An interpolation-based approach to the weighted-$\mathcal{H}_2$ model reduction problem

Branimir Anic

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mathematics

Dr. Serkan Gugercin, Chair
Dr. Christopher A. Beattie
Dr. Jeffrey T. Borggaard

August 25, 2008
Blacksburg, Virginia

Keywords: $\mathcal{H}_2$ Approximation, Rational Krylov, interpolation, model reduction

An interpolation-based approach to the weighted-$\mathcal{H}_2$ model reduction problem

Branimir Anic

Abstract

Dynamical systems and their numerical simulation are very important for investigating physical and technical problems. The more accuracy is desired, the more equations are needed to reach the desired level of accuracy. This leads to large-scale dynamical systems. The problem is that computations become infeasible due to the limitations on time and/or memory in large-scale settings. Another important issue is numerical ill-conditioning. These are the main reasons for the need of model reduction, i.e. replacing the original system by a reduced system of much smaller dimension. Then one uses the reduced models in order to simulate or control processes.

The main goal of this thesis is to investigate an interpolation-based approach to the weighted-$\mathcal{H}_2$ model reduction problem. Nonetheless, first we will discuss the regular (unweighted) $\mathcal{H}_2$ model reduction problem. We will re-visit the interpolation conditions for $\mathcal{H}_2$-optimality, also known as Meier-Luenberger conditions, and discuss how to obtain an optimal reduced order system via projection. After having introduced the $\mathcal{H}_2$-norm and the unweighted-$\mathcal{H}_2$ model reduction problem, we will introduce the weighted-$\mathcal{H}_2$ model reduction problem. We will first derive a new error expression for the weighted-$\mathcal{H}_2$ model reduction problem. This error expression illustrates the significance of interpolation at the mirror images of the reduced system poles and the original system poles, as in the unweighted case. However, in the weighted case this expression yields that interpolation at the mirror images of the weighting system is also significant. Finally, based on the new weighted-$\mathcal{H}_2$ error expression, we will propose an iteratively corrected interpolation-based algorithm for the weighted-$\mathcal{H}_2$ model

reduction problem. Moreover we will present new optimality conditions for the weighted-$\mathcal{H}_2$ approximation. These conditions occur as structured orthogonality conditions similar to those for the unweighted case which were derived in [26].

We present several numerical examples to illustrate the effectiveness of the proposed approach and compare it with the frequency-weighted balanced truncation method. We observe that, for virtually all of our numerical examples, the proposed method outperforms the frequency-weighted balanced truncation method.

Acknowledgments

The first people I would like to thank are Prof. Scherer (University of Karlsruhe, Germany) and Prof. Herdman (Virginia Polytechnic Institute and State University, USA) for establishing the exchange program between the mathematics departments of University of Karlsruhe and Virginia Polytechnic Institute and State University, and thus giving me the possibility of getting my Masters degree from Virginia Polytechnic Institute and State University.

Moreover I would like to express my gratitude to Prof. Beattie, Prof. Drmac and Prof. Gugercin (all three Virginia Polytechnic Institute and State University, USA) who always had time to listen to my concerns about my thesis and my research. Especially Prof. Gugercin had to suffer of my endless questions. Although being very busy himself, he always had time to help me with good advice.

Finally I want to thank my family and Marijana for their endless support and patience during the last years.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter we introduce the $\mathcal{H}_2$ model reduction problem, show how to obtain the reduced order model via projection, and present interpolation-based model reduction. For a general overview of model reduction, we refer to [5].

1.1 Linear Dynamical Systems

Dynamical systems are used to describe physical and technical phenomena. Differential equations describe the behaviour and the properties of these phenomena. Nowadays in industrial research centers many physical and technical problems need to be simulated or controlled. The reason for the need of simulation is obvious if we just compare the costs of real car crash tests and the simulated ones. Control of large flexible structures is an example where model reduction is needed to control systems. Since a controller has to react quickly, it is obvious that a system of (much) lower order is an advantage. The desire for high accuracy leads to an increasing number of equations; this means the order of the underlying dynamical system increases. Since the computational resources are limited, the

original system of equations has to be replaced by a (much) smaller set of equations. In addition to the limitations on computational resources, increasing inaccuracy and numerical ill-conditioning also become an issue in large scale settings. These are the main reasons why model reduction is needed.

1.2 Model Reduction

Consider a dynamical system which is described by a set of first order differential equations. The goal of model reduction is to replace the original set of equations with a (much) smaller set of equations. In this thesis we consider single input/ single output (SISO) linear dynamical systems, which are represented in state-space as

$$G : \begin{cases} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}\, u(t) \\ y(t) &= \mathbf{c}^T \mathbf{x}(t) \end{cases}, \tag{1.2.1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{b}, \mathbf{c} \in \mathbb{R}^n$; $\mathbf{x}(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$, $y(t) \in \mathbb{R}$, are respectively the *state*, *input*, and *output* of the system.

In order to explain what the transfer function of a system is, we need the definition of the Laplace transform.

