.2.3 Incremental L2 gain of a system

Let (u,y) be any input/output pair of a system. Then the incremental L2 gain γ of a dynamical system is defined as

$$\gamma := \inf_{r\geq 0} r : \inf_{T\geq 0} \int_0^T \left(r^2 \left(\|u_1(\tau) - u_2(\tau)\|_2^2 \right) - \left(\|y_1(\tau) - y_2(\tau)\|_2^2 \right) \right) d\tau \geq 0, \tag{3.11}$$

for every (u_1,y_1) and (u_2,y_2) satisfying

$$\inf_{T\geq 0} \int_0^T \|u_1(\tau) - u_2(\tau)\|_2^2 \, d\tau < \infty. \tag{3.12}$$

Incremental L2 gain of a system can be used to quantify the sensitivity of the output to a perturbation in the input. In particular, a system having a finite incremental L2 gain means for each input there is a unique output corresponding to it.

3.2.4 Small gain theorem

The small gain theorem is a collection of statements bounding the L2 gain of the feedback interconnection of a nominal model G and a disturbance Δ, using the L2 gains of the individual constituents. See for example [87], for a more detailed account of these statements. The statement relevant to the discussion of the thesis is the following.

Theorem 3.2.1. *Consider the feedback connection in Figure 3-2.*

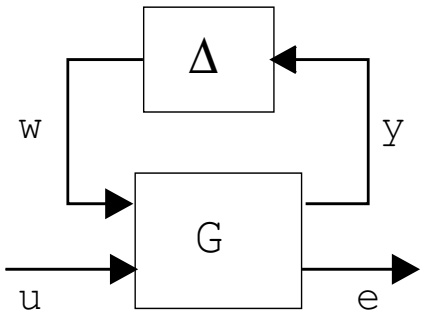


Figure 3-2: Feedback interconnection of a nominal plant G and disturbance Δ.

Let γ_G be the L2 gain of G (from $[w;u]$ to $[y;e]$), and γ_Δ be the L2 gain of Δ (from y to

w). If $\gamma_G\gamma_\Delta \leq 1$ *then the L2 gain of the feedback connection (from u to e) is less than or equal to* γ_G. ■

See, for example [87], for a proof. The small gain theorem is the fundamental tool upon which the main results of this chapter are based. The discussion of how to apply the theorem in the context of this chapter will be presented in Section 3.3.

3.2.5 Nonlinear system L2 gain upper bounding using integral quadratic constraints (IQC)

This subsection only presents the IQC analysis topics that are relevant to the development of the thesis. See [88] for the rest of the topics.

Consider the system in (3.1). If there exists a nonnegative number γ and a nonnegative and continuously differentiable function $W : \mathbb{R}^q \mapsto \mathbb{R}_+$ and the following inequality holds

$$\gamma^2 \|u\|_2^2 - \|y\|_2^2 - (\nabla_x W)' \dot{x} \geq 0, \quad \forall (x,u) \in \mathbb{R}^q \times \mathbb{R}, \text{ satisfying system (3.1)}, \tag{3.13}$$

then $\forall\, T > 0$

$$\int_0^T \left(\gamma^2 \|u\|_2^2 - \|y\|_2^2\right) d\tau \geq W(x(T)) - W(x(0)) > -\infty, \tag{3.14}$$

and therefore γ is an upper bound for the L2 gain of system (3.1) and W is a certificate for proving the L2 gain upper bound. A class of nonnegative functions $W(x)$ that is particularly convenient for analysis is the quadratic function $W(x) = x'Px$ for some symmetric positive semidefinite matrix $P \in \mathbb{R}^{q\times q}$ because the search for the matrix P can be carried out efficiently as a SDP [56]. Using quadratic certificate $W(x) = x'Px$, eq. (3.13) becomes

$$\gamma^2 \|u\|_2^2 - \|Cx\|_2^2 - 2x'P\left(Ax - V'w + Bu\right) \geq 0, \quad \forall\, (x,u) \in \mathbb{R}^q \times \mathbb{R} \text{ and } w = \Phi(Vx). \tag{3.15}$$

For a general nonlinear vector field Φ, showing the existence of $P \geq 0$ and γ that satisfy inequality (3.15) is difficult. However, the technique of IQC analysis [88] can be employed

here: first introduce a quadratic functional $\sigma(x,w)$ that satisfies the following property

$$w=\Phi(Vx) \quad \text{implies} \quad \sigma(x,w)=\begin{bmatrix} x \\ w \end{bmatrix}' \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12}' & \Sigma_{22} \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} \geq 0, \tag{3.16}$$

then remove the constraint $w=\Phi(Vx)$ in (3.15) and instead solve the following for a quadratic certificate.

$$\gamma^2\|u\|_2^2-\|Cx\|_2^2-2x'P\left(Ax-V'w+Bu\right)-\sigma(x,w)\geq 0, \quad \forall x,w,u \tag{3.17}$$

Note that if γ and P satisfy (3.17) then they automatically satisfy (3.15) by the definition of σ (3.16). But the converse is not necessarily true, therefore searching for γ and P through (3.17) results in fewer options. However, (3.17) has the advantage that it can be written as a LMI (with respect to P and $r:=\gamma^2$).

$$\begin{bmatrix} -C'C-PA-A'P-\Sigma_{11} & PV'-0.5\Sigma_{12} & -PB \\ VP-0.5\Sigma_{12}' & -\Sigma_{22} & 0 \\ -B'P & 0 & rI \end{bmatrix} \geq 0. \tag{3.18}$$

More generally, if there exist more quadratic functionals $\sigma_1,\sigma_2,\ldots$ such that

$$w=\Phi(Vx) \quad \text{implies} \quad \sigma_i(x,w)\geq 0, \quad \forall i,$$

then solving the following LMI feasibility problem (with decision variables $P,r,\tau_i\geq 0$)

$$\gamma^2\|u\|_2^2-\|Cx\|_2^2-2x'P\left(Ax-V'w+Bu\right)-\sum_i \tau_i\sigma_i(x,w)\geq 0, \quad \forall x,w,u \tag{3.19}$$

would result in a less conservative search than the feasibility problem with (3.17) because if (r,P) satisfy (3.17) then they also satisfy (3.19) simply by picking $\tau_j=0, j\geq 2$, while the converse is not necessarily true. Note also that the search with (3.19) is more restrictive than that with (3.15) for the same reason mentioned in the case of a single σ.

In summary, in order to find an upper bound of the L2 gain of a system of the form (3.1).

The following procedure can be used: first collect characterizations of the nonlinearity Φ in the form of IQCs $\sigma_1, \sigma_2, \ldots$, then setup and solve the following SDP.

$$\begin{array}{ll} \underset{r,P,\tau_i \geq 0}{\text{minimize}} & r \\ \text{subject to} & \text{LMI (3.19)} \\ & r \geq 0 \\ & P = P' \geq 0. \end{array} \tag{3.20}$$

Note that the L2 gain upper bound provided by such a procedure can be strictly greater than the true L2 gain because the class of certificates is restricted to quadratic (which is generally not rich enough except for the LTI case). Furthermore, inequalities such as (3.19) do not allow all the options (in terms of r and P) that satisfy (3.15). Nevertheless, this is a practical method for nonlinear system L2 gain upper bounding because of its tractability.

3.3 Error Bounding with the Small Gain Theorem

This section first sets up the L2 gain error bounding problem as the L2 gain upper bounding problem of the difference system. The difference system is formulated as a feedback connection between a "nominal" plant that does not contain any approximation vector field, and the "disturbance" part consisting of the error of the vector fields. The L2 gain upper bounding problem is then analyzed by the small gain theorem, which is a standard part of robustness analysis. However, the small gain theorem can be conservative in some cases, especially when the L2 gain of the disturbance part is small. To allow a more general use of the small gain theorem, the first contribution of the chapter is presented, namely a scaling parameter is introduced in the feedback. Finally the ramification of the reformulations will be discussed.

3.3.1 System error bounding problem

Definition 3.3.1. *The error between systems (3.1) and (3.2) is defined as the L2 gain (from u to e) of the following difference system (see Figure 3-1 for its block diagram).*

$$\begin{aligned} \dot{x_1} &= Ax_1 + \Phi(x_1) + Bu \\ \dot{x_2} &= Ax_2 + \tilde{\Phi}(x_2) + Bu \\ e &= C(x_1 - x_2). \end{aligned} \tag{3.21}$$

Therefore, the error bounding problem of this chapter is to find upper bounds of the L2 gain of system (3.21) using the L2 gain information of $\Phi - \tilde{\Phi}$. ■

3.3.2 Difference system formulated as a feedback interconnection

System (3.21) can equivalently be written as

$$\begin{aligned} \dot{x_1} &= Ax_1 + \Phi(x_1) + Bu \\ \dot{x_2} &= Ax_2 + \Phi(x_2) + Bu + w \\ e &= C(x_1 - x_2) \\ y &= x_2 \\ w &= \tilde{\Phi}(y) - \Phi(y). \end{aligned} \tag{3.22}$$

It can be seen that system (3.22) fits in the small gain theorem framework in Figure 3-2. In particular, system G in the figure corresponds to the part of system (3.22) with input/output $[w;u]$ and $[y;e]$ and the disturbance in the figure being $\Delta(y) = \tilde{\Phi}(y) - \Phi(y)$. The feedback structure of system (3.22) suggests the use of the small gain theorem in Subsection 3.2.4. However, the small gain theorem cannot be readily applied because the assumption $\gamma_G\gamma_\Delta \leq 1$ might not be satisfied. More importantly, even if the assumption $\gamma_G\gamma_\Delta \leq 1$ is satisfied, direct application of the small gain theorem can lead to a too conservative L2 gain upper bound of system (3.22) – the small gain theorem provides the bound $\gamma = \gamma_G$ which is independent of γ_Δ, while it would be desirable if $\lim_{\gamma_\Delta \to 0} \gamma = 0$, since the L2 gain of the difference of two identical systems should be zero. This latter difficulty can be resolved through the use of a scaling parameter discussed in the next subsection.

3.3.3 Small gain theorem applied to a scaled feedback

Consider Figure 3-3, which is equivalent to Figure 3-2.

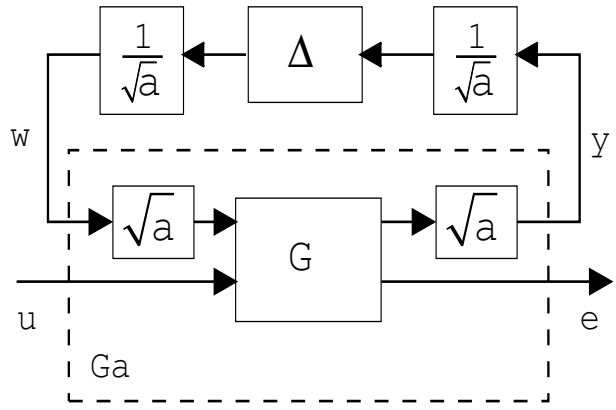


Figure 3-3: Feedback interconnection of a nominal plant G and disturbance Δ with mutually cancelling parameters $\sqrt{a}$ and $\frac{1}{\sqrt{a}}$. G_a is the original plant parameterized by the scalar a.

For the rest of the chapter the scalar a is assumed to be nonnegative. System G_a in the figure has the form

$$\begin{aligned} \dot{x_1} &= Ax_1 + \Phi(x_1) + Bu \\ \dot{x_2} &= Ax_2 + \Phi(x_2) + Bu + \sqrt{a}w \\ e &= C(x_1 - x_2) \\ y &= \sqrt{a}x_2. \end{aligned} \tag{3.23}$$

Application of the small gain theorem to the feedback system in Figure 3-2 results in the following statement.

Theorem 3.3.2. *Let γ_{G_a} be the L2 gain of system (3.23), from $[u;w]$ to $[e;y]$. If $\frac{\gamma_{G_a}\gamma_\Delta}{a} \leq 1$, then the L2 gain of the feedback interconnection (3.22), from u to e, is $\gamma \leq \gamma_{G_a}$.* ■

Remark 3.3.3. Since Theorem 3.3.2 holds for all value of a, it would be natural to choose the value of a which minimizes the small gain theorem L2 gain bound γ_{G_a}. In order to manipulate the L2 gain bound, it would be necessary to study how γ_{G_a} and $\gamma_{G_a}\gamma_\Delta/a$ change with a. In Section 3.4 a statement (Lemma 3.4.1) will be shown that $\gamma_{G_a} = O(\sqrt{a})$ if $a \leq 1$, then Theorem 3.3.2 can be applied to form another statement (Theorem 3.4.3) that gives some theoretical insight into the solution of the error bounding problem in Definition 3.3.1.

On the other hand, in section 3.5, the IQC analysis procedure described in 3.2.5 will be applied to directly compute an upper bound for γ_{G_a} numerically. Then the application of Theorem 3.3.2 leads to a numerical procedure to solve the problem in Definition 3.3.1. ■

3.4 A Theoretical Linear Error Bound in the Limit

In this section, to apply theorem 3.3.2 in Subsection 3.3.3 to solve the error bounding problem in Definition 3.3.1, it will be shown as Lemma 3.4.1 in Subsection 3.4.1 that under some assumption, the inequality

$$\exists c \geq 0 : \gamma_{G_a} \leq c\sqrt{a}, \quad \forall a \leq 1 \tag{3.24}$$

holds. With eq. (3.24), the following can be implied.

- If $a > 1$, then inequality (3.24) does not hold, hence in this case unfortunately Theorem 3.3.2 does not apply.
- If $a \leq 1$, then eq. (3.24) holds. From eq. (3.24) it can be seen that it would be desirable to choose a as small as possible. However, from what can be guaranteed by eq. (3.24), as a goes to zero, the term $\gamma_{G_a}\gamma_\Delta/a$ goes to infinity, hence violating the small gain theorem assumption $\gamma_{G_a}\gamma_\Delta/a \leq 1$. Therefore, there is a tradeoff between choosing a small to obtain the tightest possible L2 gain upper bound and choosing a large enough so that Theorem 3.3.2 still applies. The choice of the nontrivial minimum of a will be given in Subsection 3.4.2 as part of Theorem 3.4.3.
- When $a = 0$, eq. (3.24) states that the L2 gain of G_a should be zero. This is indeed the case because $G_a|_{a=0}$ is the difference of two identical systems. Therefore, if $\gamma_\Delta = 0$ (i.e., $\tilde{\Phi} = \Phi$), then Theorem 3.3.2 can be applied with a chosen to be zero, thus providing the expected zero L2 gain bound.

The rest of this section of this section is organized as follows. In Subsection 3.4.1 Lemma 3.4.1 will be shown, and then in Subsection 3.4.2 Theorem 3.4.3 will be shown as a direct consequence of Lemma 3.4.1.

3.4.1 A preliminary lemma

First consider the system with input g and output z

$$\begin{aligned} \dot{x} &= Ax + \Phi(x) + g \\ z &= Cx, \end{aligned} \tag{3.25}$$

where the matrices and functions are as defined in (3.21), except for the arbitrary function g. Define

$$\begin{aligned} &\gamma_1 \text{ as the incremental L2 gain of (3.25) from } g \text{ to } z, \\ &\gamma_2 \text{ as L2 gain of (3.25) from } [u;w] \text{ to } x \text{ when } g \equiv Bu + w, \end{aligned} \tag{3.26}$$

Lemma 3.4.1. *Let γ_1 and γ_2 be the quantities defined in (3.26). Denote γ_{G_a} as the L2 gain of system G_a (3.23), from $[u;w]$ to $[e;y]$. If $\gamma_1 < \infty$ and $\gamma_2 < \infty$, then*

$$\gamma_{G_a} \leq \sqrt{2a} \max\{\gamma_1, \gamma_2\}, \quad \forall a \in [0,1]. \tag{3.27}$$

■

Proof of Lemma 3.4.1. First let

$$\begin{aligned} g_1 &:= B\tilde{u}, \\ g_2 &:= B\tilde{u} + \tilde{w} \end{aligned}$$

be two inputs to system (3.25) and z_1 and z_2 be the corresponding outputs. $\gamma_1 < \infty$ implies that for the system

$$\begin{aligned} \dot{x_1} &= Ax_1 + \Phi(x_1) + B\tilde{u} \\ \dot{x_2} &= Ax_2 + \Phi(x_2) + B\tilde{u} + \tilde{w} \\ \tilde{e} &= C(x_1 - x_2) \end{aligned}$$

the following integral inequality holds

$$\inf_{T \geq 0} \int_0^T \left(\gamma_1^2 \|B\tilde{u} - B\tilde{u} - \tilde{w}\|_2^2 - \|\tilde{e}\|_2^2 \right) d\tau > -\infty,$$

which implies, $\forall a > 0$,

$$\inf_{T\geq 0}\int_0^T \left(a\gamma_1^2\left(\|\tilde{u}\|_2^2+\frac{1}{a}\|\tilde{w}\|_2^2\right)-\|\tilde{e}\|_2^2\right)d\tau > -\infty,$$

or

$$\inf_{T\geq 0}\int_0^T \left(a\gamma_1^2\left(\|u\|_2^2+\|w\|_2^2\right)-\|e\|_2^2\right)d\tau > -\infty, \tag{3.28}$$

when $u=\tilde{u}$, $w=\frac{1}{\sqrt{a}}\tilde{w}$, and $e=\tilde{e}$. That shows that the system

$$\begin{aligned}\dot{x_1} &= Ax_1+\Phi(x_1)+Bu\\ \dot{x_2} &= Ax_2+\Phi(x_2)+Bu+\sqrt{a}w\\ e &= C(x_1-x_2)\end{aligned}$$

has L2 gain from $[u;w]$ to e less than or equal to $\sqrt{a}\gamma_1$. This means that system G_a (3.23) has L2 gain from $[u;w]$ to e is less than or equal to $\sqrt{a}\gamma_1$.

Secondly, for system (3.25), let $g=B\tilde{u}+\tilde{w}$. Then $\gamma_2<\infty$ implies in the following system

$$\begin{aligned}\dot{x_1} &= Ax_1+\Phi(x_1)+B\tilde{u}\\ \dot{x_2} &= Ax_2+\Phi(x_2)+B\tilde{u}+\tilde{w}\\ \tilde{y} &= x_2\end{aligned}$$

the following inequality holds

$$\inf_{T\geq 0}\int_0^T \left(\gamma_2^2\left(\|\tilde{u}\|_2^2+\|\tilde{w}\|_2^2\right)-\|\tilde{y}\|_2^2\right)d\tau > -\infty,$$

which implies, $\forall\, a\in(0,1]$,

$$\inf_{T\geq 0}\int_0^T \left(a\gamma_2^2\left(\|\tilde{u}\|_2^2+\frac{1}{a}\|\tilde{w}\|_2^2\right)-a\|\tilde{y}\|_2^2\right)d\tau > -\infty, \tag{3.29}$$

Note that the fact that $\frac{1}{a}\geq 1$ for $a\leq 1$ was indeed used. Rewrite the signals in eq. (3.29) in terms of the signals in eq. (3.23). That is, $u=\tilde{u}$, $w=\frac{1}{\sqrt{a}}\tilde{w}$ and $y=\sqrt{a}\tilde{y}$. This results in the

following inequality:

$$\inf_{T\geq 0}\int_0^T \left(a\gamma_2^2\left(\|u\|_2^2+\|w\|_2^2\right)-\|y\|_2^2\right)d\tau > -\infty, \tag{3.30}$$

which means that the L2 gain of system G_a in eq. (3.23) from $[u;w]$ to y has L2 gain less than or equal to $\sqrt{a}\gamma_2$.

Eq. (3.28) together with eq. (3.30) implies that, in terms of the quantities associated with G_a in (3.23), the following integral

$$\inf_{T\geq 0}\int_0^T \left(2a(\max\{\gamma_1,\gamma_2\})^2\left(\|w\|_2^2+\|u\|_2^2\right)-(\|y\|_2^2+\|e\|_2^2)\right)d\tau \tag{3.31}$$

is bounded from below for all input/output pair of G_a and this proves eq. (3.27) $\forall\, a\in(0,1]$. For the case of $a=0$, $\gamma_1<\infty$ implies $\gamma_{G_a}|_{a=0}=0$, so eq. (3.27) also holds in this case. ■

Remark 3.4.2. Lemma 3.4.1 suggests that

$$\lim_{a\to 0}\frac{\gamma_{G_a}}{a^\beta}<\infty, \tag{3.32}$$

with $\beta=0.5$. In fact, the value of $\beta=0.5$ is the largest possible exponent such that the limit in eq. (3.32) is still finite. To see this, consider the LTI case where G_a can be given as a transfer matrix

$$\sqrt{a}\begin{bmatrix}\sqrt{a}G_{11} & G_{12}\\ G_{21} & 0\end{bmatrix},$$

where the "G_{22}" block is zero because the transfer matrix from u to e is zero. Then the limit in eq. (3.32) holds, that is,

$$a^{0.5-\beta}\left\|\begin{bmatrix}\sqrt{a}G_{11} & G_{12}\\ G_{21} & 0\end{bmatrix}\right\|_\infty<\infty$$

if and only if $\beta\leq 0.5$. Since eq. (3.32) must be satisfied by all systems including the LTI ones, 0.5 is the upper bound for the value of β such that eq. (3.32) still holds. ■

3.4.2 The linear error bound in the limit

Using Lemma 3.4.1, the main result is now presented.

Theorem 3.4.3. *Let γ_1 and γ_2 be the quantities defined in (3.26). Also let γ_Δ be the L2 gain of $\Phi - \tilde{\Phi}$ in (3.22). That is,*

$$\gamma_\Delta := \sup_{v \neq 0} \frac{|\Phi(v) - \tilde{\Phi}(v)|}{|v|}$$

Denote γ as the L2 gain from u to e in system (3.21).

If $\gamma_1 < \infty$, $\gamma_2 < \infty$ and $\sqrt{2}\max\{\gamma_1, \gamma_2\}\gamma_\Delta \leq 1$, then

$$\gamma \leq 2\left(\max\{\gamma_1, \gamma_2\}\right)^2 \gamma_\Delta. \tag{3.33}$$

■

Proof of Theorem 3.4.3. If $\gamma_\Delta = 0$, then by the finiteness of γ_1, $\gamma = 0$ and hence (3.33) holds because system (3.21) reduces to the difference of two identical systems. Now consider the case when $\gamma_\Delta > 0$, the small gain theorem states that

$$\gamma \leq \gamma_{G_a}, \quad \forall a : \frac{\gamma_{G_a}\gamma_\Delta}{a} \leq 1.$$

Therefore,

$$\gamma \leq \min_{a: \frac{\gamma_{G_a}\gamma_\Delta}{a} \leq 1} \gamma_{G_a}. \tag{3.34}$$

Denote $c := \sqrt{2}\max\{\gamma_1, \gamma_2\}$. Since $\gamma_1 < \infty$ and $\gamma_2 < \infty$ by statement assumption, Lemma 3.4.1 states that $\forall\, a \in (0, 1]$,

$$\gamma_{G_a} \leq c\sqrt{a} \quad \text{and hence} \quad \frac{\gamma_{G_a}\gamma_\Delta}{a} \leq \frac{c\gamma_\Delta}{\sqrt{a}}.$$

Since $c\gamma_\Delta \leq 1$ by statement assumption, the set $[c\gamma_\Delta, 1] \neq \emptyset$. $\exists\, a \in [c\gamma_\Delta, 1]$:

$$(c\gamma_\Delta)^2 \leq a \leq 1$$

and hence

$$1 \geq \frac{c\gamma_\Delta}{\sqrt{a}} \geq \frac{\gamma_{G_a}\gamma_\Delta}{a}.$$

Therefore,

$$\gamma \leq \min_{a:\frac{\gamma_{G_a}\gamma_\Delta}{a}<1} \gamma_{G_a} \leq \min_{a \geq c^2\gamma_\Delta^2} c\sqrt{a} = c^2\gamma_\Delta.$$

■

Remark 3.4.4. Intuitively, Theorem 3.4.3 asserts that if γ_Δ, the L2 gain of the difference $\Phi - \tilde{\Phi}$ (and also $\phi - \tilde{\phi}$) is *sufficiently small*, then the approximation quality in terms of the L2 gain of the error system (3.21) is also small. In particular, it provides a guideline for designing the approximation system (3.2). It states that searching for a $\tilde{\phi}$ that is close to ϕ in L2 gain sense, should be a reasonable choice, as opposed to other methods such as Taylor Series, for which the accuracy has not been rigorously established. In addition, the linear error bound (3.34) can be used to guide the design of the vector field approximation in the following sense:

- Pick a desired system error ε.
- Choose any available vector field approximation technique (not discussed in this thesis).
- Obtain an approximated reduced system; compute the vector field L2 gain error, and the difference system L2 gain, denoted as ε_Δ and ε_1 respectively.
- If $\varepsilon_1 < \varepsilon$ then the desired approximated reduced system has already been obtained. Otherwise, obtain a better approximated system (e.g., by increasing polynomial order) so that the new vector field L2 gain error is less than $\frac{\varepsilon_\Delta \varepsilon}{\varepsilon_1}$, then under the assumptions of Theorem 3.4.3, the new reduced model will satisfy the desired system error tolerance. ■

Remark 3.4.5. However, it should also be noted that Theorem 3.4.3 can be conservative and eq. (3.33) is not true for γ_Δ that is not small enough. Therefore, it would be interesting to see if there exists a less restrictive statement or a numerical procedure to compute a tighter bound. The result in the next section is an attempt to do so. ■

3.5 A Numerical Error Bound with IQC

Theorem 3.3.2 in Subsection 3.3.3 was applied in Section 3.4 via Lemma 3.4.1 and Theorem 3.4.3, which provides some theoretical insight into the solution of the error bounding problem in Definition 3.3.1. However, the practical use of Theorem 3.4.3 is limited because the coefficients in eq. (3.33) can be too conservative.

In this section, on the other hand, a numerical procedure, based on the IQC analysis described in Subsection 3.2.5, is proposed to apply Theorem 3.3.2 by directly computing an upper bound of the L2 gain of γ_{G_a} in the theorem. The procedure is summarized as follows.

3.5.1 The numerical procedure

The proposed numerical procedure is as follows.

- For a discrete set of $\{a_1, a_2, \ldots\}$ (e.g., $a_k := 10^{-k}$), use IQC analysis to find $\gamma_1, \gamma_2, \ldots$ as the L2 gain upper bounds for the parameterized systems $G_{a_1}, G_{a_2}, \ldots$

- For any approximation vector field $\tilde{\Phi}$, evaluate the L2 gain of $\Phi - \tilde{\Phi}$. Denote it as γ_Δ. Find the index i such that

$$i = \operatorname*{argmin}_k a_k : \frac{\gamma_k \gamma_\Delta}{a_k} < 1$$

- γ_i is returned as the upper bound of the L2 gain of the difference system (3.21).

Since the order of system G_a (3.23) is $2q$ and q is assumed to be small, solving the LMIs to obtain L2 gain upper bounds γ_k for all a_k is relatively cheap. Once the L2 gain upper bounds $\gamma_1, \gamma_2, \ldots$ have been found, the numerical procedure requires a trivial amount of time to analyze the system L2 gain error for all $\tilde{\Phi}$ such that γ_Δ is small enough. As a final note, it should be pointed out that since the numerical procedure is based on the small gain theorem, it is possible that when γ_Δ is large, the procedure fails to return any conclusive result.

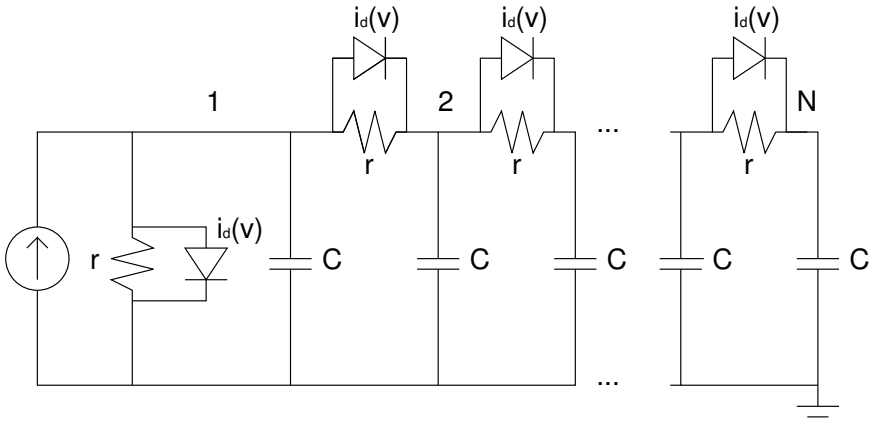


Figure 3-4: A transmission line with diodes.

3.6 Numerical Experiment

In this section the numerical procedure described in Section 3.5 is applied to analyze the L2 gain of the difference system due to approximation of the nonlinear vector field. The specific application example is a transmission line with diodes described in [83] and shown in Figure 3-4. Using nodal analysis, the model of the diode line has the form

$$\begin{aligned} \dot{x_f} &= A_f x_f - M'\Phi(Mx_f) + B_f u \\ y_f &= C_f x_f, \end{aligned}$$

with

$$\begin{aligned} &A_f \in \mathbb{R}^{N\times N}, M \in \mathbb{R}^{N\times N}, B_f \in \mathbb{R}^{N\times 1}, C_f \in \mathbb{R}^{1\times N}, \\ &\Phi(v) = \mathrm{diag}\,(\phi(v_1), \phi(v_2), \ldots) \quad \text{and} \quad \phi(v_k) = e^{-v_k} - 1, \end{aligned}$$

with M being a sparse matrix relating branch voltages to node voltages. Suppose there exists a projection matrix $V \in \mathbb{R}^{N\times q}$ (e.g., dominant singular vectors of some matrix stacked by columns of trajectories), then the reduced model is

$$\begin{aligned} \dot{x} &= A_r x - V_r'\Phi(V_r x) + B_r u \\ y &= C_r x, \end{aligned} \tag{3.35}$$

with $A_r = V'A_fV$, $V_r = MV$, $B_r = V'B$ and $C_r = CV$. System (3.35) is of the form of (3.1), hence the numerical procedure described in Section 3.5 can be applied. γ_{G_a} and $\frac{a}{\gamma_{G_a}}$ are

plotted in Figure 3-5 for a range of values of a.

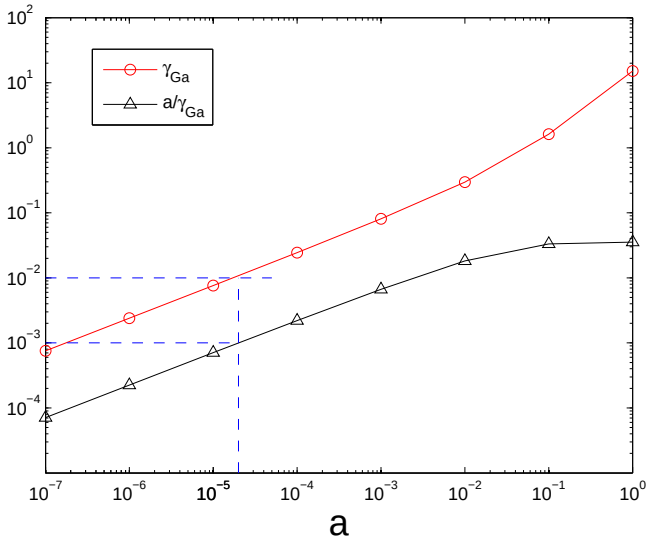


Figure 3-5: Transmission line example. The upper line (circles) is the numerical upper bound for the L2 gain of the difference system. The lower line (triangles) is the minimum allowable a such that $\frac{\gamma_\Delta \gamma_{G_a}}{a} < 1$, and hence the small gain theorem still applies. For instance, if we want the system L2 gain error to be less than 10^{-2}, then a should be at most 2×10^{-5}, corresponding to a maximum allowable vector field error γ_Δ of about 10^{-3}.

In this figure, the upper line (circles) is γ_{G_a} that is used as the upper bound for the L2 gain of difference system (3.21). On the other hand, the lower line (triangles) is the quantity $\frac{a}{\gamma_{G_a}}$ used in determining the minimum a, for a specific γ_Δ, such that $\frac{\gamma_\Delta \gamma_{G_a}}{a} \leq 1$ (hence the small gain theorem applies).

As an example to illustrate how Figure 3-5 can be applied, let the desired system level error be 1% or less. By the small gain theorem, if $\gamma_{G_a} < 1\%$ then the accuracy is achieved. According to Figure 3-5, the maximum allowable a for the small gain theorem to be applicable is about 2×10^{-5} (the x coordinate where horizontal $y = 10^{-2}$ intersects the upper line). For $a = 2 \times 10^{-5}$, the corresponding value of $\frac{a}{\gamma_{G_a}}$ is about 10^{-3}, which means that the vector field L2 gain error γ_Δ should be at most 10^{-3}.

3.7 Conclusion

This chapter investigated the estimation of the L2 gain system error produced by the approximation of the nonlinear vector field within any nonlinear model order reduction algorithm for systems in the form of (3.1). This problem was formulated as an L2 gain upper bounding problem of a feedback interconnection of a "nominal" plant and a "disturbance" (i.e., vector field error). The chapter proposed a framework for broadening the use of the small gain theorem by introducing the mutually cancelling gains $\sqrt{a}$ and $\frac{1}{\sqrt{a}}$ in the feedback loop. While this modification failed exactly when the small gain theorem failed to apply, it was nevertheless able to tighten the L2 gain upper bound (by the use of γ_{G_a}), and the bound was asymptotically tight. Based on the scaled feedback setup, we have shown that the difference system L2 gain γ was upper bounded by a linear function of the vector field difference L2 gain γ_Δ, provided γ_Δ was sufficiently small. In an attempt to fight the conservatism of the bound, this thesis also proposed a numerical procedure that combined IQC/LMI techniques and small gain theorem. Although the numerical procedure still did not apply for large errors in the vector field, it did produce a more readily computable bound than the theoretical linear bound. Finally, a numerical example was given to demonstrate the use of our numerical procedure.

Chapter 4

A Convex Relaxation Approach to the Identification of the Wiener-Hammerstein Model

4.1 Introduction

Efficient hierarchical system level design and optimization could be facilitated by the availability of automatic and accurate behavioral modeling tools for system blocks such as nonlinear circuits (e.g. operational amplifiers) or nonlinear devices (e.g. MEMs). In the current state of the art, analog designers and system architects generate analytical or semi-empirical behavioral models of their blocks using their intuition and expertise formed on thousands of hours spent running slow circuit simulators such as SPICE, or even slower Partial Differential Equation (PDE) field solvers. Most of the efforts in the field of automatic and accurate modeling of nonlinear system blocks involve development of techniques for efficiently and accurately *reducing* available large nonlinear systems generated by circuit schematics and parasitic extractors [85, 83, 89, 90, 86]. When only input/output physical measurements are available for a given circuits or systems, system identification may be the only valuable option. Furthermore, even when internal circuit schematics are available, or when the internal information of PDE solvers used to simulate MEMS is accessible, system identifi-

cation may still represent a both efficient and powerful alternative method to model order reduction. For instance, the authors of [39, 40] presented comprehensive surveys of the use of system identification for power-amplifier related modeling.

The theory for linear time-invariant (LTI) system identification is relatively mature and complete [91]. On the other hand, the practice of nonlinear system identification tends to be case dependent [37, 38]. Volterra series [92] is a general approach, and it has been very popular among engineers working on behaviorial modeling (e.g., [93, 94]). In this chapter, only a specific class of nonlinear system identification problem will be considered – the identification of the Wiener-Hammerstein system with feedback. Classical treatments of the Wiener-Hammerstein system identification problem can be found, for example, in [91, 95]. Many more recent treatments of the problem can be found, for example, in [45, 46, 47]. In those references, however, the identification of the nonlinearity is parametric (i.e., the nonlinearity is assumed to be of some form such as piecewise linear or polynomial functions). Therefore, those previous results can be restrictive in application. Non-parametric identification of block oriented models, on the other hand, are more flexible in terms of modeling power. Reference [96] proposed an algorithm for the non-parametric identification of the Wiener system under the assumption that the input is Gaussian noise. The authors of [97], assuming that the LTI block is known, reduced the identification problem of the Wiener system to a least squares problem. [98] proposed an unbiased identification algorithm based on maximum likelihood estimation.

In a sense, the idea of the system identification scheme proposed in this chapter has been explored under the banner of model validation [99, 100, 101, 102, 103, 104, 105]. In this problem, a model with a given block diagram is to be invalidated by proving that it is inconsistent with some input/output measurement obtained from experiment. The invalidation is typically performed through the finding of some infeasibility certificate of some constraint set. Conversely, the finding of a feasibility certificate will prove the consistency of a model with the given input/output measurement data. This forms the basis of the block diagram oriented system identification schemes such as [106, 107, 108]. In particular, [108] proposed a very general approach for the identification of the Wiener system assuming only the monotonicity of the nonlinearity. [108] set up a convex QP based on the

idea of enforcing input/output functional relationship of the nonlinearity. The algorithm proposed in this chapter can be considered as an extension of the idea in [108]. In fact, the formulation of the optimization problem in this chapter also centers around some sector bound property of the nonlinearity. However, because of the more complicated Wiener-Hammerstein structure, the resultant optimization problem is more involved. In fact, it is a non-convex QP. Nevertheless, with the proposed SDP relaxation, it will be demonstrated that the non-convex QP formulated in this chapter is not necessarily hard to solve.

The rest of the chapter is organized as follows: in Section 4.2 some technical background and definitions will be given. The main ideas of the problem formulation and solution procedure, explained in Section 4.3 and Section 4.4 respectively, will be given through a special setup in which there is no output measurement noise or feedback. Then in Section 4.5 the identification setup with output measurement noise is considered. Differences in the analysis and algorithm due to the noise will be highlighted. After that, the full feedback Wiener-Hammerstein system identification problem will be considered in Section 4.6. Finally, in Section 4.7 a brief account of the complexity of the proposed algorithm will be given, and application examples will be presented in Section 4.8. Table 4.1 summarizes the development of the proposed system identification algorithm.

Table 4.1: The organization of Chapter 4

	no noise	with noise
no feedback	Sec 4.3 – 4.4	Sec 4.5
with feedback	–	Sec 4.6

4.2 Technical Background and Definitions

4.2.1 System and model

In this chapter, a **system** is a function which maps its input signal to its output signal. On the other hand, the term **model** can have two meanings: a **model** can mean 1) a collection of parameterized systems usually of some specific form, or 2) a specific instance of the

collection defined in 1). For example, for the finite impulse response (FIR) transfer function fitting problem, the unknown system can be $\frac{1+z^{-1}}{2-z^{-1}}$, whereas the model is of the form $a_0 + a_1 z^{-1} + \ldots + a_n z^{-n}$ for arbitrary $a_k \in \mathbb{R}$. On the other hand, an instance such as $1 + 2z^{-1} + \ldots + (n+1)z^{-n}$ is also called a model. In the subsequent discussions, the meaning of the term "model" should be obvious from the context. The definitions of the terms system and model will allow us in this chapter to distinguish the fixed (but unknown) input/output relationship (i.e., the system) from the one that is to be determined by the identification algorithm (i.e., the model).

4.2.2 Input/output system identification problem

Definition 4.2.1. *The input/output system identification problem considered in this chapter is as follows: given the input/output measurement pairs of an unknown dynamical system, find a stable model such that the given input/output measurement pairs satisfy the input/output relationship of the model.* ■

Remark 4.2.2. Contrary to many other problems which seek to ensure the "generalization capacity" of the solutions (e.g., variance minimization in statistical modeling), the solution criterion of Definition 4.2.1 is based entirely on the matching of the given problem data. It is assumed that the given problem data covers all the dynamics of interest. ■

Remark 4.2.3. System identification problems in the subsequent sections will be defined according to Definition 4.2.1. ■

4.2.3 Feedback Wiener-Hammerstein system

In this thesis, the unknown system in the input/output system identification problem described in Subsection 4.2.2 is assumed to be from a specific class – either of the Wiener-Hammerstein form, or the Wiener-Hammerstein with feedback in Figure 4-1.