Definition 1.2.1. *Let f(t) be a function defined for t ≥0. Then the unilateral Laplace transform of f(t) is the function F(s), defined by*

$$F(s) := \mathscr{L}(f(t)) = \int_0^\infty f(t)e^{-st}dt. \tag{1.2.2}$$

An important property of the Laplace transform is that $\mathscr{L}(\dot{f}(t)) = s \cdot \mathscr{L}(f(t)) = s \cdot F(s)$. This property and other properties of the Laplace transform can be found in [30]. Applying the Laplace transform to (1.2.1) yields

$$s \cdot \mathbf{X}(s) = \mathbf{A}\mathbf{X}(s) + \mathbf{b}\,U(s) \text{ and } Y(s) = \mathbf{c}^T\mathbf{X}(s). \tag{1.2.3}$$

We obtain the relation $Y(s) = G(s)U(s)$ from (1.2.3), where $\mathbf{X}(s) = \mathscr{L}(\mathbf{x}(t))$, $U(s) = \mathscr{L}(u(t))$, $Y(s) = \mathscr{L}(y(t))$, and

$$G(s) = \mathbf{c}^T(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}.$$

The function $G(s)$ is called the transfer function of the system. It is common to denote both the system and the transfer function with G. Assuming that the system G is stable is equivalent to assuming that all eigenvalues of $\mathbf{A}$ are contained in the left halfplane. The dimension of $\mathbf{A}$ is the same as the dimension of the *state* $\mathbf{x}(t)$. This means the dimension of G is n. After reducing the original system G, we obtain a new system G_r:

$$G_r : \begin{cases} \dot{\mathbf{x}}_r(t) &= \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{b}_r u(t) \\ y_r(t) &= \mathbf{c}_r^T\mathbf{x}_r(t) \end{cases} \quad \text{or} \quad G_r(s) = \mathbf{c}_r^T(s\mathbf{I} - \mathbf{A}_r)^{-1}\mathbf{b}_r \tag{1.2.4}$$

where $\mathbf{A}_r \in \mathbb{R}^{r\times r}$ and $\mathbf{b}_r$, $\mathbf{c}_r \in \mathbb{R}^r$ with $r \ll n$. In order to be able to use G_r in place of G in simulations and control, G_r should be a high fidelity approximation to G. This means that for the outputs $y(t)$ and $y_r(t)$, we want to hold $y_r(t) \approx y(t)$ over a large class of inputs $u(t)$; i.e. $\max_{t>0}|y(t) - y_r(t)|$ to be small uniformly over all inputs $u(t)$. Assume that the input $u(t)$ has bounded energy, which can be described as $\int_0^\infty |u(t)|^2\, dt \leq 1$. The Laplace transform gives us $Y(s) = G(s)U(s)$ and $Y_r(s) = G_r(s)U(s)$, which yields $Y(s) - Y_r(s) = [G(s) - G_r(s)]\, U(s)$.

As shown in [26], this yields following:

$$\begin{aligned} \max_{t>0}|y(t) - y_r(t)| &= \max_{t>0}\left|\tfrac{1}{2\pi}\int_{-\infty}^{\infty}(Y(\imath\omega) - Y_r(\imath\omega))\, e^{\imath\omega t}\, d\omega\right| \\ &\leq \frac{1}{2\pi} \int_{-\infty}^{\infty}|Y(\imath\omega) - Y_r(\imath\omega)|\, d\omega = \tfrac{1}{2\pi}\int_{-\infty}^{\infty}|G(\imath\omega) - G_r(\imath\omega)|\, |U(\imath\omega)|\, d\omega \\ &\leq \left(\tfrac{1}{2\pi}\int_{-\infty}^{\infty}|G(\imath\omega) - G_r(\imath\omega)|^2\, d\omega\right)^{1/2}\left(\tfrac{1}{2\pi}\int_{-\infty}^{\infty}|U(\imath\omega)|^2\, d\omega\right)^{1/2} \\ &\leq \left(\tfrac{1}{2\pi}\int_{-\infty}^{\infty}|G(\imath\omega) - G_r(\imath\omega)|^2\, d\omega\right)^{1/2}\left(\int_0^{\infty}|u(t)|^2\, dt\right)^{1/2} \\ &\leq \left(\tfrac{1}{2\pi}\int_{-\infty}^{+\infty}|G(\imath\omega) - G_r(\imath\omega)|^2\, d\omega\right)^{1/2}. \end{aligned} \tag{1.2.5}$$

(1.2.5) defines the $\mathcal{H}_2$ norm as $\|G\|_{\mathcal{H}_2} := \left(\tfrac{1}{2\pi}\int_{-\infty}^{+\infty}|G(\imath\omega)|^2\, d\omega\right)^{1/2}$, and due to (1.2.5) we obtain

$$\max_{t>0} |y(t) - y_r(t)| \leq \|G - G_r\|_{\mathcal{H}_2},$$

for all $u(t)$ such that $\int_0^\infty |u(t)|^2 \, dt \leq 1$. Since the goal is to have $y_r(t) \approx y(t)$ over a large class of inputs $u(t)$, this means we want $\|G - G_r\|_{\mathcal{H}_2}$ to be as small as possible, because then $\max_{t>0} |y(t) - y_r(t)|$ will be small for bounded energy inputs. We note that, in addition to have a small error norm $\|G - G_r\|_{\mathcal{H}_2}$, the computation of the reduced system G_r should be feasible; this means the costs of computing $\mathbf{A}_r$, $\mathbf{b}_r$, and $\mathbf{c}_r$ should not be too high and the algorithm should be stable. This is of critical importance in large-scale settings.