The following notations in Figure 4-1 will be used throughout the chapter:

- The input of the unknown system is denoted as u. This is part of the problem data.

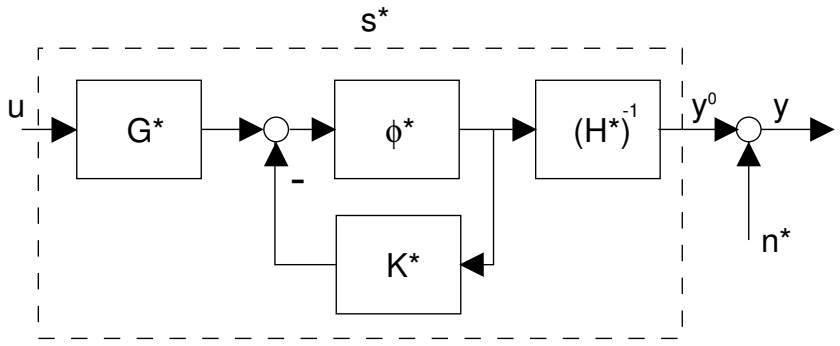


Figure 4-1: The Wiener-Hammerstein system with feedback. S^* denotes the unknown system. $K \equiv 0$ corresponds to the Wiener-Hammerstein system without feedback. The output measurement y is assumed to be corrupted by some noise n^*.

- The output *measurement* of the unknown system is denoted as y. This is part of the problem data.
- The true output of the unknown system is denoted as y^0. This is not available to the system identification process.
- The output measurement noise denoted as n^*. The output measurement noise is additive. That is,

$$y[t] = y^0[t] + n^*[t], \quad \forall t. \tag{4.1}$$

The following assumptions are made in Figure 4-1.

1. The signals u, y, y^0 and n^* are one-sided and of finite length N. For example,

$$u[t] = \begin{cases} u_t & \text{if } t = 0,1,\ldots,N-1, \\ 0 & \text{otherwise} \end{cases}.$$

2. G^*, H^* and K^* are assumed to be single-input-single-output (SISO) FIR systems. In addition, H^* and K^* are assumed to be positive-real passive. That is,

$$\begin{aligned} &\mathrm{Re}\left\{H^*\left(e^{j\omega}\right)\right\} > 0, \quad \forall \omega \in [0, 2\pi) \\ &\mathrm{Re}\left\{K^*\left(e^{j\omega}\right)\right\} > 0, \quad \forall \omega \in [0, 2\pi) \end{aligned} \tag{4.2}$$

3. Nonlinearity ϕ^* is assumed to be scalar valued, memoryless, and is assumed to satisfy certain sector bound criterion in incremental sense. That is, there exists a scalar $0 < \beta < \infty$ such that

$$\Big(\phi^*(b) - \phi^*(a)\Big)\Big(\phi^*(b) - \phi^*(a) - \beta b + \beta a\Big) \leq 0, \quad \forall a, b \in \mathbb{R}. \tag{4.3}$$

Practically speaking condition (4.3) means that the nonlinearity ϕ^* is monotonically non-decreasing and its derivative has an upper bound. This is summarized by the following lemma.

Lemma 4.2.4. *Let $\phi^* : \mathbb{R} \mapsto \mathbb{R}$ and $\beta > 0$, then condition (1) and (2) in the following are equivalent.*

$$(1) \quad \Big(\phi^*(b) - \phi^*(a)\Big)\Big(\phi^*(b) - \phi^*(a) - \beta b + \beta a\Big) \leq 0, \quad \forall a, b \in \mathbb{R}$$

$$(2) \quad \begin{cases} \Big(\phi^*(b) - \phi^*(a)\Big)(b-a) \geq 0, \quad \forall a, b \in \mathbb{R} \quad \textit{(monotonicity)} \\ \Big(\phi^*(b) - \phi^*(a)\Big)^2 \leq \beta^2 (b-a)^2, \quad \forall a, b \in \mathbb{R} \quad \textit{(derivative bound)}. \end{cases}$$

■

Proof of Lemma 4.2.4. Denote the set $\mathcal{E} := \big\{(v_1, v_2) \in \mathbb{R}^2 \big| \, v_1 = v_2 \text{ or } \phi^*(v_1) = \phi^*(v_2)\big\}$. Then the statement is trivially true if $(a,b) \in \mathcal{E}$. Therefore, it will be assumed for the rest of the proof that $(a,b) \in \big(\mathbb{R}^2 \setminus \mathcal{E}\big) := \mathcal{E}^c$.

First we show the direction "(1) $\Rightarrow$ (2)". Note that (1) implies

$$(\phi^*(b) - \phi^*(a))(b-a) \geq \frac{1}{\beta}(\phi^*(b) - \phi^*(a))^2 > 0, \quad \forall\, (a,b) \in \mathcal{E}^c, \tag{4.4}$$

hence showing the first statement of (2). Then, dividing by $(\phi^*(b) - \phi^*(a))(b-a)$ and multiplying with β, eq. (4.4) becomes

$$\beta \geq \frac{\phi^*(b) - \phi^*(a)}{b-a} \geq 0, \quad \forall\, (a,b) \in \mathcal{E}^c. \tag{4.5}$$

Squaring both sides of eq. (4.5) yields the second statement of (2).

Now we show the direction "(2) $\Rightarrow$ (1)". Dividing the first statement of (2) by $(b-a)^2$ yields

$$\frac{\phi^*(b)-\phi^*(a)}{b-a} \geq 0, \quad \forall\, (a,b) \in \mathcal{E}^c. \tag{4.6}$$

On the other hand, the second statement of (2) implies that

$$\frac{(\phi(b)-\phi(a))^2}{(b-a)^2} \leq \beta^2, \quad \forall\, (a,b) \in \mathcal{E}^c. \tag{4.7}$$

Eq. (4.6) allows the squared root of eq. (4.7) to hold, resulting in

$$\frac{\phi(b)-\phi(a)}{b-a} \leq \beta, \quad \forall\, (a,b) \in \mathcal{E}^c. \tag{4.8}$$

Since $(\phi^*(b)-\phi^*(a))\,(b-a) \geq 0$ by the first statement of (2), multiplying both sides of eq. (4.8) with $(\phi^*(b)-\phi^*(a))\,(b-a)$ yields (1), thus concluding the proof. ■

Remark 4.2.5. The derivative bound in Lemma 4.2.4 does not result in much loss of generality because any physical system is supposed to have a finite gain. The monotonicity assumption, however, is made due to stability concerns: together with the positive-real assumption in eq. (4.2), the system in Figure 4-1 can be shown to be stable using the circle criterion (see [109] Chapter 4). ■

4.2.4 Non-parametric identification of nonlinearity

Typically, the identification of a scalar memoryless nonlinearity can be done in two ways: parametric and non-parametric. **Parametric** identification means that the to-be-determined nonlinearity is assumed to be of some pre-defined form which carries some to-be-determined parameters. A very popular class of the pre-defined forms is the linear combination of some basis functions, with polynomials and piecewise polynomials being some popular choices. A more extensive treatment of the topic of parametric identification can be found in the field of machine learning. See, for example, [110, 111] for more details. **Non-parametric** identification, on the other hand, does not assume any form of the to-be-determined nonlinearity. Instead, the nonlinearity is specified through a lookup table of the samples of

its input and output. Values of the nonlinearity not specified in the lookup table are typically obtained using some interpolation schemes such as splines [112]. The particular type of interpolation scheme chosen in this thesis is linear interpolation. That is, let (v_k, w_k), $k = 1, 2, \ldots, N$ be the lookup table of the nonlinearity ϕ. Without loss of generality, assume $v_1 < v_2 < \ldots < v_N$. Then the nonlinearity ϕ is defined as

$$\phi(v) = \begin{cases} w_k & \text{if } v = v_k, \quad \text{for some } k, \\ w_i + \frac{w_{i+1}-w_i}{v_{i+1}-v_i}(v - v_i) & \text{if } v \neq v_k, \quad \text{for all } k \text{ and } \exists\, i : v_i < v < v_{i+1}, \\ w_N + \frac{w_N - w_{N-1}}{v_N - v_{N-1}}(v - v_N) & \text{if } v \neq v_k, \quad \text{for all } k \text{ and } v > v_N, \\ w_1 + \frac{w_2 - w_1}{v_2 - v_1}(v - v_1) & \text{if } v \neq v_k, \quad \text{for all } k \text{ and } v < v_1. \end{cases} \tag{4.9}$$

In general, when the samples given in the lookup table are dense enough, the linear interpolation scheme in eq. (4.9) is sufficient to provide an accurate characterization of the nonlinearity ϕ. An added benefit of the linear interpolation scheme is that if ϕ satisfies the sector bound eq. (4.3) at v_k (specifying the lookup table), then it satisfies the sector bound for all values of its input argument as well (see Lemma 4.3.1 in Subsection 4.3.1).

4.3 Identification of Wiener-Hammerstein System – No Measurement Noise

The first problem to be considered in this chapter is the identification of the Wiener-Hammerstein system without the feedback or the output measurement noise. The identification problem will be formulated as two equivalent optimization problems in Subsections 4.3.1 and 4.3.3 respectively. The solution technique for the optimization problems will be described in Section 4.4.

4.3.1 System identification problem formulation

4.3.1 A: Problem data

The problem data is the input signal u and the output measurement signal y of the true (but unknown) system S^* in Figure 4-1. For ease of exposition, a signal will also be denoted as the vector of its non-zero values. For example,

$$\mathbf{u} := \begin{bmatrix} u[0] & u[1] & \dots & u[N-1] \end{bmatrix}'.$$

The symbol $(\mathbf{u}, \mathbf{y})$ will denote a pair of corresponding input and output measurement. In a realistic system identification setup, there are more than one pair of $(\mathbf{u}, \mathbf{y})$. However, for simplicity, this chapter will only deal with the case with only one pair. Nevertheless, the technique introduced in this chapter can be extended to the general case.

4.3.1 B: System identification model and decision variables

It is natural to choose a model with the same structure as the true but unknown system (i.e., the Wiener-Hammerstein structure in Figure 4-2). In the model in Figure 4-2 the G and H are FIR systems, and ϕ is a scalar memoryless nonlinearity (i.e., a nonlinear function). Obviously, the model is specified when G, H and ϕ are specified.

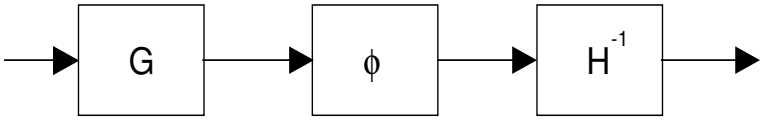


Figure 4-2: The Wiener-Hammerstein model – G and H are FIR systems, and ϕ is a scalar memoryless nonlinearity. The last block is chosen to be H^{-1} for computation reasons.

FIR systems G and H are characterized by their impulse responses of length N_g and N_h respectively. That is,

$$\begin{aligned} \mathbf{g} &:= \begin{bmatrix} g_0 & g_1 & \dots & g_{N_g-1} \end{bmatrix}', \\ \mathbf{h} &:= \begin{bmatrix} h_0 & h_1 & \dots & h_{N_h-1} \end{bmatrix}', \end{aligned} \tag{4.10}$$

and the corresponding transfer functions of G and H are

$$\begin{aligned} G(z) &= g_0 + g_1 z^{-1} + \ldots + g_{N_g-1} z^{-(N_g-1)}, \\ H(z) &= h_0 + h_1 z^{-1} + \ldots + h_{N_h-1} z^{-(N_h-1)}. \end{aligned} \tag{4.11}$$

The identification of the nonlinearity ϕ is non-parametric. That is, ϕ is specified only by some samples of its input/output pair. The values of ϕ other than those given by the samples can be obtained using an interpolation scheme (e.g., eq. (4.9) in Subsection 4.2.3). In addition, the samples will be restricted to those computable by the FIR impulse response **g** and **h**. Therefore, **g** and **h** are the decision variables sufficient to specify ϕ as well as the full model in Figure 4-2.

4.3.1 C: Treatment of the passivity constraint

In order to be a candidate solution of the system identification problem according to Definition 4.2.1 in Subsection 4.2.2, the model in Figure 4-2 must be stable.

A sufficient condition for stability is that the FIR system H in Figure 4-2 is positive real passive. That is,

$$\mathrm{Re}\left\{H\left(e^{j\omega}\right)\right\} = h_0 + h_1\cos(\omega) + \ldots + h_{N_h-1}\cos\left((N_h-1)\omega\right) > 0, \quad \forall\,\omega \in [0, 2\pi)\,. \tag{4.12}$$

Then H^{-1} will also be positive real passive, and then the "feedback loop" of H^{-1} and the monotonic nonlinearity of a zero function will be stable by the circle criterion (see [109], Chapter 3). Consequently, the entire model in Figure 4-2 will be stable.

Ideally the positive real constraint in eq. (4.12) should be enforced. However, constraint eq. (4.12) turns out to be inconsistent with the solution technique proposed. Therefore, in all subsequent sections the stability requirement will not be dealt with explicitly. In Subsection 4.4.3 this issue will be revisited, and a post-processing algorithm will be given to enforce the passivity of H (and hence the stable of the final model).

4.3.1 D: System identification problem formulation – a feasibility problem

The only requirement left for a model to become a solution to the system identification problem according to Definition 4.2.1 is that the input/output measurement $(\mathbf{u},\mathbf{y})$ is satisfied by the model. The satisfiability problem is formulated as a feasibility problem in the following sense. Consider the Wiener-Hammerstein model in Figure 4-3 in which the output and the input are constrained to be the given data $(\mathbf{u},\mathbf{y})$. Let's investigate the possible the choices of the decision variables $\mathbf{g}$ and $\mathbf{h}$ so that there exist signals $\mathbf{v} \in \mathbb{R}^N$ and $\mathbf{w} \in \mathbb{R}^N$ with the property that $(\mathbf{u},\mathbf{v})$, $(\mathbf{v},\mathbf{w})$, $(\mathbf{y},\mathbf{w})$ are valid input/output pairs of the blocks G, ϕ and H respectively.

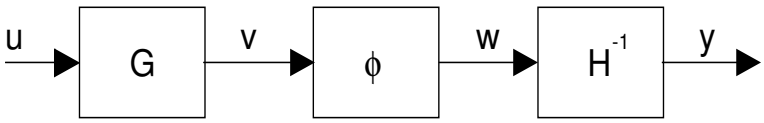


Figure 4-3: A feasibility problem to determine the impulse responses of the FIR systems G and H. Here $\mathbf{u}$ and $\mathbf{y}$ are the given input and output measurements generated by the true (but unknown) system. The signals $\mathbf{v}$ and $\mathbf{w}$ are the outputs of G and H, respectively. $\mathbf{v}$ and $\mathbf{w}$ are chosen so that they define a function ϕ satisfying sector bound constraint eq. (4.16).

The pairs $(\mathbf{u},\mathbf{v})$ and $(\mathbf{y},\mathbf{w})$ satisfy the following convolution relationship.

$$\begin{aligned} \mathbf{v} &= \mathbf{U}\mathbf{g}, \\ \mathbf{w} &= \mathbf{Y}\mathbf{h}, \end{aligned} \tag{4.13}$$

where $\mathbf{U} \in \mathbb{R}^{N\times N_g}$ and $\mathbf{Y} \in \mathbb{R}^{N\times N_h}$ are defined as

$$\mathbf{U} := \begin{bmatrix} u[0] & 0 & \dots & 0 \\ u[1] & u[0] & \ddots & \vdots \\ & & \ddots & 0 \\ \vdots & \vdots & & u[0] \\ & & & \vdots \\ u[N-1] & u[N-2] & \dots & u[N-N_g] \end{bmatrix}_{N\times N_g}, \tag{4.14}$$

$$\mathbf{Y} := \begin{bmatrix} y[0] & 0 & \dots & 0 \\ y[1] & y[0] & \ddots & \vdots \\ & & \ddots & 0 \\ \vdots & \vdots & & y[0] \\ & & & \vdots \\ y[N-1] & y[N-2] & \dots & y[N-N_h] \end{bmatrix}_{N\times N_h} . \tag{4.15}$$

For the pair $(\mathbf{v},\mathbf{w})$, in principle, the only constraint imposed is that there exists some function ϕ such that $\mathbf{w}_i = \phi(\mathbf{v}_i)$, $\forall i = 0,1,\dots,N-1$. However, to maximally reduce the redundancy of the possible choices of $(\mathbf{v},\mathbf{w})$, the addition constraint is enforced that ϕ should satisfy the sector bound of the form of eq. (4.3). That is,

$$(\phi(b)-\phi(a))\,(\phi(b)-\phi(a)-\beta b+\beta a) \le 0, \quad \forall a,b \in \mathbb{R}. \tag{4.16}$$

Constraint eq. (4.16) imposed on the function $\phi : \mathbb{R} \mapsto \mathbb{R}$ is equivalent to a constraint on the generating pair $(\mathbf{v},\mathbf{w})$ as

$$\left(\mathbf{w}_i-\mathbf{w}_j\right)\left(\mathbf{w}_i-\mathbf{w}_j-\beta\mathbf{v}_i+\beta\mathbf{v}_j\right) \le 0, \quad \forall N-1 \ge i > j \ge 0. \tag{4.17}$$

The following lemma certifies the equivalence.

Lemma 4.3.1. *Let $(\mathbf{v},\mathbf{w}) \in \mathbb{R}^N \times \mathbb{R}^N$, then there exists a function $\phi : \mathbb{R} \mapsto \mathbb{R}$ such that*

1. $\phi(v_k) = w_k, \quad \forall k = 0,1,\dots,N-1.$
2. *Constraint eq. (4.16) is satisfied by ϕ.*

if and only if constraint eq. (4.17) is satisfied by $(\mathbf{v},\mathbf{w})$. ■

Proof of Lemma 4.3.1. The "only if" part is trivially shown by applying statement 1 to constraint eq. (4.16), which is assumed true by statement 2.

For the "if" part, first notice that eq. (4.17) implies that $\mathbf{w}_i = \mathbf{w}_j$ if $\mathbf{v}_i = \mathbf{v}_j$. Therefore, it can be assumed that the entries of $\mathbf{v}$ are unique (i.e., $\mathbf{v}_i \neq \mathbf{v}_j$ if $i \neq j$). In addition, let $\tilde{\mathbf{v}}$ be a sorted version of $\mathbf{v}$ (i.e., $\tilde{\mathbf{v}}_i > \tilde{\mathbf{v}}_j$ if $i > j$) with the corresponding $\tilde{\mathbf{w}}$, then eq. (4.9)

can be applied to define a piecewise linear function ϕ such that statement 1 is satisfied. Furthermore, eq. (4.17) implies that

$$(\tilde{\mathbf{w}}_{k+1} - \tilde{\mathbf{w}}_k)\,(\tilde{\mathbf{w}}_{k+1} - \tilde{\mathbf{w}}_k - \beta\tilde{\mathbf{v}}_k + \beta\tilde{\mathbf{v}}_{k+1}) \leq 0, \quad \forall k = 0, 1, \ldots, N-2. \tag{4.18}$$

Using a similar argument as in the proof of Lemma 4.2.4 in Subsection 4.2.3, eq. (4.18) implies that

$$\phi(\tilde{\mathbf{v}}_{k+1}) \geq \phi(\tilde{\mathbf{v}}_k), \qquad \forall k = 0, 1, \ldots, N-2 \tag{4.19a}$$

$$\phi(\tilde{\mathbf{v}}_{k+1}) - \phi(\tilde{\mathbf{v}}_k) \leq \beta\,(\tilde{\mathbf{v}}_{k+1} - \tilde{\mathbf{v}}_k)\,, \qquad \forall k = 0, 1, \ldots, N-2. \tag{4.19b}$$

That is, ϕ is piecewise monotonic and has piecewise slope upper bound.

Now to prove statement 2, it suffices to prove the case when $b > a$ (the case of $a = b$ is trivially true, and the case of $b < a$ is the same as the case of $b > a$). By Lemma 4.2.4, constraint eq. (4.16) is equivalent to the following two constraints

$$\phi(b) \geq \phi(a), \qquad \forall b > a \tag{4.20a}$$

$$\phi(b) - \phi(a) \leq \beta\,(b - a)\,, \qquad \forall b > a. \tag{4.20b}$$

First consider the case when a and b are in one "piece" of the piecewise linear function ϕ. There are three possibilities: **i)** there is no $k \in \{1, 2, \ldots, N-2\}$ such that $a < \tilde{\mathbf{v}}_k < b$, **ii)** $\tilde{\mathbf{v}}_{N-2} \leq a < b$, or **iii)** $a < b \leq \tilde{\mathbf{v}}_1$. According to eq. (4.9), there exists $i \in \{0, 1, \ldots, N-2\}$ such that

$$\phi(b) = \phi(a) + \frac{\phi(\tilde{\mathbf{v}}_{i+1}) - \phi(\tilde{\mathbf{v}}_i)}{\tilde{\mathbf{v}}_{i+1} - \tilde{\mathbf{v}}_i}\,(b - a)\,. \tag{4.21}$$

Application of eq. (4.19a) and eq. (4.19b) to eq. (4.21) shows eq. (4.20a) and eq. (4.20b), respectively.

Next consider the case when a and b are in different "pieces" of the piecewise linear function ϕ. That is, there exists $k \in (1, 2, \ldots, N-2)$ such that $a < \tilde{\mathbf{v}}_k < b$. According to eq.

(4.9), there exists $i \geq j \in \{0,1,\ldots,N-2\}$ such that

$$\begin{aligned}\phi(b) = \phi(a) &+ \frac{\phi(\tilde{\mathbf{v}}_{i+1}) - \phi(\tilde{\mathbf{v}}_i)}{\tilde{\mathbf{v}}_{i+1} - \tilde{\mathbf{v}}_i}(b - \tilde{\mathbf{v}}_i) + \frac{\phi(\tilde{\mathbf{v}}_{j+1}) - \phi(\tilde{\mathbf{v}}_j)}{\tilde{\mathbf{v}}_{j+1} - \tilde{\mathbf{v}}_j}\left(\tilde{\mathbf{v}}_{j+1} - a\right) \\ &+ \sum_{k=j+1}^{i-1} \frac{\phi(\tilde{\mathbf{v}}_{k+1})\phi(\tilde{\mathbf{v}}_k)}{\tilde{\mathbf{v}}_{k+1} - \tilde{\mathbf{v}}_k}(\tilde{\mathbf{v}}_{k+1} - \tilde{\mathbf{v}}_k)\,.\end{aligned} \tag{4.22}$$

Application of eq. (4.19a) and eq. (4.19b) to eq. (4.22) shows eq. (4.20a) and eq. (4.20b), respectively. ■

In summary, the Wiener-Hammerstein system identification problem in the noiseless case can be defined as

Definition 4.3.2. [Wiener-Hammerstein system identification problem – noiseless case] *Given the input/output measurement* $(\mathbf{u},\mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^N$ *of an unknown Wiener-Hammerstein system and positive integers* N_g *and* N_h*, find decision vectors* $\mathbf{g} \in \mathbb{R}^{N_g}$ *and* $\mathbf{h} \in \mathbb{R}^{N_h}$ *such that there exist signals* $\mathbf{v} \in \mathbb{R}^N$ *and* $\mathbf{w} \in \mathbb{R}^N$ *satisfying eq. (4.13, 4.17).* ■

Remark 4.3.3. It is assumed that $(\mathbf{u},\mathbf{y})$ sufficiently represents the dynamics of the true (but unknown) system. Therefore, a Wiener-Hammerstein model specified by the solution of the problem in Definition 4.3.2 should reasonably describes the dynamics of the system of interest. ■

Remark 4.3.4. The signals $(\mathbf{v},\mathbf{w})$ can be used as the input/output samples of the nonlinearity ϕ in Figure 4-2. ϕ can be defined, for example, using the linear interpolation scheme described in eq. (4.9) in Subsection 4.2.3. ■

Remark 4.3.5. Under the assumption that N_g and N_h are large enough, the impulse responses of the true (but unknown) system $\mathbf{g}^*$ and $\mathbf{h}^*$ constitute a solution to the problem in Definition 4.3.2. Therefore, the problem has at least one solution. The case when N_g and N_h are not large enough can be handled. The discussion will be deferred to Subsection 4.5.2. ■

Remark 4.3.6. Typically there are infinitely many solutions of the problem in Definition 4.3.2, the corresponding normalization issue will be discussed in Subsection 4.3.2. ■

4.3.1 E: Comparison with the model validation techniques

The principles of the identification problem in Definition 4.3.2 and that of the problem of model validation (e.g., [99]) are very similar. Both problems call for a certificate to the satisfiability of the input/output relationships of the blocks in the respective model structures concerned. Definition 4.3.2 seeks a feasibility certificate while model validation seeks an infeasibility certificate. However, there are two major distinctions between the proposed identification setup and the model validation setup. First, for the model validation problem, proving the *existence* of the infeasibility certificate is sufficient. For example, in [99, 104] the question of whether an infeasibility certificate exists is answered by a structured singular value bounding problem. The Wiener-Hammerstein identification problem in Definition 4.3.2, on the other hand, requires the computation of all signals presented in the model. This computation can potentially be expensive. The second distinction of the proposed identification setup from the model validation setup is that the feasibility problem in Definition 4.3.2 will lead to a *non-convex* quadratic program, while most of the previously considered model validation setups lead to the formulation of convex problems. The convexity properties of the optimization problems also lead to a distinction in the solution approaches. The published model validation results are mostly based on rigorous analysis, while the approach adopted in this chapter will be more experimental – some observations will be substantiated by numerical experiments only.

4.3.2 Non-uniqueness of solutions and normalization

The system identification problem in Definition 4.3.2 is feasible with decision vectors $\mathbf{g}^*$ and $\mathbf{h}^*$ (i.e., the impulse responses of the FIR systems in Figure 4-1). However, there are actually infinitely many solutions. Figure 4-4 depicts a way to generate those solutions. The non-uniqueness of solutions requires the normalization of $\mathbf{g}$ and $\mathbf{h}$. However, the normalization issue is not trivial. In fact, uniqueness of solutions cannot be guaranteed in general for the identification problem in this chapter. Two normalization schemes are allowed in Figure 4-4:

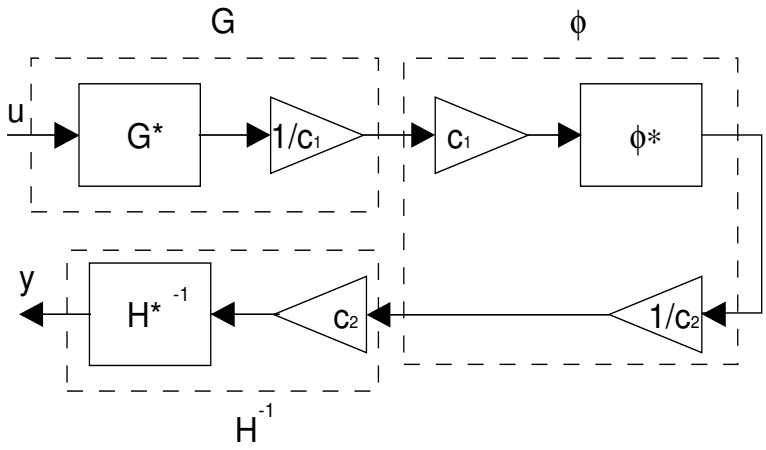


Figure 4-4: Non-uniqueness of the optimal solutions without normalization. Given G^* and H^*, G and H characterize the family of FIR systems with the same input/output relationship. c_1 and c_2 are positive because (G^*, H^*) and (G, H) are assumed/constrained to be positive-real.

- **Partial normalization**: Only one of **g** or **h** is normalized. For example, suppose **h** is normalized, then c_2 is fixed. Then the identification engine can pick c_1 to be any positive number smaller than c_2, so that $\phi(\cdot) = \phi^*(c_1 \cdot)/c_2$ satisfies constraint (4.16). Therefore, there will be an infinite number of solutions.

- **Full normalization**: Both **g** and **h** are normalized. Then the identification engine must fix both c_1 and c_2 to be some function (depending on the type of normalization chosen) of $\mathbf{g}^*$ and $\mathbf{h}^*$, respectively. If β^* is the maximum value of the derivative (where it is defined) of ϕ^*, then $\phi(\cdot) = \phi^*(c_1 \cdot)/c_2$ has maximum derivative $c_1\beta^*/c_2$ (again, where it is defined). It is clear that sector bound condition eq. (4.16) would not allow the identification engine to choose the appropriate ϕ if c_1/c_2 is too large (i.e., when $c_1/c_2 > \beta/\beta^*$). Here the problem is that there is no upper bound of c_1/c_2. For any given normalization scheme, there exist $\mathbf{g}^*$ and $\mathbf{h}^*$ such that $c_1/c_2 > \beta/\beta^*$. Therefore, normalizing both **g** and **h** might be too restrictive in the sense that the identification cannot return any solution when there should be one.

Two conclusions can be made in this subsection:

- Partial normalization should be used because it does not cause any restriction. However, this implies non-uniqueness of the solutions. Therefore, for the rest of the chapter, a particular choice of partial normalization will be assumed:

$$\mathbf{h}_0 \equiv 1. \tag{4.23}$$

While the choice of normalization in eq. (4.23) is somewhat arbitrary, it is not unjustified because $\mathbf{h}_0 = \int_0^{2\pi} \mathrm{Re}\left\{H\left(e^{j\omega}\right)\right\} d\omega > 0$.

- With partial normalization, the constant β in sector bound (4.17) can always be assumed to be one, otherwise it can be absorbed in the part of the decision vector which is not normalized. Therefore, throughout this chapter, all sector bound constraints have their values of β equal to one.

4.3.3 Formulation of the system identification optimization problem

In this subsection the system identification problem defined in Definition 4.3.2 will be simplified and put in a format that would facilitate the study of its solution strategy. Some properties of the optimization problem will also be discussed in Subsection 4.3.4.

Definition 4.3.2 defines a system identification feasibility problem with three constraints given in eq. (4.13) and eq. (4.17). The discussion in Subsection 4.3.2 concludes that a partial normalization of $\mathbf{h}$ (i.e., eq. (4.23)) can be assumed. In addition, with the partial normalization, β in eq. (4.17) can be assumed to be one. Substituting the variables $\mathbf{v}$ and $\mathbf{w}$ using eq. (4.13), the constraint set eq. (4.13) and eq. (4.17) reduces to

$$\left(\Delta\mathbf{Y}_{ij}\mathbf{h}\right)^2 - \left(\Delta\mathbf{Y}_{ij}\mathbf{h}\right)\left(\Delta\mathbf{U}_{ij}\mathbf{g}\right) \leq 0, \quad \forall N-1 \geq i > j \geq 0, \tag{4.24}$$

where

$$\begin{aligned} \Delta\mathbf{U}_{ij} &:= \mathbf{U}_i - \mathbf{U}_j, \\ \Delta\mathbf{Y}_{ij} &:= \mathbf{Y}_i - \mathbf{Y}_j, \end{aligned} \tag{4.25}$$

and

$$\mathbf{U}_i \in \mathbb{R}^{1\times N_g}, \quad \mathbf{U}_i := \begin{bmatrix} \mathbf{U}(i,1) & \mathbf{U}(i,2) & \cdots & \mathbf{U}(i,N_g) \end{bmatrix},$$
$$\mathbf{Y}_i \in \mathbb{R}^{1\times N_h}, \quad \mathbf{Y}_i := \begin{bmatrix} \mathbf{Y}(i,1) & \mathbf{Y}(i,2) & \cdots & \mathbf{Y}(i,N_h) \end{bmatrix},$$

with $\mathbf{U}$ and $\mathbf{Y}$ defined in eq. (4.14) and eq. (4.15), respectively.

Conforming to the standard notation in the field of optimization, define the vector of decision variables $x \in \mathbb{R}^{N_g+N_h}$ as

$$x := \begin{bmatrix} \mathbf{g} \\ \mathbf{h} \end{bmatrix}, \tag{4.26}$$

then corresponding to eq. (4.23), the partial normalization constraint set will be denoted as

$$\mathcal{X} := \left\{ x = \begin{bmatrix} \mathbf{g} \\ \mathbf{h} \end{bmatrix} \in \mathbb{R}^{N_g+N_h} \,\middle|\, \mathbf{h}_0 = 1 \right\}. \tag{4.27}$$

In addition, define the matrices $A_{ij} \in \mathbb{R}^{(N_g+N_h)\times(N_g+N_h)}$ as

$$A_{ij} := \begin{bmatrix} \left(\Delta\mathbf{Y}_{ij}\right)'\left(\Delta\mathbf{Y}_{ij}\right) & -\frac{1}{2}\left(\Delta\mathbf{Y}_{ij}\right)'\left(\Delta\mathbf{U}_{ij}\right) \\ -\frac{1}{2}\left(\Delta\mathbf{U}_{ij}\right)'\left(\Delta\mathbf{Y}_{ij}\right) & 0 \end{bmatrix}. \tag{4.28}$$

Then eq. (4.24) is the same as

$$x' A_{ij} x \leq 0, \quad \forall N-1 \geq i > j \geq 0. \tag{4.29}$$

Using the notation A_{ij} defined in eq. (4.28), the system identification optimization problem can be formulated as follows.

$$\begin{aligned} \underset{x\in\mathcal{X}, r\in\mathbb{R}}{\text{minimize}} \quad & r \\ \text{subject to} \quad & x' A_{ij} x \leq r, \quad \forall i > j \\ & r \geq 0, \end{aligned} \tag{4.30}$$

where $\mathcal{X}$ is defined in eq. (4.27) and A_{ij} are defined in eq. (4.28). Program (4.30) and the feasibility problem in Definition 4.3.2 are equivalent in the following sense: $\hat{x}$ is an optimal of program (4.30) if and only if the corresponding $\hat{\mathbf{g}}$ and $\hat{\mathbf{h}}$ (see eq. (4.26)) is a

feasible solution of the problem in Definition 4.3.2. The equivalence can be explained in the following schematics (with $\hat{x}$ and $\hat{\mathbf{g}}$ and $\hat{\mathbf{h}}$ related by eq. (4.26)).

$$\begin{aligned} & \hat{\mathbf{g}} \text{ and } \hat{\mathbf{h}} \text{ is a solution according to Definition 4.3.2.} \\ \Longleftrightarrow\ & \hat{\mathbf{g}} \text{ and } \hat{\mathbf{h}} \text{ satisfies eq. (4.24).} \\ \Longleftrightarrow\ & \hat{x} \text{ satisfies eq. (4.29)} \\ \Longleftrightarrow\ & \hat{x} \text{ is an optimal solution of program (4.30).} \end{aligned} \tag{4.31}$$

In eq. (4.31) all but the last equivalence have been discussed. The last equivalence is true only in the noiseless identification case – the normalized FIR system coefficients $\mathbf{g}^*$ and $\mathbf{h}^*$ is an optimal solution of program (4.30) with an optimal objective value of zero, hence any optimal solution of program (4.30) satisfies eq. (4.29).

The reason for formulating the system identification problem as an optimization problem in (4.30) will become clear in Section 4.5, in which an optimization problem of the same form will be formulated.

4.3.4 Properties of the system identification optimization problem

The matrices A_{ij} in (4.28) can be written as

$$A_{ij} = p_{ij}\left(p_{ij}\right)' - q_{ij}\left(q_{ij}\right)',$$

where

$$p_{ij} = \begin{bmatrix} \left(\Delta\mathbf{Y}_{ij}\right)' \\ -\frac{1}{2}\left(\Delta\mathbf{U}_{ij}\right)' \end{bmatrix} \quad \text{and} \quad q_{ij} = \begin{bmatrix} 0 \\ -\frac{1}{2}\left(\Delta\mathbf{U}_{ij}\right)' \end{bmatrix} \tag{4.32}$$

From (4.32), it can be seen that A_{ij} are rank two matrices with one positive eigenvalue and one negative eigenvalue. Therefore, program (4.30) is a non-convex QP, which is $\mathcal{NP}$ hard.

On the other hand, it can be seen that the absolute value of the positive eigenvalue is (much) greater than that of the negative eigenvalue. This fact suggests that program (4.30) might be an "easy" $\mathcal{NP}$ hard problem. This hypothesis is indeed justified by the following

numerical experiment. Define a proximity function $R : \mathbb{R}^{N_g+N_h} \mapsto \mathbb{R}_+$ as

$$R(x) := \max_{N-1 \geq i > j \geq 0} \left\{ 0, x' A_{ij} x \right\}. \tag{4.33}$$

Then let $\tilde{d} \in \mathbb{R}^{N_g+N_h}$ be such that $\tilde{d}(i)$ is a zero mean unit variance Gaussian random variable for all i, and let x^* be the vector corresponding to $\mathbf{g}^*$ and $\mathbf{h}^*$. Then normalize $\tilde{d}$ to d such that $x^* + sd \in \mathcal{X}$ for all $s \in \mathbb{R}$ and $\|d\| = 1$. Consider one dimensional function $\tilde{R} : \mathbb{R} \mapsto \mathbb{R}_+$ such that $\tilde{R}(s) := R(x^* + sd)$. Plot this function for a range of s (e.g., $s \in [-0.1, 0.1]$). Repeat the process with another randomly generated d for many times and check the shape of the function $\tilde{R}$ (for different d) around $s = 0$. The outcome of the numerical experiment is shown in Figure 4-5. Figure 4-5 suggests that program (4.30) is

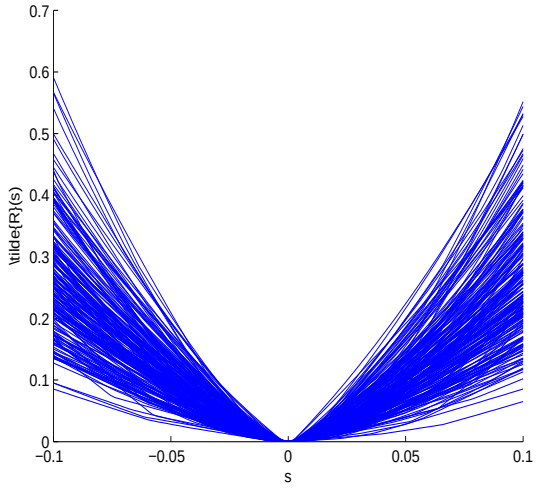


Figure 4-5: Plot of $\tilde{R}(s)$ in 200 (normalized) randomly generated directions. Note that $\tilde{R}(s)$ is not a convex function, but it is almost convex.

almost convex, substantiating the previous notion that program (4.30) should not be a too difficult problem to solve.

Finally, the following properties of the proximity function R defined in eq. (4.33) will be assumed but not formally proved.

$$\exists K \in \mathbb{R}_+ : \forall x \in \mathcal{X},\ \exists \hat{x} \in \operatorname*{argmin}_{\tilde{x} \in \mathcal{X}} R(\tilde{x}) : \|x - \hat{x}\| \leq K R(x), \tag{4.34}$$

and

$$\lim_{\|x\|\to\infty} R(x) = \infty. \tag{4.35}$$

4.4 Solving the Optimization Problem

Subsection 4.3.3 concludes with the formulation of program (4.30), which is repeated here

$$\begin{aligned} \underset{x\in\mathcal{X},r\in\mathbb{R}}{\text{minimize}} \quad & r \\ \text{subject to} \quad & x'A_{ij}x \le r, \quad \forall i > j \\ & r \ge 0, \end{aligned} \tag{4.36}$$

where $\mathcal{X} \subset \mathbb{R}^{N_g+N_h}$ (defined in (4.27)) is the normalization constraint set, and A_{ij} are sign-indefinite matrices defined in (4.28) and (4.32). Optimization problem (4.36) is a non-convex QP, which is $\mathcal{NP}$ hard. The solution to this computation challenge will be the topic for the rest of this section.