Due to the fact that finding a global minimizer of $\|G - G_r\|_{\mathcal{H}_2}$ is a hard task, the common approach, which has also been done in [11, 27, 29, 33, 34, 47], is to find a reduced order model G_r that satisfies necessary conditions for local optimality. Recently, Gugercin *et. al* [26] introduced an "Iterative Rational Krylov Algorithm" to solve the optimal $\mathcal{H}_2$ approximation problem in an effective way in large-scale settings. In this algorithm, the necessary conditions that the reduced system G_r interpolates the original system G at the mirror images of the poles of G_r, are being used in order to compute G_r. We will focus on this method in Chapter 2. This approach will be our starting point in developing the proposed method for weighted-$\mathcal{H}_2$ model reduction in Chapter 4.

1.3 Model Reduction by Projection and Interpolation

We want to replace the original system $G(s)$ by a reduced order system $G_r(s)$. In this section we will show how to obtain $\mathbf{A}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$ via projection. The reduced order model $G_r(s)$

will be constructed by Galerkin approximation. Assume $\mathcal{V}_r$ and $\mathcal{W}_r$ are given r-dimensional subspaces of $\mathbb{R}^n$. Moreover assume that for these spaces $\mathcal{V}_r \cap \mathcal{W}_r^{\perp} = \{0\}$ holds. Then we define for any input $u(t)$ the reduced-order output $y_r(t)$ as follows:

$$\begin{aligned} &\text{If} \quad \mathbf{v}(t) \in \mathcal{V}_r \quad \text{solves} \quad \dot{\mathbf{v}}(t) - \mathbf{A}\mathbf{v}(t) - \mathbf{b}u(t) \perp \mathcal{W}_r \quad \text{for all } t \\ &\qquad \text{then we define} \quad y_r(t) \stackrel{\text{def}}{=} \mathbf{c}^T\mathbf{v}(t). \end{aligned} \tag{1.3.1}$$

Assume $\mathbf{V}_r \in \mathbb{R}^{n\times r}$ and $\mathbf{W}_r \in \mathbb{R}^{n\times r}$ are matrices such that $\mathcal{V}_r = \text{Range}(\mathbf{V}_r)$ and $\mathcal{W}_r = \text{Range}(\mathbf{W}_r)$. Since $\mathcal{V}_r \cap \mathcal{W}_r^{\perp} = \{0\}$, $\mathbf{W}_r^T\mathbf{V}_r$ is invertible. The Galerkin approximation (1.3.1) can be interpreted as $\mathbf{v}(t) = \mathbf{V}_r\mathbf{x}_r(t)$ with $\mathbf{x}_r(t) \in \mathbb{R}^r$ for each t. Then $\dot{\mathbf{v}}(t) = \mathbf{V}_r\dot{\mathbf{x}}_r(t)$. Since $\dot{\mathbf{v}}(t) - \mathbf{A}\mathbf{v}(t) - \mathbf{b}u(t) \perp \mathcal{W}_r$ has to hold for all t and $\mathcal{W}_r = \text{Range}(\mathbf{W}_r)$ we obtain

$$\mathbf{W}_r^T(\dot{\mathbf{v}}(t) - \mathbf{A}\mathbf{v}(t) - \mathbf{b}u(t)) = 0,$$

which is equivalent to

$$\mathbf{W}_r^T(\mathbf{V}_r\dot{\mathbf{x}}_r(t) - \mathbf{A}\mathbf{V}_r\mathbf{x}_r(t) - \mathbf{b}u(t)) = \mathbf{W}_r^T\mathbf{V}_r\dot{\mathbf{x}}_r(t) - \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r\mathbf{x}_r(t) - \mathbf{W}_r^T\mathbf{b}u(t) = 0.$$

Now consider (1.2.4) to see that we have to solve this equation for $\dot{\mathbf{x}}_r(t)$ in order to get $\mathbf{A}_r$,

$\mathbf{b}_r$ and $\mathbf{c}_r$. Solving the equation above for $\dot{\mathbf{x}}_r(t)$ yields $\mathbf{A}_r$, $\mathbf{b}_r$ and $\mathbf{c}_r$:

$$\begin{aligned}\dot{\mathbf{x}}_r(t) &= (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{A}\mathbf{V}_r\mathbf{x}_r(t) + (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{b}u(t)\\ \Rightarrow \mathbf{A}_r &= (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{A}\mathbf{V}_r\ ,\ \mathbf{b}_r = (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{b}\\ \text{and with}\quad y_r(t)&=\mathbf{c}^T\mathbf{v}(t) = \mathbf{c}^T\mathbf{V}_r\mathbf{x}_r(t) \text{ we get } \mathbf{c}_r^T = \mathbf{c}^T\mathbf{V}_r.\end{aligned}$$

This yields the following reduced order model via projection:

$$\mathbf{A}_r = (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{A}\mathbf{V}_r\ ,\ \mathbf{b}_r = (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{b} \text{ and } \mathbf{c}_r^T = \mathbf{c}^T\mathbf{V}_r \tag{1.3.2}$$