The solution procedure for solving optimization problem (4.36) can be divided into three steps, which will be discussed in detail in the following three subsections.

1. A convex semidefinite programming (SDP) relaxation of (4.36) is set up and solved.

2. The optimal solution of the SDP relaxation will be used as an initial guess for a local minimization algorithm, which brings the solution closer to the true optimum.

3. A partial optimization is performed to find the lookup table for the nonlinearity ϕ. Another (easily solvable) convex optimization will be solved to make sure that the FIR systems of the final identified model will be positive real passive.

4.4.1 Semidefinite programming relaxation

SDP relaxation is a standard attempt to solve non-convex QP's (e.g., [113]). To understand the relaxation, it is noted that in optimization problem (4.36) the following is true

$$x'A_{ij}x = \mathrm{Tr}\left(A_{ij}xx'\right) = \mathrm{Tr}\left(A_{ij}X\right), \quad X = X' \ge 0,\ \mathrm{rank}\,(X) = 1. \tag{4.37}$$

A standard procedure to obtain a SDP relaxation is to drop the rank constraint in (4.37), which leads to

$$\begin{aligned} \underset{X\in\mathcal{X}_{\mathbf{s}},r\in\mathbb{R}}{\text{minimize}} \quad & r \\ \text{subject to} \quad & \text{Tr}\left(A_{ij}X\right)\leq r, \quad \forall\, i>j \\ & r\geq 0 \\ & X=X'\geq 0, \end{aligned} \tag{4.38}$$

where $\mathcal{X}_{\mathbf{s}}$ is the normalization constraint set for X corresponding to $\mathcal{X}$ for x. For example, if it is a constraint in (4.36) that $x(i)=1$ for some $i\in\mathbb{N}$, then the corresponding constraint for X in (4.38) is $X(1,i)=X(i,1)=X(i,i)=1$. Once the relaxation (4.38) is solved, the singular vector corresponding to the largest singular value of the matrix solution is returned as the best suboptimal solution to (4.36). It is obvious that the lower the rank of X is, the better the quality of the suboptimal solution will be.

For the noiseless setup in this section, the minimum value of r is actually zero, attainable by, for example, $x^*:=\left[(\mathbf{g}^*)' \quad (\mathbf{h}^*)'\right]'$. Hence, the matrix solution $X^*\equiv x^*x^{*\prime}$ is an optimal solution to relaxation (4.38). This in turn allows (4.38) to be formulated as a minimization problem with an objective function. The choice of a zero objective function leads back to program (4.36), but a more reasonable choice is the trace of the matrix because it has been shown that minimizing this objective function leads to low rank matrix solutions (e.g., [114]). Consequently, the relaxation of (4.38) is reformulated as

$$\begin{aligned} \underset{X\in\mathcal{X}_{\mathbf{s}}}{\text{minimize}} \quad & \text{Tr}\left(X\right) \\ \text{Subject to} \quad & \text{Tr}\left(A_{ij}X\right)\leq 0 \\ & X=X'\geq 0 \end{aligned} \tag{4.39}$$

The tightness of the relaxation depends upon the nonlinearity in Figure 4-2, but not too much on the FIR systems G and H. The above observation is made through the following numerical experiment: 300 instances of program (4.39) were solved. The input/output data were generated by driving 300 randomly generated Wiener-Hammerstein systems with the block diagram in Figure 4-2. G and H were randomly generated, but the nonlinearity ϕ were fixed. For the first one hundred cases, ϕ was a hyperbolic tangent (i.e., $\phi(v)=\tanh(v)$). For

the next one hundred cases, ϕ was a saturated linearity (i.e., $\phi(v) = \mathrm{sgn}(v)\max\{|v|, 1\}$). For the last one hundred cases, ϕ was a cubic nonlinearity (i.e., $\phi(v) = v^3$). It is clear that the cubic nonlinearity does not have a derivative bound, whereas the former two nonlinearities have. After solving the 300 instances of program (4.39), the histograms of the percentage ratios of the second largest and the largest singular values of the symmetric solution matrix X are plotted in Figure 4-6, Figure 4-7 and Figure 4-8, respectively.

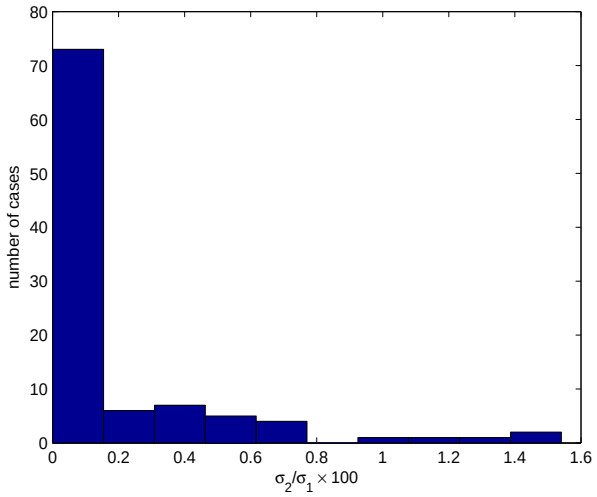


Figure 4-6: **Hyperbolic tangent test case**. Histogram of the percentage of the second largest singular value to the maximum singular value of the optimal SDP relaxation solution matrix X. The second largest singular values never exceed 1.6% of the maximum singular values in the experiment. Data was collected from 100 randomly generated test cases. $N_h = N_g = 4$.

While the relaxation (4.39) provides a reasonably good approximation to the true optimal solution of the original non-convex problem (4.36), the approximation should always be refined by some inexpensive procedure such as a linearized local search described in the next subsection.

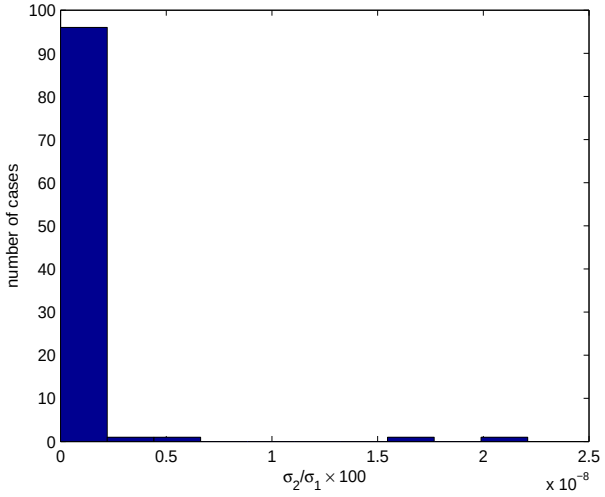


Figure 4-7: **Saturated linearity test case**. Histogram of the percentage of the second largest singular value to the maximum singular value of the optimal SDP relaxation solution matrix X. X is practically a rank one matrix. Data was collected from 100 randomly generated test cases. $N_h = N_g = 4$.

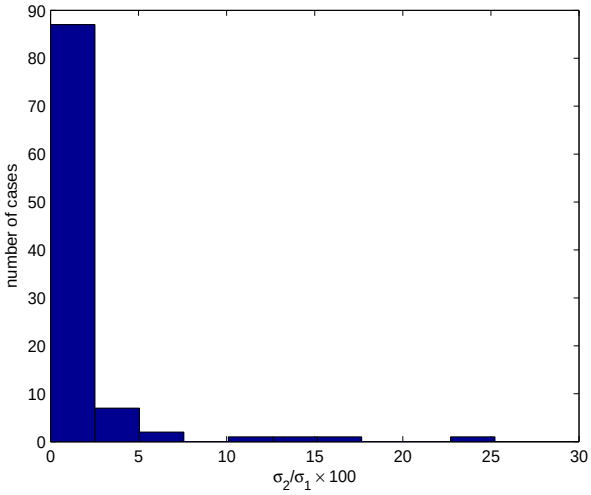


Figure 4-8: **Cubic nonlinearity test case**. Histogram of the percentage of the second largest singular value to the maximum singular value of the optimal SDP relaxation solution matrix X. For a lot of cases, the second largest singular values never exceed 5% of the maximum singular values in the experiment, but there are some cases when the SDP relaxation performs poorly. Data was collected from 100 randomly generated test cases. $N_h = N_g = 4$.

4.4.2 Local search

A local search is the following optimization procedure:

Definition 4.4.1. *Given an initial guess $x_0 \in \mathbb{R}^{N_g+N_h}$, generate a sequence $\{x_1, x_2, \ldots, x_m\}$ using the formula*

$$x_{k+1} = x_k + s_k \Delta x_k, \quad k = 0, 1, \ldots, m-1$$

where $\Delta x_k \in \mathbb{R}^{N_g+N_h}$ is the search direction and $s_k \in \mathbb{R}$ is the step length defined to minimize some objective function. The sequence $\{x_k\}$ terminates when certain criterion is met (e.g., $\|\Delta x_k\| < \varepsilon$ for some pre-specified small number $\varepsilon > 0$). ■

In this thesis, the search direction is chosen such that the *linearized* (at the current iterate x_k) proximity function defined in eq. (4.33) in Subsection 4.3.4 is minimized. Given the current iterate x_k, a search direction Δx_k should also be admissible. That is,

$$\Delta x_k \in \mathcal{X}_\Delta(x_k) := \left\{ y \in \mathbb{R}^{N_g+N_h} \middle| x_k + sy \in \mathcal{X},\ \forall s \in \mathbb{R} \right\}$$

Then it is natural to seek $\Delta x_k \in \mathcal{X}_\Delta(x_k)$ such that

$$\max_{i>j} \left\{ 0, (x_k + \Delta x_k)' A_{ij} (x_k + \Delta x_k) \right\} \to \min. \tag{4.40}$$

Problem (4.40), however, is as difficult as (4.36). Nevertheless, if the term $(\Delta x_k)' A_{ij} \Delta x_k$ is ignored, then it leads to

$$\begin{array}{ll} \underset{\Delta x_k, r \in \mathbb{R}}{\text{minimize}} & r \\ \text{subject to} & x_k' A_{ij} x_k + 2 x_k' A_{ij} \Delta x_k \le r, \quad \forall i > j \\ & r \ge 0 \\ & \Delta x_k \in \mathcal{X}_\Delta(x_k). \end{array} \tag{4.41}$$

Optimization problem (4.41) is a linear program (LP) with respect to decision variables r and Δx_k. It can be solved relatively cheaply [65].

Once the search direction Δx_k has been found by solving program (4.41), the line search

procedure can be applied to solve the *nonlinear* problem for the optimal step length.

$$s_k := \underset{s}{\operatorname{argmin}} \max_{i>j} \left\{0, (x_k + s\Delta x_k)' A_{ij} (x_k + s\Delta x_k)\right\}. \tag{4.42}$$

Note that program (4.42) is typically a non-convex problem, and therefore it is not supposed to be solved to optimality. Nevertheless, program (4.42) is a one-dimensional optimization problem and good algorithms exist to approximately solve it. For example, the algorithm implemented in this thesis work was based on a quadratic function approximation scheme described in [115].

4.4.3 Final optimizations

There are two reasons for performing some optimizations after the SDP relaxation (Subsection 4.4.1) and the local search (Subsection 4.4.2). The two reasons will lead to two optimization tasks: **partial optimization** and **passivity enforcement**.

The first reason is to solve some relatively inexpensive problems to further improve the quality of the identification. Note that the constraint eq. (4.24) is convex with respect to $\mathbf{g}$ and $\mathbf{h}$ *individually* – eq. (4.24) is a linear constraint with respect to $\mathbf{g}$, and a convex quadratic constraint with respect to $\mathbf{h}$. Suppose $\hat{\mathbf{g}}$ and $\hat{\mathbf{h}}$ are the solutions of the local search. Then the following optimization problem can be solved to improve the quality of $\hat{\mathbf{g}}$.

$$\begin{aligned} \underset{r,\,\mathbf{g}:(\mathbf{g},\hat{\mathbf{h}})\in\mathcal{X}}{\text{minimize}} \quad & r \\ \text{subject to} \quad & \left(\Delta\mathbf{Y}_{ij}\hat{\mathbf{h}}\right)^2 - \left(\Delta\mathbf{Y}_{ij}\hat{\mathbf{h}}\right)\left(\Delta\mathbf{U}_{ij}\mathbf{g}\right) \le r, \quad \forall i > j, \\ & r \ge 0. \end{aligned} \tag{4.43}$$

Program (4.43) is a LP with decision variables r and $\mathbf{g}$. It can be solved efficiently [65]. Conversely, the following optimization problem can be solved to improve the quality of $\hat{\mathbf{h}}$.

$$\begin{aligned} \underset{r,\,\mathbf{h}:(\hat{\mathbf{g}},\mathbf{h})\in\mathcal{X}}{\text{minimize}} \quad & r \\ \text{subject to} \quad & \left(\Delta\mathbf{Y}_{ij}\hat{\mathbf{g}}\right)^2 - \left(\Delta\mathbf{Y}_{ij}\hat{\mathbf{g}}\right)\left(\Delta\mathbf{U}_{ij}\mathbf{h}\right) \le r, \quad \forall i > j, \\ & r \ge 0. \end{aligned} \tag{4.44}$$

Program (4.44) is a convex QP with decision variables r and $\mathbf{h}$. It can also be solved efficiently [116]. Other partial refinements in the spirit of programs (4.43) and (4.44) are also possible. See [108] for an example.

The second reason for the final optimization is the positive real passivity enforcement of the final model of $\mathbf{h}$. Recall the definition of positive real passivity

$$\mathrm{Re}\left\{H\left(e^{j\omega}\right)\right\} = h_0 + h_1\cos(\omega) + \ldots + h_{N_h-1}\cos\left(\left(N_h-1\right)\omega\right) > 0. \tag{4.45}$$

It can be verified (see [117], for example) that eq. (4.45) is true if and only if there exists $Q = Q' \in \mathbb{R}^{(N_h-1)\times(N_h-1)}$ such that

$$\begin{bmatrix} Q & \frac{1}{2}\check{\mathbf{h}} \\ \frac{1}{2}\check{\mathbf{h}}' & h_0 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & Q \end{bmatrix} > 0, \tag{4.46}$$

where

$$\check{\mathbf{h}} := \begin{bmatrix} h_{N_h-1} & h_{N_h-2} & \cdots & h_1 \end{bmatrix}' \in \mathbb{R}^{N_h-1},$$

and inequality (4.46) means that the left side is a positive definite matrix. Note that (4.46) is a linear matrix inequality with variables Q, h_0 and $\check{\mathbf{h}}$ (a truncated reversed version of $\mathbf{h}$).

Now suppose $\hat{\mathbf{h}}$ is the identified FIR system impulse response coefficients by the relaxation/local search procedure. Then the passive refinement of $\hat{\mathbf{h}}$ can be found by solving

$$\begin{array}{ll} \underset{\mathbf{h}}{\text{minimize}} & \left\|\mathbf{h}-\hat{\mathbf{h}}\right\|_2 \\ \text{subject to} & (4.46). \end{array} \tag{4.47}$$

Optimization problem (4.47) is a SDP with very few decision variables and constraints. It can be solved efficiently [67]. In addition, it is noted that while program (4.47) is given with $\mathbf{h}$ being the decision variables, exactly the same procedure can be applied to enforce the passivity of $\mathbf{g}$ as well.

Finally, note that while the tasks of partial optimization and passivity enforcement are described separately, they can be combined to formulate a single optimization problem. For example, constraint (4.46) can be incorporated into program (4.44) to form a convex SDP.

Similarly, an analogous version of (4.46) can also be incorporated into program (4.43).

4.4.4 System identification algorithm summary

The solution procedure to solve the Wiener-Hammerstein system identification problem according to Definition 4.3.2 can be summarized into the following steps.

Algorithm: W-H (noiseless)
Input: Input/output measurement $(\mathbf{u}, \mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^N$, lengths of FIR systems N_g and N_h
Output: FIR system coefficients $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}\right) \in \mathbb{R}^{N_g} \times \mathbb{R}^{N_h}$, piecewise linear nonlinearity $\hat{\phi}$

1. Given $(\mathbf{u}, \mathbf{y})$, use eq. (4.14) and eq. (4.15) to define Toeplitz matrices $\mathbf{U}$ and $\mathbf{Y}$.
2. Use eq. (4.28) and eq. (4.25) to define sign indefinite matrices A_{ij} for all time indices $N-1 \geq i > j \geq 0$.
3. Set up and solve SDP (4.39) to obtain the solution matrix X. Denote x_0 as the dominant singular vector of X.
4. With x_0 being the initial guess, solve the local search problem in Definition 4.4.1.
5. Refine the optimal solution of the local search by apply the positive real passivity enforcement program (4.47) and/or the partial optimization of program (4.44), (4.43). Denote $\hat{x}$ as the optimal solution after all the final optimizations.
6. Define $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}\right) := \hat{x}$, and $\hat{\mathbf{v}} := \mathbf{U}\hat{\mathbf{g}}$, $\hat{\mathbf{w}} := \mathbf{Y}\hat{\mathbf{h}}$. Define the output nonlinearity $\hat{\phi}$ specified by $(\hat{\mathbf{v}}, \hat{\mathbf{w}})$ (sorting and extracting unique $\hat{\mathbf{v}}$ entries if necessary) using eq. (4.9). Return the outputs $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}, \hat{\phi}\right)$.

4.5 Identification of Wiener-Hammerstein System – with Measurement Noise

The development of this section will be parallel to the combination of Section 4.3 and Section 4.4. Differences between the noiseless and the noisy cases will be highlighted.

4.5.1 System identification problem formulation

The model to be identified is still of the Wiener-Hammerstein structure in Figure 4-2 with decision variables **g** and **h** and ϕ being specified by a lookup table. Because of the output measurement noise, however, the system identification feasibility problem will be different and it is shown in Figure 4-9.

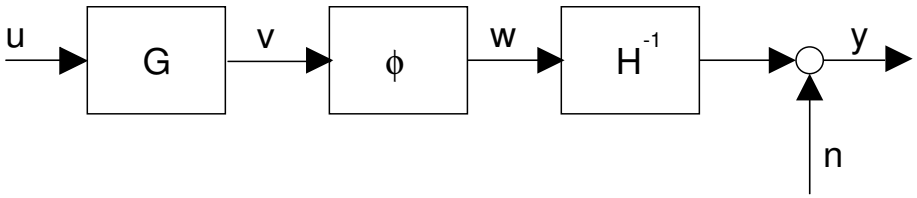


Figure 4-9: A feasibility problem to determine the impulse responses of the FIR systems G and H. Here **u** and **y** are the given input and output measurement generated by the true (but unknown) system. The signals **v** and **w** are the outputs of G and H, respectively. The signal **n** is the noise corrupting the output measurement. In the feasibility problem, **v**, **w** and **n** are extra variables chosen so that, together with **g** and **h**, they define a function ϕ satisfying sector bound constraint eq. (4.16).

There is an extra signal $\mathbf{n} \in \mathbb{R}^N$ to be determined in the feasibility problem in Figure 4-9. Define the Toeplitz matrix $\mathbf{N} \in \mathbb{R}^{N \times N_h}$:

$$\mathbf{N} := \begin{bmatrix} n[0] & 0 & \dots & 0 \\ n[1] & n[0] & \ddots & \vdots \\ & & \ddots & 0 \\ \vdots & \vdots & & n[0] \\ & & & \vdots \\ n[N-1] & n[N-2] & \dots & n[N-N_h] \end{bmatrix}_{N \times N_h} . \tag{4.48}$$

The constraint set defined in Figure 4-9 can be given as follows.

$$\mathbf{v} = \mathbf{U}\mathbf{g}, \tag{4.49a}$$

$$\mathbf{w} = (\mathbf{Y} - \mathbf{N})\,\mathbf{h}, \tag{4.49b}$$

$$\left(\mathbf{w}_i - \mathbf{w}_j\right)\left(\mathbf{w}_i - \mathbf{w}_j - \mathbf{v}_i + \mathbf{v}_j\right) \le 0, \quad \forall N-1 \ge i > j \ge 0. \tag{4.49c}$$

Then the Wiener-Hammerstein system identification problem with output measurement noise can be defined as

Definition 4.5.1. [Wiener-Hammerstein system identification problem – noisy case] *Given the input/output measurement* $(\mathbf{u},\mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^N$ *of an unknown Wiener-Hammerstein system and positive integers* N_g *and* N_h*, find decision vectors* $\mathbf{g} \in \mathbb{R}^{N_g}$ *and* $\mathbf{h} \in \mathbb{R}^{N_h}$ *such that there exist signals* $\mathbf{v} \in \mathbb{R}^N$, $\mathbf{w} \in \mathbb{R}^N$ *and* $\mathbf{n} \in \mathbb{R}^N$ *satisfying eq. (4.49a, 4.49b, 4.49c).* ■

4.5.2 Formulation of the system identification optimization problem

Parallel to the development in Subsection 4.3.3, the feasibility problem in Definition 4.5.1 will be simplified. However, instead of formulating and solving an equivalent optimization problem as it was in Subsection 4.3.3, a *relaxation* will be formulated due to computation considerations.

Substituting eq. (4.49a) and eq. (4.49b) into eq. (4.49c) yields

$$\left(\Delta\mathbf{Y}_{ij}\mathbf{h}\right)^2 - \left(\Delta\mathbf{Y}_{ij}\mathbf{h}\right)\left(\Delta\mathbf{U}_{ij}\mathbf{g}\right) \leq \left(\Delta\mathbf{N}_{ij}\mathbf{h}\right)\left(2\Delta\mathbf{Y}_{ij}\mathbf{h} - \Delta\mathbf{U}_{ij}\mathbf{g}\right) - \left(\Delta\mathbf{N}_{ij}\mathbf{h}\right)^2, \quad \forall i > j, \tag{4.50}$$

where

$$\Delta\mathbf{N}_{ij} := \mathbf{N}_i - \mathbf{N}_j \tag{4.51}$$

and

$$\mathbf{N}_i \in \mathbb{R}^{1\times N_h}, \quad \mathbf{N}_i := \begin{bmatrix} \mathbf{N}(i,1) & \mathbf{N}(i,2) & \cdots & \mathbf{N}(i,N_h) \end{bmatrix},$$

with $\mathbf{N}$ defined in eq. (4.48). Constraint (4.50) is difficult to handle because of the terms in the right-hand side with the extra variables of $\mathbf{n}$. Therefore, it is proposed in this thesis that the following *relaxed* constraint should be imposed instead. That is,

$$\left(\Delta\mathbf{Y}_{ij}\mathbf{h}\right)^2 - \left(\Delta\mathbf{Y}_{ij}\mathbf{h}\right)\left(\Delta\mathbf{U}_{ij}\mathbf{g}\right) \leq \mathbf{r}_{ij}, \quad \forall N-1 \geq i > j \geq 0, \tag{4.52}$$

with variables $\mathbf{g}$, $\mathbf{h}$ and $\mathbf{r} \in \mathbb{R}_+^{N(N-1)/2}$. Constraint eq. (4.52) is linear with respect to $\mathbf{r}$, and therefore it is no more difficult to handle than eq. (4.24) in Subsection 4.3.3. Based on the "robustness principle" that eq. (4.50) should be satisfied by a noise vector $\mathbf{n}$ with the min-

imum norm, the relaxed system identification optimization problem should be formulated to minimize some kind of norm of $\mathbf{r}$ as well. In this thesis, the norm is chosen to be the infinity norm. Then, using the notations x defined in eq. (4.26), $\mathcal{X}$ defined in eq. (4.27) and A_{ij} in eq. (4.28) in Subsection 4.3.3. The relaxed system identification optimization problems can be given as

$$\begin{array}{ll} \underset{x\in\mathcal{X},r\in\mathbb{R}}{\text{minimize}} & r \\ \text{subject to} & x'A_{ij}x \leq r, \quad \forall\, i > j \\ & r \geq 0. \end{array} \tag{4.53}$$

Note that program (4.53) has exactly the same form as program (4.36), the noiseless case in Subsection 4.3.3. However, in general, the minimum objective value of program (4.53) will not be zero. Accordingly, the solution procedure described in Section 4.4 should be modified. This will be explained in Subsection 4.5.3.

A question of great concern is how good the relaxed optimization problem (4.53) is. This can be answered by a characterization of the distance between the optimal solutions to program (4.53) with or without output measurement noise. The following statement gives a theoretical guideline.

Lemma 4.5.2. *Denote* $\mathbf{n}^*$ *as the vector of output measurement noise. Let* $\hat{\mathbf{g}}$ *and* $\hat{\mathbf{h}}$ *be a solution of program (4.53) when the matrices* A_{ij} *are defined with input/output measurement* $(\mathbf{u},\mathbf{y})$ **with** *noise* $\mathbf{n}^*$. *Let* $\mathbf{g}^*$ *and* $\mathbf{h}^*$ *be a solution of program (4.36) when the matrices* A_{ij} *are defined with input/output measurement* $(\mathbf{u},\mathbf{y})$ **without** *noise* $\mathbf{n}^*$. *Then if the proximity function property in eq. (4.34) (when* A_{ij} *are defined with noise) is satisfied, then*

$$\left\| \left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right) - \left(\mathbf{g}^*,\mathbf{h}^*\right) \right\|_2 = O\left(\|\mathbf{n}^*\|_2\right), \quad \textit{when } \|\,\mathbf{n}^*\|_2 \textit{ is small enough.} \tag{4.54}$$

■

Proof of Lemma 4.5.2. First, note that $\mathbf{g}^*$ and $\mathbf{h}^*$ satisfies the sector bound (with system

input $\mathbf{u}$ and output $\mathbf{y}-\mathbf{n}^*$), which simplifies to

$$\left(\Delta\mathbf{Y}_{ij}\mathbf{h}^*\right)^2-\left(\Delta\mathbf{Y}_{ij}\mathbf{h}^*\right)\left(\Delta\mathbf{U}_{ij}\mathbf{g}^*\right)\leq\left(\Delta\mathbf{N}_{ij}^*\mathbf{h}^*\right)\left(2\Delta\mathbf{Y}_{ij}\mathbf{h}^*-\Delta\mathbf{U}_{ij}\mathbf{g}^*\right)-\left(\Delta\mathbf{N}_{ij}^*\mathbf{h}^*\right)^2,\quad\forall i>j, \tag{4.55}$$

where

$$\Delta\mathbf{N}_{ij}^*:=\mathbf{N}_i^*-\mathbf{N}_j^*$$

and

$$\mathbf{N}_i^*\in\mathbb{R}^{1\times N_h},\quad \mathbf{N}_i^*:=\begin{bmatrix}\mathbf{N}^*(i,1) & \mathbf{N}^*(i,2) & \cdots & \mathbf{N}^*(i,N_h)\end{bmatrix},$$

with

$$\mathbf{N}^*:=\begin{bmatrix} n^*[0] & 0 & \dots & 0\\ n^*[1] & n^*[0] & \ddots & \vdots\\ & & \ddots & 0\\ \vdots & \vdots & & n^*[0]\\ & & & \vdots\\ n^*[N-1] & n^*[N-2] & \dots & n^*[N-N_h]\end{bmatrix}_{N\times N_h}.$$

Then, comparing the definition of R in eq. (4.33), the relation in eq. (4.55) suggests that

$$\begin{aligned} R((\mathbf{g}^*,\mathbf{h}^*)) &= \max_{i>j}\left\{0,\left(\Delta\mathbf{Y}_{ij}\mathbf{h}^*\right)^2-\left(\Delta\mathbf{Y}_{ij}\mathbf{h}^*\right)\left(\Delta\mathbf{U}_{ij}\mathbf{g}^*\right)\right\}\\ &= O\left(\|\mathbf{n}^*\|_2\|\mathbf{h}^*\|_2\right),\quad \text{when } \|\mathbf{n}^*\|_2 \text{ is small,}\end{aligned} \tag{4.56}$$

where the fact that $\Delta\mathbf{N}_{ij}^*\mathbf{h}^*=O\left(\|\mathbf{n}^*\|_2\|\mathbf{h}^*\|_2\right)$ has been used because $\Delta\mathbf{N}_{ij}^*$ is a linear function of $\mathbf{n}^*$.

On the other hand, by the statement $\left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right)$ is a minimizer of R. Therefore,

$$R\left(\left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right)\right)\leq R\left((\mathbf{g}^*,\mathbf{h}^*)\right),$$

and hence

$$R\left(\left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right)\right)=O\left(\|\mathbf{n}^*\|_2\|\mathbf{h}^*\|_2\right). \tag{4.57}$$

Application of the triangular inequality to eq. (4.56) and eq. (4.57) yields

$$\left\| R\left(\left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right)\right) - R\left(\left(\mathbf{g}^*,\mathbf{h}^*\right)\right)\right\|_2 = O\left(\|\mathbf{n}^*\|_2 \|\mathbf{h}^*\|_2\right). \tag{4.58}$$

Finally, applying proximity function property in eq. (4.34) to eq. (4.58) implies the existence of a constant K such that

$$\begin{aligned} \| \left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right) - \left(\mathbf{g}^*,\mathbf{h}^*\right) \|_2 &= O\left(K\|\mathbf{n}^*\|_2 \|\mathbf{h}^*\|_2\right) \\ &= O\left(\|\mathbf{n}^*\|_2\right), \end{aligned}$$

thus concluding the proof. ■

Remark 4.5.3. Eq. (4.54) in Lemma 4.5.2 states that the difference of the solutions in the noisy and noiseless setups are linearly upper bounded by the norm of the noise vector $\mathbf{n}$. This justifies the use of the relaxed system identification optimization problem (4.53). ■

Remark 4.5.4. The proximity function property defined in eq. (4.34) is central to the proof of Lemma 4.5.2 – it relates the proximity in terms of objective function value to the proximity in terms of the decision vector itself. Although a formal proof is not available at this stage, this conjecture is supported by numerical evidence shown in Figure 4-5. ■

Finally, it is noted that the minimization of the norm of $\mathbf{r}$ in program (4.53) has additionally the following implication: suppose the lengths N_g or N_h is not large enough to sufficiently represent the impulse response of the corresponding FIR systems in the true (but unknown) system, then the minimization of $\mathbf{r}$ seeks to minimize the violation of feasibility of the left-hand side of eq. (4.52).

4.5.3 Reformulation of SDP relaxation

The relaxation of the feasibility problem in Definition 4.5.1 leads to the optimization problem (4.53), which has exactly the same form as program (4.36) with only one exception – the minimum of program (4.53) is not necessarily zero in the presence of output measurement noise. Therefore, all of the solution steps described in Section 4.4 apply to the noisy problem (4.53) with the exception that the feasibility problem (4.39) is infeasible,

and hence it cannot be part of the solution procedure. The following SDP will be solved in place of program (4.39).

$$\begin{aligned} \underset{X\in\mathcal{X}_{\mathbf{s}},r\in\mathbb{R}}{\text{minimize}} \quad & \mathrm{Tr}\,(X)+\lambda r \\ \text{Subject to} \quad & \mathrm{Tr}\left(A_{ij}X\right)\leq r \\ & X=X'\geq 0 \\ & r\geq 0 \end{aligned} \tag{4.59}$$

In program (4.59) the constraint set $\mathcal{X}_{\mathbf{s}}$ is defined in (4.39), and the matrices A_{ij} are defined in eq. (4.28). $\lambda > 0$ is a tuning parameter. It turns out that $\lambda = 100$ works pretty well in general. Note that the objective function in program (4.59) represents a tradeoff between the desire to obtain a low-rank solution and the minimization of the norm of the noise.

4.5.4 Section summary

A feasibility problem is given in Definition 4.5.1 to characterize the solution of the Wiener-Hammerstein system identification problem with output measurement noise. The feasibility problem turns out to be difficult to solve and therefore it is further relaxed to form an optimization problem in (4.53). The quality of the relaxation is characterized by Lemma 4.5.2. The relaxation has the same form as program (4.36) in the noiseless case with only one exception – the minimum objective value of the relaxation is above zero. Accordingly, the algorithm for solving the relaxation is the same as that for the noiseless setup except for step 3 below.

Algorithm: W-H (noisy)
Input: Input/output measurement $(\mathbf{u},\mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^N$, lengths of FIR systems N_g and N_h
Output: FIR system coefficients $\left(\hat{\mathbf{g}},\hat{\mathbf{h}}\right) \in \mathbb{R}^{N_g} \times \mathbb{R}^{N_h}$, piecewise linear nonlinearity $\hat{\phi}$

1. Given $(\mathbf{u},\mathbf{y})$, use eq. (4.14) and eq. (4.15) to define Toeplitz matrices $\mathbf{U}$ and $\mathbf{Y}$.
2. Use eq. (4.28) and eq. (4.25) to define sign indefinite matrices A_{ij} for all time indices $N-1 \geq i > j \geq 0$.
3. Set up and solve SDP (4.59) to obtain the solution matrix X. Denote x_0 as the dominant singular vector of X.

4. With x_0 being the initial guess, solve the local search problem in Definition 4.4.1.

5. Refine the optimal solution of the local search by apply the positive real passivity enforcement program (4.47) and/or the partial optimization of program (4.44), (4.43). Denote $\hat{x}$ as the optimal solution after all the final optimizations.

6. Define $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}\right) := \hat{x}$, and $\hat{\mathbf{v}} := \mathbf{U}\hat{\mathbf{g}}$, $\hat{\mathbf{w}} := \mathbf{Y}\hat{\mathbf{h}}$. Define the output nonlinearity $\hat{\phi}$ specified by $(\hat{\mathbf{v}}, \hat{\mathbf{w}})$ (sorting and extracting unique $\hat{\mathbf{v}}$ entries if necessary) using eq. (4.9). Return the outputs $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}, \hat{\phi}\right)$.

4.6 Identification of Wiener-Hammerstein System – with Feedback and Noise

Figure 4-10 shows the feedback Wiener-Hammerstein model which is specified by the FIR

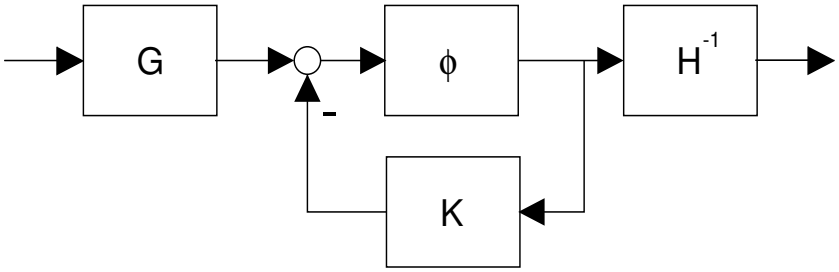


Figure 4-10: The Wiener-Hammerstein model with feedback.

systems G, H, K and the nonlinearity ϕ, which will again be identified in a non-parametric fashion. The setup of the identification feasibility problem, given in Figure 4-11, is slightly different from the model in Figure 4-10. In addition to the decision variables $\mathbf{g} \in \mathbb{R}^{N_g}$ and $\mathbf{h} \in \mathbb{R}^{N_h}$ seen in the previous sections, there are decision variables associated with the FIR system K, which is implicitly characterized by the impulse response of the product of K and H denoted as $\mathbf{k} * \mathbf{h} \in \mathbb{R}^{N_k+N_h-1}$ and the impulse response of H denoted as $\mathbf{h} \in \mathbb{R}^{N_h}$. Once the vectors $\mathbf{k} * \mathbf{h}$ and $\mathbf{h}$ have been determined, a deconvolution can be applied to retrieve the impulse response of K.

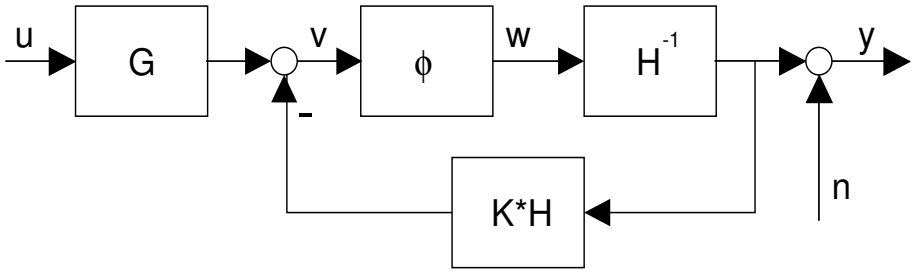


Figure 4-11: A feasibility problem to determine the impulse responses of G, H and $K*H$. Here **u** and **y** are the given input and output measurement generated by the true (but unknown) system. The signals **v** and **w** are the input and output of the nonlinearity ϕ. The signal **n** is the noise corrupting the output measurement. In the feasibility problem, **v**, **w** and **n** are extra variables chosen so that, together with **g**, **h** and $\mathbf{k}*\mathbf{h}$, they define a function ϕ satisfying sector bound constraint eq. (4.16).

The feasibility problem setup in Figure 4-11 leads to the following set of constraints.

$$\mathbf{v} = \mathbf{U}\mathbf{g} - \mathbf{Y}(\mathbf{k}*\mathbf{h}), \tag{4.60a}$$

$$\mathbf{w} = (\mathbf{Y} - \mathbf{N})\mathbf{h}, \tag{4.60b}$$

$$(\mathbf{w}_i - \mathbf{w}_j)(\mathbf{w}_i - \mathbf{w}_j - \mathbf{v}_i + \mathbf{v}_j) \leq 0, \quad \forall N-1 \geq i > j \geq 0, \tag{4.60c}$$

with **U**, **Y** and **N** defined in eq. (4.14), eq. (4.15) and eq. (4.48), respectively. Note that if the following notations are defined

$$\tilde{\mathbf{U}} := \begin{bmatrix} \mathbf{U} & -\mathbf{Y} \end{bmatrix} \quad \text{and} \quad \tilde{\mathbf{g}} := \begin{bmatrix} \mathbf{g} \\ \mathbf{k}*\mathbf{h} \end{bmatrix}, \tag{4.61}$$

then the constraint set eq. (4.60a,4.60b,4.60c) can be written as

$$\mathbf{v} = \tilde{\mathbf{U}}\tilde{\mathbf{g}}, \tag{4.62a}$$

$$\mathbf{w} = (\mathbf{Y} - \mathbf{N})\mathbf{h}, \tag{4.62b}$$

$$(\mathbf{w}_i - \mathbf{w}_j)(\mathbf{w}_i - \mathbf{w}_j - \mathbf{v}_i + \mathbf{v}_j) \leq 0, \quad \forall N-1 \geq i > j \geq 0. \tag{4.62c}$$

As far as the proposed system identification algorithm is concerned, constraint set eq. (4.62a,4.62b,4.62c) has the same form and properties as eq. (4.49a,4.49b,4.49c) in the no

feedback case. Therefore, the analysis and algorithm in Section 4.5 can be applied to the feedback Wiener-Hammerstein system identification simply by replacing constraint set eq. (4.49a,4.49b,4.49c) with eq. (4.62a,4.62b,4.62c). Once the optimal values of the decision vectors **g**, **h** and $\mathbf{k} * \mathbf{h}$ have been found, a deconvolution can be applied to obtain the value of **k** (corresponding to the impulse response of K in Figure 4-10). To summarize, the algorithm for the feedback Wiener-Hammerstein identification case is as follows.

Algorithm: W-H feedback (noisy)
Input: Input/output measurement $(\mathbf{u}, \mathbf{y}) \in \mathbb{R}^N \times \mathbb{R}^N$, lengths of FIR systems N_g, N_h and N_k.
Output: FIR system coefficients $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}, \hat{\mathbf{k}}\right) \in \mathbb{R}^{N_g} \times \mathbb{R}^{N_h} \times \mathbb{R}^{N_k}$, piecewise linear nonlinearity $\hat{\phi}$

1. Given $(\mathbf{u}, \mathbf{y})$, use eq. (4.14) and eq. (4.15) to define Toeplitz matrices **U** and **Y**. Then define $\tilde{\mathbf{U}}$ according to eq. (4.61).

2. With $\tilde{\mathbf{U}}$ in place of **U**, use eq. (4.28) and eq. (4.25) to define sign indefinite matrices A_{ij} for all time indices $N-1 \geq i > j \geq 0$.