We constructed a projector $\Pi = \mathbf{V}\mathbf{Z}^T$, where $\mathbf{V}, \mathbf{Z} \in \mathbb{R}^{n\times r}$ with $\mathbf{Z}^T\mathbf{V} = \mathbf{I}_r$ and

$$\dot{\mathbf{x}}_r = \underbrace{\mathbf{Z}^T\mathbf{A}\mathbf{V}}_{:=\mathbf{A}_r}\mathbf{x}_r(t) + \underbrace{\mathbf{Z}^T\mathbf{b}}_{:=\mathbf{b}_r}\mathbf{u}(t),\quad \mathbf{y}_r(t) = \underbrace{\mathbf{c}\mathbf{V}}_{:=\mathbf{c}_r}\mathbf{x}_r(t).$$

The underlying oblique projector is

$$\Pi = \mathbf{V}_r(\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T.$$

Now we introduce model reduction by *moment matching*. Model reduction by *moment matching* means we want to construct a system G_r, as in (1.2.4), such that $G_r(s)$ interpolates

the values of $G(s)$ at selected interpolation points which are called *shifts*. We will denote these shifts by σ_k. Sometimes it is desired that G_r not only interpolates G at these shifts but also the derivatives, which means simple Hermite interpolation is desired. If we consider (1.2.4), we see that G_r is determined by $\mathbf{A}_r$, $\mathbf{b}_r$, and $\mathbf{c}_r$. This means we want to find $\mathbf{A}_r$, $\mathbf{b}_r$, and $\mathbf{c}_r$ such that

$$G_r(\sigma_k) = G(\sigma_k) \quad \text{and} \quad G_r'(\sigma_k) = G'(\sigma_k) \quad \text{for} \quad k = 1, \ldots, r$$

or equivalently

$$\mathbf{c}^T(\sigma_k\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} = \mathbf{c}_r^T(\sigma_k\mathbf{I} - \mathbf{A}_r)^{-1}\mathbf{b}_r \ \text{ and } \ \mathbf{c}^T(\sigma_k\mathbf{I} - \mathbf{A})^{-2}\mathbf{b} = \mathbf{c}_r^T(\sigma_k\mathbf{I} - \mathbf{A}_r)^{-2}\mathbf{b}_r$$

for $k = 1, \ldots, r$.

However, the goal is to construct a reduced order system which interpolates the original system, without explicitly computing the moments. In order to unify the interpolation and the projection framework we present the following theorems which can be found with proofs in [19, 20, 26, 49, 50].

Lemma 1.3.1. *Suppose $\sigma \in \mathbb{C}$ is not an eigenvalue, neither of $\mathbf{A}$ nor of $\mathbf{A}_r$.*

$$\textit{If} \quad (\sigma\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} \in \mathcal{V}_r \quad \textit{then} \quad G_r(\sigma) = G(\sigma). \tag{1.3.3}$$

$$\textit{If} \quad (\overline{\sigma}\,\mathbf{I} - \mathbf{A}^T)^{-1}\mathbf{c} \in \mathcal{W}_r \quad \textit{then} \quad G_r(\sigma) = G(\sigma). \tag{1.3.4}$$

$$\textit{If both} \quad (\sigma\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} \in \mathcal{V}_r \quad \textit{and} \quad (\overline{\sigma}\,\mathbf{I} - \mathbf{A}^T)^{-1}\mathbf{c} \in \mathcal{W}_r$$
$$\textit{then} \quad G_r(\sigma) = G(\sigma) \quad \textit{and} \quad G_r'(\sigma) = G'(\sigma). \tag{1.3.5}$$

Corollary 1.3.2. *Consider the system G defined by $\mathbf{A}, \mathbf{b}, \mathbf{c}$, a set of distinct shifts given by $\{\sigma_k\}_{k=1}^r$, that is closed under conjugation, and subspaces spanned by the columns of $\mathbf{V}_r$ and $\mathbf{W}_r$ with*

$$Ran(\mathbf{V}_r) = span\left\{(\sigma_1\mathbf{I}-\mathbf{A})^{-1}\mathbf{b}, \quad \cdots, \quad (\sigma_r\mathbf{I}-\mathbf{A})^{-1}\mathbf{b}\right\} \quad \text{and} \tag{1.3.6}$$

$$Ran(\mathbf{W}_r) = span\left\{(\sigma_1\mathbf{I}-\mathbf{A}^T)^{-1}\mathbf{c}, \quad \cdots, \quad (\sigma_r\mathbf{I}-\mathbf{A}^T)^{-1}\mathbf{c}\right\}. \tag{1.3.7}$$

Then $\mathbf{V}_r$ and $\mathbf{W}_r$ can be chosen to be real matrices and the reduced order system G_r defined by $\mathbf{A}_r = (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{A}\mathbf{V}_r$, $\mathbf{b}_r = (\mathbf{W}_r^T\mathbf{V}_r)^{-1}\mathbf{W}_r^T\mathbf{b}$, $\mathbf{c}_r^T = \mathbf{c}^T\mathbf{V}_r$, is itself real and matches the first two moments of $G(s)$ at each of the interpolation points σ_k, i.e., $G(\sigma_k) = G_r(\sigma_k)$ and $G'(\sigma_k) = G_r'(\sigma_k)$, for $k = 1, \cdots, r$.

Lemma 1.3.1 tells us how to choose the spaces $\mathcal{V}_r$ and $\mathcal{W}_r$ and Corollary 1.3.2 establishes the connection between interpolation of transfer functions and how to obtain $\mathbf{A}_r$, $\mathbf{b}_r$, and $\mathbf{c}_r$ via projection. Notice that this way we match the moments without computing them explicitly. Corollary 1.3.2 can be generalized to match higher order derivatives of G. However, for our purposes matching G and G' will suffice. For the general case, see [20].