3. Set up and solve SDP (4.59) to obtain the solution matrix X. Denote x_0 as the dominant singular vector of X.

4. With x_0 being the initial guess, solve the local search problem in Definition 4.4.1.

5. Refine the optimal solution of the local search by apply the positive real passivity enforcement program (4.47) and/or the partial optimization of program (4.44), (4.43). Denote $\hat{x}$ as the optimal solution after all the final optimizations.

6. Define $\left(\hat{\mathbf{g}}, (\mathbf{k} \hat{*} \mathbf{h}), \hat{\mathbf{h}}\right) := \hat{x}$, and $\hat{\mathbf{v}} := \mathbf{U}\hat{\mathbf{g}} - \mathbf{Y}(\mathbf{k} \hat{*} \mathbf{h})$, $\hat{\mathbf{w}} := \mathbf{Y}\hat{\mathbf{h}}$. Define the output nonlinearity $\hat{\phi}$ specified by $(\hat{\mathbf{v}}, \hat{\mathbf{w}})$ (sorting and extracting unique $\hat{\mathbf{v}}$ entries if necessary) using eq. (4.9). Obtain $\hat{\mathbf{k}}$ by deconvoluting $(\mathbf{k} \hat{*} \mathbf{h})$ with $\hat{\mathbf{h}}$. Return the outputs $\left(\hat{\mathbf{g}}, \hat{\mathbf{h}}, \hat{\mathbf{k}}, \hat{\phi}\right)$.

4.7 Complexity Analysis

The complexity of the proposed system identification algorithm is dominated by the solving of SDP (4.36) or (4.53). Denote $N_v := N_g + N_h + N_k$ with N_g, N_h and N_k being the lengths of the impulse responses of the FIR systems G, H and K in Figure 4-10. Also, denote $N_c := N(N-1)/2$ with N being the number of samples in the given problem data $(\mathbf{u}, \mathbf{y})$. Then with SeDuMi [76], the complexity of solving program (4.36) or (4.53) is $O\left(N_v^2 N_c^{2.5} + N_c^{3.5}\right)$ [118]. Typically, the number of samples N is much larger than the total number of impulse response samples N_v. Therefore, the complexity can be given as $O\left(N^7\right)$. As a result, there is a tradeoff between using many input/output measurement samples to accurately represent the system dynamics and using fewer samples to reduce the computation cost for solving the system identification problem.

4.8 Application Examples

4.8.1 Identification of randomly generated Wiener-Hammerstein system with feedback

The numerical example given in this subsection is the identification of the feedback setup. In this test case, G^*, H^* and K^* are randomly generated positive real passive FIR filters of 4th order. The nonlinearity is $\phi^* = \mathrm{sgn}(x)\{4|x|, 0.1|x| + (4 - 0.1)\}$. The noise is such that $n[t]$ is uniformly distributed and $n[t] \in [-0.01, 0.01]$ for all t.

For the identification, 86 samples of $(u[t], y[t])$ were used to construct the matrices $\mathbf{U}$ and $\mathbf{Y}$. The identification model has the same structure as in Figure 4-10, and the orders of the FIR filters are also four. Once the identification is completed, the original test system and the identified model are driven by some test signals (different from the training signals), and the corresponding outputs are recorded. Figure 4-12 shows the matching of the output of one of the test scenarios. Figure 4-13 shows the matching of the identified nonlinearity. The identification took about 5 seconds on a PC with a 3GHz CPU and 3GB of RAM.

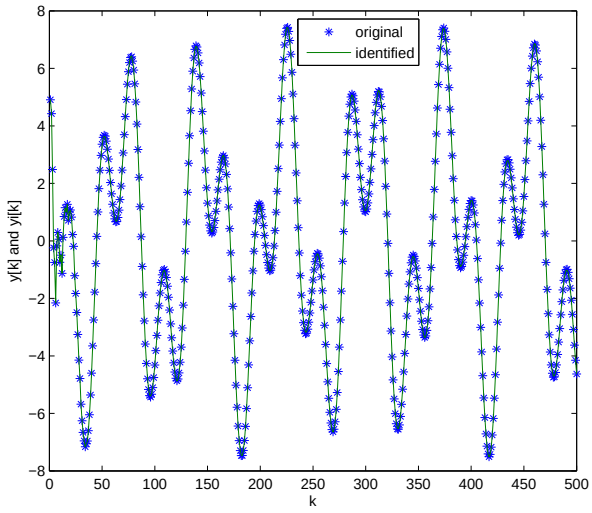


Figure 4-12: Matching of output signals by the original (unknown) system and the identified model. $y[k]$ denotes the output by the original system (star). $yi[k]$ denotes the output by the identified model (line). The plots of two output signals almost overlap.

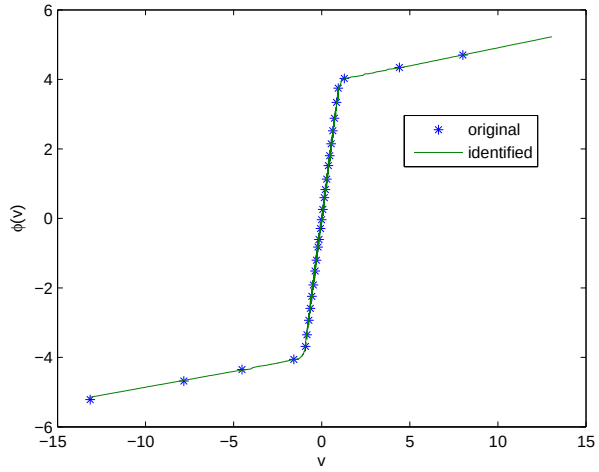


Figure 4-13: Matching of the original nonlinearity (star) and the identified nonlinearity (line).

4.8.2 Identification of a transmission line with diodes

The next application example in this section is the transmission line with diodes [83] (also described in Section 3.6). Figure 4-14 shows the circuit schematic. For simplicity, the resistance of all resistors is set to 0.1, the capacitance of all capacitors is set to 1 and all the

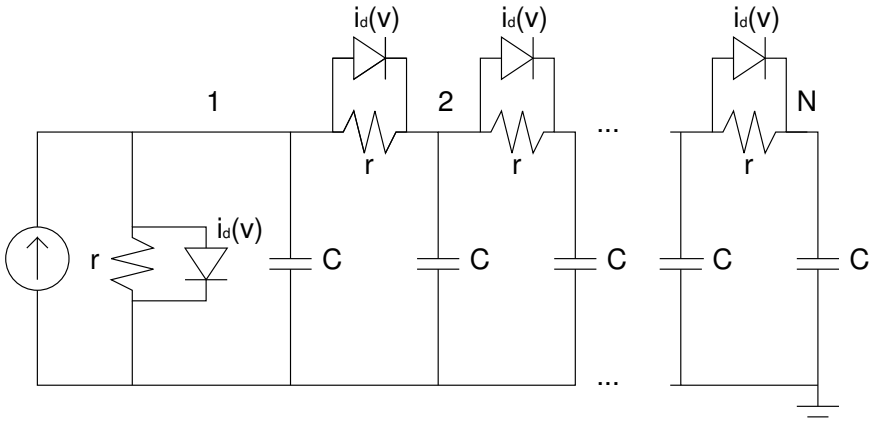


Figure 4-14: A transmission line with diodes.

diodes have the following input/output relationship: $i_d(v) = 10^{-6}\left(e^{40v} - 1\right)$. Excluding the ground node, there are N nodes in Figure 4-14 and in this subsection, N is assumed to be 30. The input of the transmission line system is the external current injected to node 1, and the output of the system is the voltage at node 1. While the transmission line system does not have the Wiener-Hammerstein structure, numerous investigations have suggested that it can be well approximated by very low order models.

210 input/output measurement samples from 7 different input/output pairs were used to construct a feedback Wiener-Hammerstein model based on Algorithm W-H feedback (noisy) in Section 4.6. The lengths of the impulse responses of G, H and K in Figure 4-10 are 1, 1 and 10, respectively. The construction of the Wiener-Hammerstein model took about 17 seconds on the PC with a 3GHz CPU and 3GB of RAM. After the model has been identified, a different set of input test signals were used to drive the model and the true transmission line system. Figure 4-15 shows the matching of the outputs of one of the test cases. While the transmission line does not have the Wiener-Hammerstein structure, the identified nonlinear does have a structure reminiscent of the exponential V-A characteristic of the diode. Figure 4-16 shows the *inverse* of the identified nonlinearity ϕ, which resembles the sum of a exponential function and a linear function.

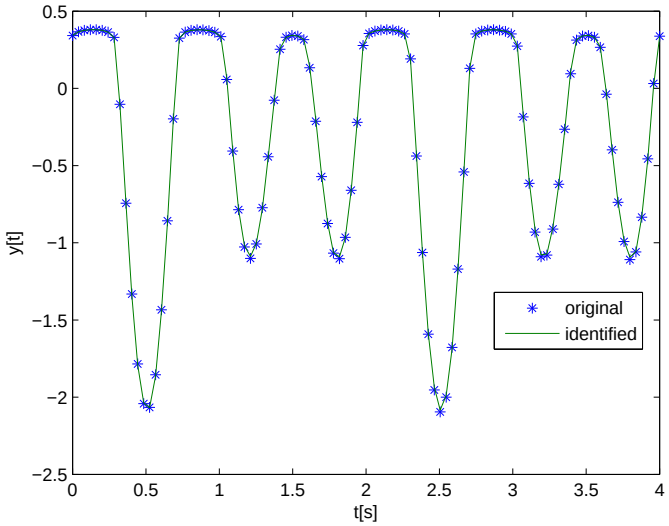


Figure 4-15: Matching of the output time sequences of the original transmission line system and the identified Wiener-Hammerstein model. Star: original system. Solid: identified model.

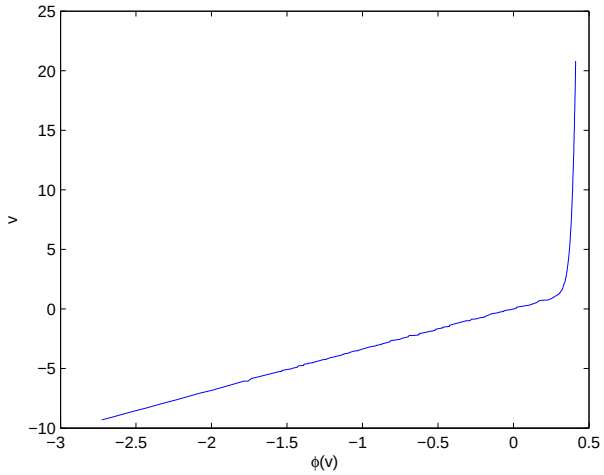


Figure 4-16: The inverse function of the identified nonlinearity ϕ. It looks like the exponential V-A characteristic with an added linear function.

4.8.3 Identification of an open loop operational amplifier

The last application example in this section is the identification of an open loop operational amplifier (OP-AMP) with a block diagram shown in Figure 4-17.

In the construction of the feedback Wiener-Hammerstein model, 300 input/output mea-

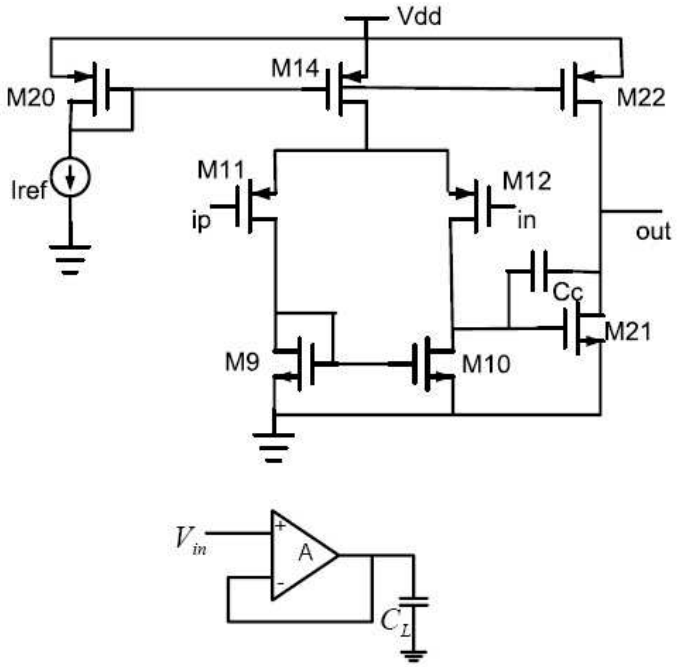


Figure 4-17: Block diagram of an operational amplifier.

surement samples from 6 different input/output pairs were used. The lengths of the impulse responses of G, H and K in Figure 4-10 are 1,1 and 2, respectively. The lengths of the impulse responses were chosen so that the Wiener-Hammerstein model can characterize the following first order system with a nonlinear pole.

$$k_0 y[t] + k_1 y[t-1] = \Psi(y[t]) + g_0 u[t]. \tag{4.63}$$

Eq. (4.63) fits in the feedback Wiener-Hammerstein structure depicted in Figure 4-18.

The construction of the model took about 26 seconds on the same 3GHz CPU machine used in the previous examples. Figure 4-19 shows the matching of the output of the true system simulated using SPICE and the output of the identified feedback Wiener-Hammerstein model simulated using MATLAB, when the test input signal is of relatively low frequency. On the other hand, Figure 4-20 shows the output matching for a test input signal of a relatively high frequency.

The identified nonlinear ϕ in the model in Figure 4-10 in Section 4.6 is shown in Figure 4-21.

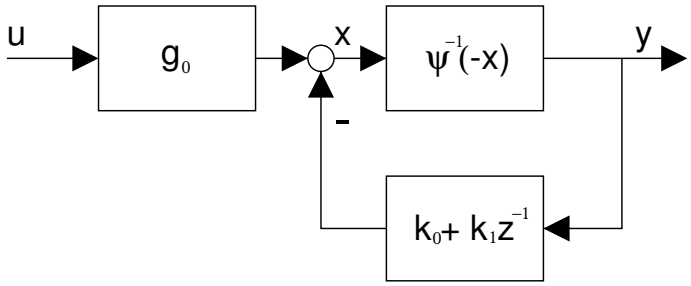


Figure 4-18: First order model for the OP-AMP. The pole of the model is a nonlinear function of the output *y*. The model fit in the feedback Wiener-Hammerstein structure discussed in this section.

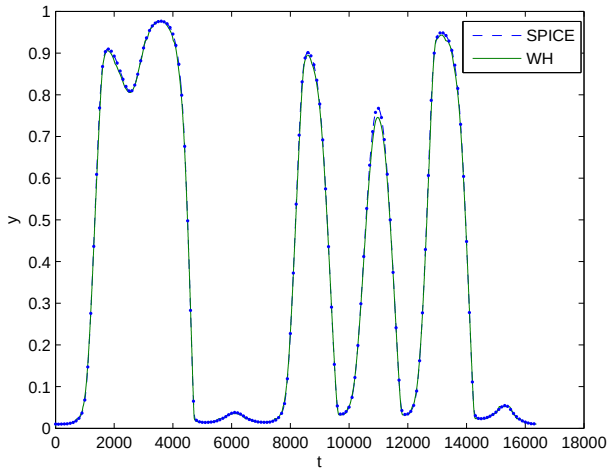
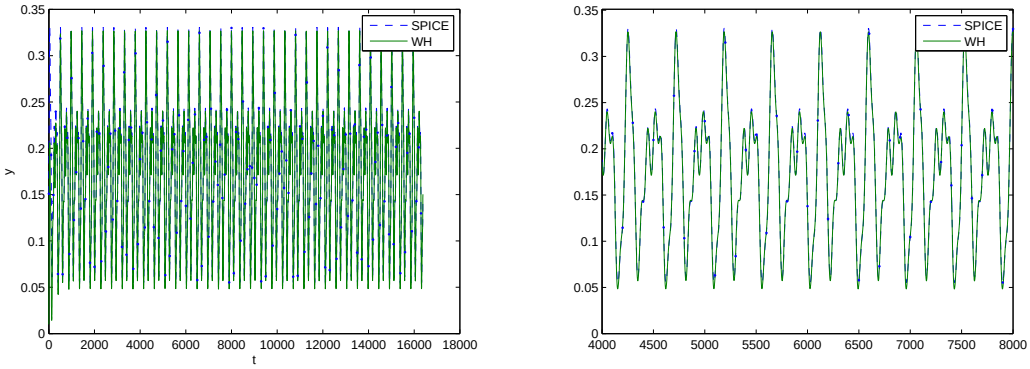


Figure 4-19: Matching of the output time sequence for a low frequency input test signal. Dash: SPICE simulated output time sequence. Dots: subset of samples of the SPICE simulated output. Solid: identified model.

4.9 Conclusion

In this chapter, the identification problems of the Wiener-Hammerstein system with and without feedback have been investigated. In the proposed algorithm, the identification of the nonlinearity is non-parametric. The chapter formulates the system identification problem as a non-convex QP. Nevertheless, it is demonstrated that the classical SDP relaxation is able to provide very good suboptimal solution to the formulated non-convex QP. Using a local search, high quality solutions of identification problem can often be found. Finally, a numerical example is given to show that the proposed relaxation framework provides an



(a) output time sequences (full time range) (b) output time sequences (zoomed in)

Figure 4-20: Matching of the output time sequence for a high frequency input test signal. Dash: SPICE simulated output time sequence. Dots: subset of samples of the SPICE simulated output. Solid: identified model.

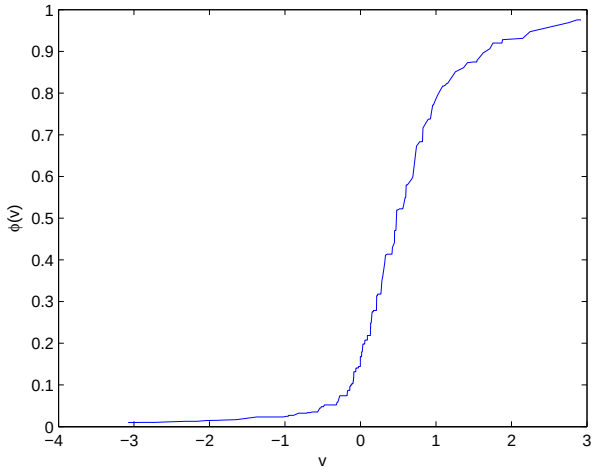


Figure 4-21: Identified nonlinearity ϕ in the feedback Wiener-Hammerstein model of Figure 4-10. Notice that there is a strong saturation for input values at the negative side, explaining the saturation phenomena in Figure 4-19.

interesting new way to solve the identification problem of the Wiener-Hammerstein system with feedback.

Chapter 5

Conclusions

The value of convex optimization in the field of model reduction has been demonstrated through three examples in three different parts of the thesis. In the first part of the thesis, quasi-convex optimization has been shown to provide a flexible framework to solve the LTI system model reduction problems. The proposed framework can handle stability, passivity constraints and it has been extended to solve the parameterized model reduction problem as well. A parameterized reduced model of a large spiral RF inductor has been constructed using the proposed algorithm. In the second part of the thesis, it has been shown that the problem of upper bounding the system input/ouput error due to nonlinear vector field approximation, a typical step in nonlinear model reduction algorithms, can be formulated as an L2 gain upper bounding problem to which the small gain theorem can be applied. Application of the small gain theorem led to a theoretical statement, as well as a numerical procedure describing the error bound. The classical example of a transmission line with diodes has been considered in the application of the proposed error bounding scheme. Finally in the third part of the thesis the nonlinear Wiener-Hammerstein system identification problem has been considered. While the Wiener-Hammerstein structure is simple, it has the potential to model important nonlinear sub-circuits, and the specific structure of Wiener-Hammerstein leads to special properties of the corresponding identification optimization problem, which has been demonstrated to be an easy non-convex QP. A SDP relaxation is presented to provide a good solution strategy to solve the non-convex QP. Wiener-Hammerstein reduced models of several practical circuits have been constructed

using the proposed identification scheme.

Bibliography

[1] B. Moore, "Principal Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction," *IEEE Transactions on Automatic Control*, vol. AC-26, no. 1, pp. 17–32, February 1981.