The goal of this thesis is to investigate the $\mathcal{H}_2$ model reduction problem for the weighted case. Electric systems work at certain frequencies, this means it is desirable to use a weighting system W which increases the accuracy of the reduced system at these special frequencies. For the unweighted case minimizing $\|G - G_r\|_{\mathcal{H}_2}$ is of interest. This becomes minimizing $\|(G - G_r) \cdot W\|_{\mathcal{H}_2}$ for the weighted case.

The remaining chapters of this thesis will be organized in the following way:

In Chapter 2, we will give a review of the $\mathcal{H}_2$-norm and discuss interpolation-based $\mathcal{H}_2$-model reduction. In Chapter 3, we will provide an introduction to the weighted-$\mathcal{H}_2$ model reduction problem and discuss what has been proposed by Halevi [27] to solve the weighted-$\mathcal{H}_2$ model reduction problem. The second part of Chapter 3 is crucial for the algorithm proposed in Chapter 4, where we will derive a new error expression for the weighted case. This new error expression shows that the weighted-$\mathcal{H}_2$ error is due to the mismatch of G and G_r at the mirror images of the poles of the original system G, the mirror images of the poles of the reduced system G_r and the mirror images of the poles of the weighting system W.

In the last section of Chapter 3 we will present new optimality conditions for the weighted-$\mathcal{H}_2$ approximation. These conditions occur as structured orthogonality conditions similar to those for the unweighted case which were derived in [26].

In Chapter 4 we propose a numerical algorithm for the weighted-$\mathcal{H}_2$ model reduction problem and present several numerical experiments for the proposed algorithm. Conclusions are given in Chapter 5. The matlab code of our algorithm is provided in the appendix.

Chapter 2

Unweighted-$\mathcal{H}_2$ model reduction

In this chapter, we will give a review of the $\mathcal{H}_2$ norm. Moreover we will re-visit the interpolation based optimality conditions for $\mathcal{H}_2$ model reduction. Finally, we will present the algorithm for the unweighted case which was suggested in [26]. Most of the discussions in this chapter are borrowed from [26].

2.1 The $\mathcal{H}_2$-norm

Consider the set of functions $g(z)$ which are analytic for z with $Re(z) > 0$. Moreover assume that these functions $g(z)$ are square integrable as a function of $y \in (-\infty, \infty)$ in such a way that

$$\sup_{x>0} \int_{-\infty}^{\infty} |g(x + \imath y)|^2 \, dy < \infty.$$

We will denote this set of functions by $\mathcal{H}_2$.

The reason why we are interested in the space $\mathcal{H}_2$ is that $\mathcal{H}_2$ is a Hilbert space and transfer

functions associated with stable singe input/ single output (SISO) finite dimensional dynamical systems are elements of $\mathcal{H}_2$. Assume $G(s)$ and $H(s)$ are transfer functions associated with real stable SISO dynamical systems. Then we can define the $\mathcal{H}_2$ inner product as

$$\begin{aligned}\langle G,\ H\rangle_{\mathcal{H}_2} &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \overline{G(\imath\omega)}H(\imath\omega)\,d\omega \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} G(-\imath\omega)H(\imath\omega)\,d\omega,\end{aligned}$$

with a norm defined as

$$\|G\|_{\mathcal{H}_2} \stackrel{\text{def}}{=} \left(\frac{1}{2\pi}\int_{-\infty}^{+\infty} |G(\imath\omega)|^2\,d\omega\right)^{1/2}. \tag{2.1.1}$$

The $\mathcal{H}_2$-norm can be shown [5] to be equivalent to

$$\|G\|_{\mathcal{H}_2}^2 = \mathbf{c}^T\mathbf{Q}\mathbf{c}, \tag{2.1.2}$$

where $\mathbf{Q}$ solves the reachability Lyapunov equation $\mathbf{A}\mathbf{Q} + \mathbf{Q}\mathbf{A}^T + \mathbf{b}\mathbf{b}^T = 0$. If G is the original system of order n and G_r is the reduced system of order r with $n \gg r$, then

$$J := \|G - G_r\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi}\int_{-\infty}^{+\infty} |G(\imath\omega) - G_r(\imath\omega)|^2\,d\omega. \tag{2.1.3}$$

2.1.1 Interpolation based $\mathcal{H}_2$ optimality conditions

The goal of $\mathcal{H}_2$ model reduction is to replace a system G by a reduced order system G_r such that $J = \|G(s) - G_r(s)\|_{\mathcal{H}_2}^2$ is minimized. This means, if G is a stable SISO finite dimensional dynamical system as described in (1.2.1), the goal is to find a stable reduced order system

G_r of order r as described in (1.2.4), which is the best stable r^{th} order dynamical system approximating G with respect to the $\mathcal{H}_2$ norm:

$$\|G - G_r\|_{\mathcal{H}_2} = \min_{\substack{\dim(\widetilde{G}_r) = r \\ \widetilde{G}_r : \text{stable}}} \|G - \widetilde{G}_r\|_{\mathcal{H}_2}. \tag{2.1.4}$$

Let the poles of the full-order model G be $\lambda_1, ..., \lambda_n$ and let the poles of the reduced order model G_r be $\hat{\lambda}_1, ..., \hat{\lambda}_r$. In [26], we can find Theorem 2.1.1 which yields the error expression $J = \|G(s) - G_r(s)\|^2_{\mathcal{H}_2}$.

Theorem 2.1.1. *Given the full-order model $G(s)$ and a reduced order model $G_r(s)$, let λ_i and $\hat{\lambda}_j$ be the poles of $G(s)$ and $G_r(s)$, respectively, and suppose that the poles of $G_r(s)$ are distinct. Let Φ_i and $\hat{\Phi}_j$ denote the residues of the transfer functions $G(s)$ and $G_r(s)$ at their poles λ_i and $\hat{\lambda}_j$, respectively: $\Phi_i = \mathsf{res}[G(s), \lambda_i]$ $i = 1, \cdots, n$ and $\hat{\Phi}_j = \mathsf{res}[G_r(s), \hat{\lambda}_j]$ for $j = 1, \cdots, r$. The $\mathcal{H}_2$ norm of the error system is given by*

$$\begin{aligned} \|G - G_r\|^2_{\mathcal{H}_2} &= \sum_{i=1}^{n} \mathsf{res}[(G(-s) - G_r(-s))\,(G(s) - G_r(s))\,, \lambda_i] \\ &\quad + \sum_{j=1}^{r} \mathsf{res}[(G(-s) - G_r(-s))\,(G(s) - G_r(s))\,, \hat{\lambda}_j] \\ &= \sum_{i=1}^{n} \Phi_i \Big(G(-\lambda_i) - G_r(-\lambda_i) \Big) - \sum_{j=1}^{r} \hat{\Phi}_j \Big(G(-\hat{\lambda}_j) - G_r(-\hat{\lambda}_j) \Big). \end{aligned} \tag{2.1.5}$$

As we can see, the error is due to the mismatch of the transfer functions G and G_r at the mirror images of the full order poles $\lambda_1, ..., \lambda_n$ and the reduced order poles $\hat{\lambda}_1, ..., \hat{\lambda}_r$.

Now let the partial fraction expansions of $G(s)$ and $G_r(s)$ be $G(s) = \sum_{k=1}^{n} \frac{\Phi_k}{s-\lambda_k}$ and $G_r(s) = \sum_{k=1}^{r} \frac{\hat{\Phi}_k}{s-\hat{\lambda}_k}$. Next, we re-derive the interpolation-based necessary conditions for $\mathcal{H}_2$-optimality, also known as Meier-Luenberger [34] conditions. The importance of the poles of the reduced order system is shown in this theorem.

Theorem 2.1.2. *Let $G(s) = \sum_{k=1}^{n} \frac{\Phi_k}{s-\lambda_k}$ be a full order system and $G_r(s) = \sum_{k=1}^{r} \frac{\hat{\Phi}_k}{s-\hat{\lambda}_k}$ be a minimizer for $J = \|G(s) - G_r(s)\|_{\mathcal{H}_2}^2$. Then:*

$$G(-\hat{\lambda}_m) = G_r(-\hat{\lambda}_m) \text{ and } G'(-\hat{\lambda}_m) = G_r'(-\hat{\lambda}_m) \text{ for } k = 1, ..., r. \tag{2.1.6}$$

Proof. First consider the following derivatives of G and G_r, which follow from the partial fraction expansions:

1) $\frac{\partial}{\partial \hat{\Phi}_m} G_r(-\lambda_j) = -\frac{1}{\lambda_j + \hat{\lambda}_m}$

2) $\frac{\partial}{\partial \hat{\Phi}_m} G_r(-\hat{\lambda}_j) = -\frac{1}{\hat{\lambda}_j + \hat{\lambda}_m}$

3) $\frac{\partial}{\partial \hat{\lambda}_m} G(-\lambda_j) = 0$

4) $\frac{\partial}{\partial \hat{\lambda}_m} G_r(-\lambda_j) = \frac{\hat{\Phi}_m}{(\lambda_j + \hat{\lambda}_m)^2}$

5) $\frac{\partial}{\partial \hat{\lambda}_m} G(-\hat{\lambda}_j) = \begin{cases} \sum_{k=1}^{n} \frac{\Phi_k}{(\hat{\lambda}_m + \lambda_k)^2}, & j = m \\ 0, & j \neq m \end{cases}$

$$6)\ \frac{\partial}{\partial\hat{\lambda}_m}G_r(-\hat{\lambda}_j) = \begin{cases} \sum_{k=1,k\neq m}^{r}\frac{\hat{\Phi}_k}{(\hat{\lambda}_m+\hat{\lambda}_k)^2}+\frac{\hat{\Phi}_m}{2\hat{\lambda}_m^2}, & j=m \\ \frac{\hat{\Phi}_m}{(\hat{\lambda}_j+\hat{\lambda}_m)^2}, & j\neq m \end{cases}$$