[2] L. Pernebo and L. Silverman, "Model reduction via balanced state space representations," *IEEE Transactions on Automatic Control*, vol. 27, no. 2, pp. 382–387, 1982.

[3] J. Phillips, L. Daniel, and M. Silverira, "Guaranteed passive balancing transformations for model order reduction," in *the 39th Conference on Design automation*, 2002.

[4] K. Glover, "All optimal Hankel-norm approximations of linear multivariable systems and their L^∞-error bounds," *International Journal on Control*, vol. 39, no. 6, pp. 1115–1193, June 1984.

[5] L. T. Pillage and R. A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis," *IEEE Transactions on Computer-Aided Design*, vol. 9, no. 4, pp. 352–366, April 1990.

[6] P. Feldmann and R. W. Freund, "Efficient linear circuit analysis by Padé approximation via the Lanczos process," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 14, pp. 639–649, 1995.

[7] E. Grimme, "Krylov projection methods for model reduction," Ph.D. dissertation, Coordinated-Science Laboratory, University of Illinois at Urbana-Champaign, Urbana-Champaign, IL, 1997.

[8] A. Odabasioglu, M. Celik, and L. Pileggi, "PRIMA: passive reduced-order interconnect marcomodeling algorithm," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 17, no. 8, pp. 645–654, August 1998.

[9] G. Berkooz, P. Holmes, and J. Lumley, "The proper orthogonal decomposition in the analysis of turbulent," *Ann. Rev. Fluid Mech*, vol. 25, pp. 539–575, 1993.

[10] J. Phillips, "Variational Interconnect Analysis via PMTBR," in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, 2004.

[11] D. Vasilyev and J. White, "A more reliable reduction algorithm for behavioral model extraction," in *IEEE International Conference on Computer-Aided Design*, 2005.

[12] Yunkai Zhou, "Numerical methods for large-scale matrix equations with applications in lti system model reduction," Ph.D. dissertation, Rice University, 2002.

[13] C. Coelho, J. Phillips, and L. Silveira, "A Convex Programming Approach to Positive Real Rational Approximation," in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, 2001 2001, pp. 4–8.

[14] B. Gustavsen and A. Semlyen, "Rational Approximation of Frequency Domain Responses by Vector Fitting," *IEEE Trans. on Power Delivery*, vol. 14, no. 3, pp. 1052–1061, July 1999.

[15] D. S. Weile, E. Michielssen, E. Grimme, and K. Gallivan, "A method for generating rational interpolant reduced order models of two-parameter linear systems," *Applied Mathematics Letters*, vol. 12, pp. 93–102, 1999.

[16] P. Gunupudi and M. Nakhla, "Multi-dimensional model reduction of VLSI interconnects," in *Proc. of the Custom Integrated Circuits Conference*, Orlando, FL, 2000, pp. 499–502.

[17] C. Prud'homme, D. Rovas, K. Veroy, Y. Maday, A. Patera, and G. Turinici, "Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bounds methods," *Journal of Fluids Engineering*, 2002.

[18] L. Daniel and J. White, "Automatic generation of geometrically parameterized reduced order models for integrated spiral RF-inductors," in *Behavioral Modeling and Simulation*, October 2003.

[19] P. Gunupudi, R. Khazaka, D. Saraswat, and M. Nakhla, "Closed-form parameterized simulation of high-speed transmission line networks using model-reduction techniques," in *IEEE MTT-S International Microwave Symposium*, 2004.

[20] L. Daniel, O. Siong, C. L., K. Lee, and J. White, "A multiparameter moment matching model reduction approach for generating geometrically parameterized interconnect performance models," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 23, no. 5, pp. 678–693.

[21] B. Bond and L. Daniel, "Parameterized model order reduction of nonlinear dynamical systems," in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, November 2005.

[22] X. Li, P. Li, and L. Pileggi, "Parameterized interconnect order reduction with explicit-and-implicit multi-parameter moment matching for inter/intra-die variations," in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, 2005.

[23] P. Li, F. Liu, S. Nassif, , and L. Pileggi, "Modeling interconnect variability using efficient parametric model order reduction," in *Design, Automation and Test Conference in Europe*, March 2005.

[24] Y. Chen, “Model order reduction for nonlinear systems,” M.S. Thesis, Massachusetts Institute of Technology, September 1999.

[25] M. Rewienski and J. K. White, “A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices.” in *Proc. of IEEE/ACM International Conference on Computer Aided-Design*, San Jose, CA, USA, November 2001, pp. 252–7.

[26] M. Rewienski and J. White, “A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices,” *IEEE Trans. Computer-Aided Design*, vol. 22, no. 2, pp. 155–70, Feb 2003.

[27] D. Vasilyev, M. Rewienski, and J. White, “A tbr-based trajectory piecewise-linear algorithm for generating accurate low-order models for nonlinear analog circuits and mems,” in *Proc. of the ACM/IEEE Design Automation Conference*, June 2003, pp. 490–5.

[28] N. Dong and J. Roychowdhury, “Piecewise polynomial nonlinear model reduction,” in *Proc. of the ACM/IEEE Design Automation Conference*, June 2003.

[29] ——, “Automated extraction of broadly applicable nonlinear analog macromodels from spice-level descriptions,” in *Proc. of the IEEE Custom Integrated Circuits Conference*, Oct. 2004.

[30] B. Bond and L. Daniel, “Parameterized model order reduction for nonlinear dynamical systems,” in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, San Jose, CA, 2005, pp. 487–494.

[31] M. Celik and A. Atalar and M. Tan, “Transient analysis of nonlinear circuits by combining asymptotic waveform evaluation with volterra series,” *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, vol. 42, no. 8, pp. 470–473, August 1995.

[32] J. Chen and S. M. Kang, “An algorithm for automatic model-order reduction of nonlinear MEMS devices,” in *Proceedings of ISCAS 2000*, 2000, pp. 445–448.

[33] J. R. Phillips, “Projection frameworks for model reduction of weakly nonlinear systems,” in 37^{th} *ACM/IEEE Design Automation Conference*, 2000, pp. 184–189.

[34] ——, “Automated extraction of nonlinear circuit macromodels,” in *Proceedings of the Custom Integrated Circuit Conference*, Orlando, FL, May 2000, pp. 451–454.

[35] J. Phillips, “Projection-based approaches for model reduction of weakly nonlinear, time-varying systems,” *IEEE Trans. Computer-Aided Design*, vol. 22, no. 2, pp. 171–87, 2003.

[36] P. Li and L. T. Pileggi, “Norm: Compact model order reduction of weakly nonlinear systems,” in *Proc. of the ACM/IEEE Design Automation Conference*, June 2003.

[37] L. Ljung, “Identification of nonlinear systems,” Department of Electrical Engineering, Linkping University, SE-581 83 Linkping, Sweden, Tech. Rep. LiTH-ISY-R-2784, June 2007.

[38] R. Haber and L. Keviczky, *Nonlinear System Identification - Input-Output Modeling Approach: Volume 1: Nonlinear System Parameter Identification*. Springer, 1999.

[39] J. Pedro and S. Maas, “A comparative overview of microwave and wireless power-amplifier behavioral modeling approaches,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 4, April 2005.

[40] M. Isaksson and D. Wisell and D. Rönnow, “A comparative analysis of behavioral models for rf power amplifiers,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 1, 2006.

[41] P. Crama and Y. Rolain, “Broadband measurement and identification of a Wiener-Hammerstein model for an RF amplifier,” in *ARFTG Conference*, 2002, pp. 49–57.

[42] E.-W. Bai, “An Optimal Two Stage Identification Algorithm for Hammerstein-Wiener Nonlinear Systems,” in *American Control Conference*, June 1998.

[43] ——, “A blind approach to the hammersteinwiener model identication,” *Automatica*, vol. 38, pp. 967–979, 2002.

[44] J. Schoukens, T. Dobrowiecki, and R. Pintelon, “Parametric and Nonparametric Identification of Linear Systems in the Presence of Nonlinear DistortionsA Frequency Domain Approach,” *IEEE Transcation on Automatic Control*, vol. 43, no. 2, pp. 176–190, 1998.

[45] M. Boutayeb and M. Darouach, “Recursive identification method for miso wiener hammerstein model,” *IEEE Transactions on Automatic Control*, vol. 40, no. 2, pp. 287–291, February 1995.

[46] M. Kozek and C. Hametner, “Block-oriented identification of Hammerstein/Wiener-models using the RLS-algorithm,” *International Journal of Applied Electromagnetics and Mechanics*, vol. 25, pp. 529–535, 2007.

[47] A. Tan and K. Godfrey, “Identification of WienerHammerstein Models Using Linear Interpolation in the Frequency Domain (LIFRED),” *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 3, pp. 509–521, June 2002.

[48] F. Ding, Y. Shi, and T. Chen, “Auxiliary model-based least-squares identification methods for Hammerstein output-error systems,” *Systems & Control Letters*, vol. 56, pp. 373–380, 2007.

[49] S. Pullela, N. Menezes, and L. Pileggi, “Moment-sensitivity-based wire sizing for skew reduction in on-chip clock nets,” *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, no. 2, pp. 210–215, February 1997.

[50] Y. Liu, L. T. Pileggi, and A. J. Strojwas, "Model order-reduction of RCL interconnect including variational analysis," in *Proc. of the IEEE/ACM Design Automation Conference*, 1999, pp. 210–206.

[51] P. Heydari and M. Pedram, "Model reduction of variable-geometry interconnects using variational spectrally-weighted balanced truncation," in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, San Jose, CA, November 2001.

[52] H. Liu, A. Singhee, R. A. Rutenbar, and L. R. Carley, "Remembrance of circuits past: macromodeling by data mining in large analog design spaces," in *Proc. of the IEEE/ACM Design Automation Conference*, June 2002, pp. 437–442.

[53] J. Lim and A. Oppenheim, *Advanced Topics in Signal Processing*. Prentice Hall, 1988.

[54] W. Beyene and Schutt-Aine, "Efficient Transient Simulation of High-Speed Interconnects Characterized by Sampled Data," *IEEE Trans on Components, Packaging, and Manufacturing Technology-Part B*, vol. 21, no. 1, pp. 105–114, February 1998.

[55] C. Coelho, J. Phillips, and L. Silveira, "Robust Rational Function Approximation Algorithm for Model Generation," in *Proc. of the IEEE/ACM Design Automation Conference*, June 1999, pp. 207–212.

[56] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. Philadelphia, PA: SIAM, June 1994, vol. 15.

[57] C. Coelho, J. Phillips, and L. Silveira, "Optimization Based Passive Constrained Fitting," in *Proc. of the IEEE/ACM International Conference on Computer-Aided Design*, San Jose, California, November 2002, pp. 10–14.

[58] B. Anderson, M. Mansour, and F. Kraus, "A New Test For Strict Positive Realness," *IEEE Trans on Circuit and Systems I: Fundamental Theory and Applications*, vol. 42, pp. 226–229, 1995.

[59] T. Kailath, *Linear Systems*. Prentice-Hall, 1980.

[60] R.T. Rockafellar, *Convex Analysis*. Princeton University Press, 1970.

[61] B. Bertsekas, A. Nedić, and A. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific, 2003.

[62] K. Zhou, J. Doyle, and K. Glover, *Robust and Optimal Control*, 1996.

[63] A. Megretski, "H-Infinity Model Reduction with Guaranteed Suboptimality Bound," in *American Control Conference*, 2006.

[64] R. Bland, D. Goldfarb, and M. Todd, "The Ellipsoid Method: A Survey," *Operation Research*, vol. 29, no. 6, pp. 1039–1091, Nov-Dec 1981.

[65] D. Bertsimas and J. Tsitsiklis, *Introduction to Linear Optimization*, 1997.

[66] J.-L. Goffin and J.-P. Vial, "Convex Nondifferentiable Optimization: A Survey Focussed on the Analytic Center Cutting Plane Method," *Optimization Methods and Software*, vol. 6, pp. 805–867, 2002.

[67] Y. Nesterov and A. Nemirovsky, *Interior Point Polynomial Algorithms in Convex Programming*. Society for Industrial and Applied Mathematics, 1994.

[68] A. Megretski, MIT 6.245 course reader, 2005.

[69] S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo, *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, Available from `http://www.cds.caltech.edu/sostools` and `http://www.mit.edu/~parrilo/sostools`, 2004.

[70] B. Dumitrescu, *Positive Trigonometric Polynomials and Signal Processing Applications*. Springer, 2007.

[71] M.A. Dritschel, "On Factorization of Trigonometric Polynomials," *Integral Equations and Operator Theory*, vol. 49, no. 1, pp. 11–42, 2004.

[72] A. Megretski, "Positivity of Trigonometric Polynomials," in *IEEE Conference on Decision and Control*, vol. 4, December 2003, pp. 3814–3817.

[73] P. Parrilo, "Structed Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization," Ph.D. dissertation, California Institute of Technology, 2000.

[74] M. Choi, T. Lam, and B. Reznick, "Sum of squares of real polynomials," in *Proceedings of Symposia in Pure Mathematics*, vol. 58, no. 2, 1995.

[75] V. Powers and T. Wörmann, "An algorithm for sums of squares of real polynomials," *Journal of Pure and Applied Linear Algebra*, vol. 127, 1998.

[76] J. Sturm, "Using SeDuMi 1.02, a MATLAB Toolbox for Optimization over Symmetric Cones," Tilburg University, Tech. Rep., 2001.

[77] Z. Zhu, B. Song, and J. White, "Algorithms in fastimp: A fast and wideband impedance extraction program for complicated 3-D geometries," in *Proc. of the IEEE/ACM Design Automation Conference*, 2003.

[78] S. Kapur and D. Long, "Ies3: Efficient electrostatic and electromagnetic simulation," *IEEE Computational Science and Engineering*, vol. 05, no. 4, pp. 60–67, 1998.

[79] T. E. Moselhy, X. Hu, and L. Daniel, "pFFT in FastMaxwell: A Fast Impedance Extraction Solver for 3D Conductor Structures over Substrate."

[80] D. Vasilyev and J. White, RLE Internal Memorandum, MIT, 2005.

[81] J. Peters, "Design of High Quality Factor Inductos in RF MCM-D," Master's thesis, MIT, 2004.

[82] X. Hu, J. White, and L. Daniel, "Analysis of full-wave conductor system impedance over substrate using novel integration techniques," in *Proc. of the IEEE/ACM Design Automation Conference*, June 2005.

[83] M. Rewienski and J. White, "A Trajectory Piecewise-linear Approach to Model Order Reduction and Fast Simulation of Nonlinear circuits and Micromachined Devices," *IEEE Trans. Computer-Aided Design*, vol. 22, pp. 155–170, 2003.

[84] J. Phillips and J. Afonso and A. Oliveria and L. Silverira, "Analog macromodeling using kernel methods," in *2003 IEEE/ACM ICCAD*, 2003, pp. 446–453.

[85] J. Phillips, "Projection-based approaches for model reduction of weakly nonlinear, time-varying systems," *IEEE Trans. Computer-Aided Design*, vol. 22, pp. 171–187, 2003.

[86] P. Li and L. Pileggi, "NORM: Compact model order reduction of weakly nonlinear systems," in *40th ACM/IEEE Design Automation Conference*, June 2003, pp. 472–477.

[87] K. Zhou, *Essentials of Robust Control*. Prentice Hall, 1998.

[88] A. Megretski and A. Rantzer, "System Analysis via Integral Quadratic Constraints," *IEEE Trans. Automatic Control*, vol. 42, no. 6, June 1997.

[89] B. Bond and L. Daniel, "A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 12, pp. 2116–2129, Dec 2007.

[90] N. Dong and J. Roychowdhury, "General purpose nonlinear model-order reduction using piecewise-polynomial representations," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 2, pp. 249–264, Feb 2008.

[91] L. Ljung, *System Identification, Theory for the User*, 2nd ed. Prentice Hall, 1999.

[92] S. Boyd, "Volterra Series: Engineering Fundamentals," Ph.D. dissertation, University of California, Berkeley, 1985.

[93] S. Maas, "How to model intermodulation distortion," in *Microwave Symposium Digest, 1991., IEEE MTT-S International*, Boston, MA, 1991, pp. 149–151.

[94] A. Soury, E. Ngoya, and J. Rousset, "Behavioral modeling of RF and microwave circuit blocks for hierarchical simulation of modern transceivers," in *Microwave Symposium Digest, 2005 IEEE MTT-S International*, 2005, pp. 975–978.

[95] S. Billings and S. Fakhouri, "Identification of a class of nonlinear systems using correlation analysis," *Proceedings of Institution of Electrical Engineers*, vol. 125, pp. 691–697, July 1978.

[96] W. Greblicki, "Nonparametric identification of wiener systems," IEEE Transcations on Information Theory, vol. 38, no. 5, pp. 1487–1493, September 1992.

[97] K. Hsu, T. Vincent, and K. Poolla, "Identification of Nonlinear Maps in Interconnected Systems," in *the 44th IEEE Conference on Decision and Control*, December 2005, pp. 6430–6435.

[98] A. Hagenblad, L. Ljung, and A. Wills, "Maximum Likelihood Identification of Wiener Models," November 2007, submitted to Automatica.

[99] R. Smith, "Model Validation for Uncertain Systems," Ph.D. dissertation, California Institute of Technology, 1990.

[100] R. Smith and J. Doyle, "Model Validation: A Connection Between Robust Control and Identification," *IEEE Trans. on Automatic Control*, vol. 37, no. 7, pp. 942–952, July 1992.

[101] K. Poolla, P. Khargonekar, A. Tikku, J. Krause, and K. Nagpal, "A Time-Domain Approach to Model Validation," *IEEE Trans. on Automatic Control*, vol. 39, no. 5, pp. 951–959, May 1994.

[102] R. Smith and G. Dullerud, "Modeling and Validation of Nonlinear Feedback Systems," in *Robustness in Identification and Control*. Springer-Verlag, 1999, pp. 87–101.

[103] R. Smith and J. Doyle, "Model Invalidation: A Connection between Robust Control and Identification," in *American Control Conference*, 1989, pp. 1435–1440.

[104] M. Newlin and R. Smith, "A Generalization of the Structured Singular Value and Its Application to Model Validation," *IEEE Trans. on Automatic Control*, vol. 43, no. 7, pp. 901–907, July 1998.

[105] R. Smith and G. Dullerud, "Continuous-Time Control Model Validation Using Finite Experimental Data," *IEEE Trans. on Automatic Control*, vol. 41, no. 8, pp. 1094–1105, August 1996.

[106] T. Zhou and H. Kimura, "Time domain identification for robust control," *Systems & Control Letters*, vol. 20, pp. 167–178, 1993.

[107] R. Kosut, M. Lau, and S. Boyd, "Set-Membership Identification of Systems with Parametric and Nonparametric Uncertainty," *IEEE Trans. on Automatic Control*, vol. 37, no. 7, pp. 929–941, July 1992.

[108] Q. Zhang, A. Iouditski, and L. Ljung, "Identification of wiener systems with monotonous nonlinearity," Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden, Tech. Rep. LiTH-ISY-R-2787, 2007.

[109] S. Sastry, *Nonlinear Systems*. Springer, 1999.

[110] V. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 1998.

[111] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001.

[112] C. De Boor, *A Practical Guide to Splines*. Springer, 2001.

[113] M. Goemans and D. Williamson, "Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming," *Journal of ACM*, vol. 42, pp. 1115–1145, 1995.

[114] M. Fazel, H. Hindi, and S. Boyd, "Rank Minimization and Applications in System Theory," in *Proceedings of American Control Conference*, Boston, Massachusetts, June 2004, pp. 3273–3278.

[115] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

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

[117] B. Alkire and L. Vandenberghe, "Convex optimization problems involving finite autocorrelation sequences," *Mathematical Programming, Series A*, vol. 93, no. 3, pp. 331–359, 2002.

[118] Y. Labit, D. Peaucelle, and D. Henrion, "SEDUMI INTERFACE 1.02 A tool for solving LMI problems with SEDUMI," in *IEEE International Symposium on Computer Aided Control System Design*, 2002, pp. 272–277.