Using the partial fraction expansions of $G(s)$ and $G_r(s)$ and considering the derivatives above we get following derivatives of J:

$$\begin{aligned}
\frac{\partial J}{\partial\hat{\Phi}_m} &= \sum_{k=1}^{n}[\frac{1}{\lambda_k+\hat{\lambda}_m}]\cdot\Phi_k + G_r(-\hat{\lambda}_m) - G(-\hat{\lambda}_m) + \sum_{k=1}^{r}[-\frac{1}{\hat{\lambda}_k+\hat{\lambda}_m}]\cdot\hat{\Phi}_k \\
&= \sum_{k=1}^{n}[\frac{-1}{-\lambda_k-\hat{\lambda}_m}]\cdot\Phi_k + G_r(-\hat{\lambda}_m) - G(-\hat{\lambda}_m) + \sum_{k=1}^{r}[\frac{1}{-\hat{\lambda}_k-\hat{\lambda}_m}]\cdot\hat{\Phi}_k \\
&= -G(-\hat{\lambda}_m) + G_r(-\hat{\lambda}_m) - G(-\hat{\lambda}_m) + G_r(-\hat{\lambda}_m) \\
&= 2(G_r(-\hat{\lambda}_m) - G(-\hat{\lambda}_m)) \qquad (2.1.7)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial J}{\partial\hat{\lambda}_m} &= \sum_{k=1}^{n}[\frac{-\hat{\Phi}_m}{(\lambda_k+\hat{\lambda}_m)^2}]\cdot\Phi_k + \sum_{k=1,k\neq m}^{r}[\frac{\hat{\Phi}_m}{(\hat{\lambda}_k+\hat{\lambda}_m)^2}]\cdot\hat{\Phi}_k \\
&+ (\sum_{k=1,k\neq m}^{r}[\frac{\hat{\Phi}_k}{(\hat{\lambda}_k+\hat{\lambda}_m)^2}]+\frac{\hat{\Phi}_m}{2\hat{\lambda}_m^2})\cdot\hat{\Phi}_m - \sum_{k=1}^{n}[\frac{\Phi_k}{(\lambda_k+\hat{\lambda}_m)^2}]\cdot\hat{\Phi}_m \\
&= -\hat{\Phi}_m(2\sum_{k=1}^{n}[\frac{\Phi_k}{(\lambda_k+\hat{\lambda}_m)^2}] - 2\sum_{k=1,k\neq m}^{r}[\frac{\hat{\Phi}_k}{(\hat{\lambda}_k+\hat{\lambda}_m)^2}] - \frac{\hat{\Phi}_m}{2\hat{\lambda}_m^2})
\end{aligned}$$

$$
\begin{aligned}
&= -\hat{\Phi}_m(2\sum_{k=1}^{n}[\frac{\Phi_k}{(\lambda_k+\hat{\lambda}_m)^2}] - 2\sum_{k=1,k\neq m}^{r}[\frac{\hat{\Phi}_k}{(\hat{\lambda}_k+\hat{\lambda}_m)^2}] - 2\frac{\hat{\Phi}_m}{(\hat{\lambda}_m+\hat{\lambda}_m)^2}) \\
&= -\hat{\Phi}_m(2\sum_{k=1}^{n}[\frac{\Phi_k}{(\lambda_k+\hat{\lambda}_m)^2}] - 2\sum_{k=1}^{r}[\frac{\hat{\Phi}_k}{(\hat{\lambda}_k+\hat{\lambda}_m)^2}]) \\
&= 2\hat{\Phi}_m(G'(-\hat{\lambda}_m) - G_r'(-\hat{\lambda}_m)) \qquad (2.1.8)
\end{aligned}
$$

Setting (2.1.7) and (2.1.8) equal to zero yields (2.1.6) and we have proved that the Meier-Luenberger conditions (2.1.6) are necessary conditions for G_r to be a minimizer for the error expression J.

□

2.2 An Iterative Rational Krylov Algorithm

In the literature we can find many approaches for solving the $\mathcal{H}_2$-minimization problem (2.1.4), e.g. in [11, 27, 29, 33, 34, 47]. A common strategy is to compute a reduced order system which satisfies first-order necessary conditions. However, in large-scale settings, these approaches are very expensive. The problem with these approaches is that either solving several Lyapunov equations in every step is required, or computing the coefficients of the transfer functions is required. We want to avoid working with transfer functions, because computing the coefficients of the transfer functions can be extremely ill-conditioned.

The authors of [26] proposed an approach which allows to compute a reduced-order model

without solving Lyapunov equations in every step, and without computing the transfer functions explicitly. By means of rational Krylov iterations, a reduced order model is constructed in an numerically effective and stable way.

Assume $G_r(s)$ interpolates $G(s)$ and $G'(s)$ at the interpolation points $\boldsymbol{\sigma}$:=$\{\sigma_1, \ldots, \sigma_r\}$. Let $\boldsymbol{\lambda}(\boldsymbol{\sigma}) = \{\hat{\lambda}_1, \ldots, \hat{\lambda}_r\}$ denote the poles of the computed reduced order system G_r. $\boldsymbol{\lambda}(\boldsymbol{\sigma})$ is a function from $\mathbb{C}^r \mapsto \mathbb{C}^r$. If we define $\mathbf{g}(\boldsymbol{\sigma}) := \boldsymbol{\lambda}(\boldsymbol{\sigma}) + \boldsymbol{\sigma}$ then $\mathbf{g}(\boldsymbol{\sigma})$ is also from $\mathbb{C}^r \mapsto \mathbb{C}^r$. Now remember the Meier-Luenberger conditions (2.1.6). The Meier-Luenberger conditions tell us that, for $\mathcal{H}_2$-optimality, necessary conditions are that the interpolation points are the mirror images of the reduced order poles. Setting $\mathbf{g}(\boldsymbol{\sigma}) = \mathbf{0}$ yields $\boldsymbol{\lambda}(\boldsymbol{\sigma}) = -\boldsymbol{\sigma}$, and thus is equivalent to the Meier-Luenberger conditions. This discussion yields that constructing an $\mathcal{H}_2$-optimal reduced order system can be done by solving the root finding problem $\mathbf{g}(\boldsymbol{\sigma}) = \mathbf{0}$. The authors of [26] solve an equivalent problem to the root finding problem above by using a variant of the Newton method

$$\boldsymbol{\sigma}^{(k+1)} = \boldsymbol{\sigma}^{(k)} - (\mathbf{I} + \mathbf{J})^{-\mathbf{1}} \left(\boldsymbol{\sigma}^{(\mathbf{k})} + \boldsymbol{\lambda} \left(\boldsymbol{\sigma}^{(\mathbf{k})} \right) \right), \tag{2.2.1}$$

where $\mathbf{J}$ denotes the usual $r \times r$ Jacobian of $\boldsymbol{\lambda}(\boldsymbol{\sigma})$ with respect to $\boldsymbol{\sigma}$. Due to the fact that the entries of the Jacobian matrix become small in the neighborhood of an $\mathcal{H}_2$-optimal shift set, [26] chose setting $\mathbf{J} = 0$ as relaxed iteration strategy. This is the motivation behind the strategy of assigning $\sigma_i \leftarrow -\lambda_i(\mathbf{A}_r)$ in every step, i.e. choosing the new interpolation points as the mirror images of the poles of the $\mathbf{A}_r$ from the previous step. Next we present the algorithm which was suggested in [26]. Moreover, we will illustrate the performance of this algorithm.

Algorithm 2.2.1. An Iterative Rational Krylov Algorithm (IRKA):

1. *Make an initial selection of σ_i, for $i = 1, \ldots, r$ that is closed under conjugation and fix a convergence tolerance tol.*

2. *Choose $\mathbf{V}_r$ and $\mathbf{W}_r$ so that Ran$(\mathbf{V}_r) =$ span$\left\{(\sigma_1\mathbf{I} - \mathbf{A})^{-1}\mathbf{b},\ \cdots,\ (\sigma_r\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}\right\}$, Ran$(\mathbf{W}_r) =$ span$\left\{(\sigma_1\mathbf{I} - \mathbf{A}^T)^{-1}\mathbf{c}, \cdots, (\sigma_r\mathbf{I} - \mathbf{A}^T)^{-1}\mathbf{c}\right\}$, and $\mathbf{W}_r^T\mathbf{V}_r = \mathbf{I}$.*

3. *while (relative change in $\{\sigma_i\} >$ tol)*

   (a) $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r$,

   (b) *Assign $\sigma_i \longleftarrow -\lambda_i(\mathbf{A}_r)$ for $i = 1, \ldots, r$*

   (c) *Update $\mathbf{V}_r$ and $\mathbf{W}_r$ so Ran$(\mathbf{V}_r) =$ span$\left\{(\sigma_1\mathbf{I} - \mathbf{A})^{-1}\mathbf{b},\ \cdots,\ (\sigma_r\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}\right\}$ Ran$(\mathbf{W}_r) =$ span$\left\{(\sigma_1\mathbf{I} - \mathbf{A}^T)^{-1}\mathbf{c}, \cdots, (\sigma_r\mathbf{I} - \mathbf{A}^T)^{-1}\mathbf{c}\right\}$ and $\mathbf{W}_r^T\mathbf{V}_r = \mathbf{I}$.*

4. $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r$, $\mathbf{b}_r = \mathbf{W}_r^T\mathbf{b}$, $\mathbf{c}_r^T = \mathbf{c}^T\mathbf{V}_r$

The algorithm is designed in a way such that the reduced order system G_r satisfies the Meier-Luenberger conditions. In other words, the algorithm forces the necessary conditions for $\mathcal{H}_2$-optimality to be satisfied. Now we illustrate the performance of the algorithm. We use the algorithm for reducing a system of order 120 to order 14, 16 and 18. In Figures 2.1, 2.2 and 2.3 the green graph represents the original system G and the red graph represents the reduced system G_r.

The following figures illustrate the satisfying performance of IRKA. Despite of its simplicity, IRKA yields satisfying errors and converges quickly.

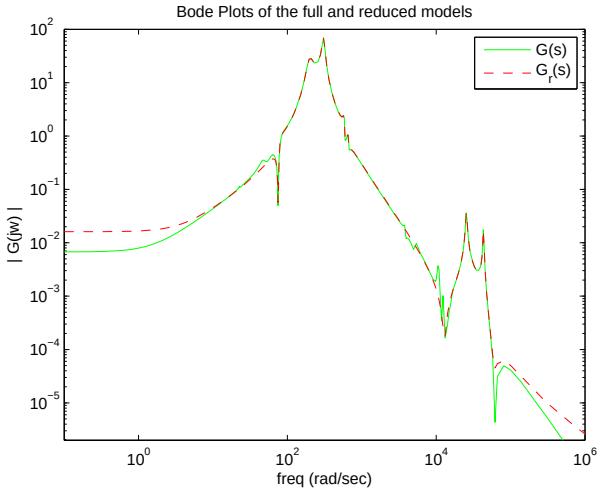

Figure 2.1: Reduction of order 120 to order 14

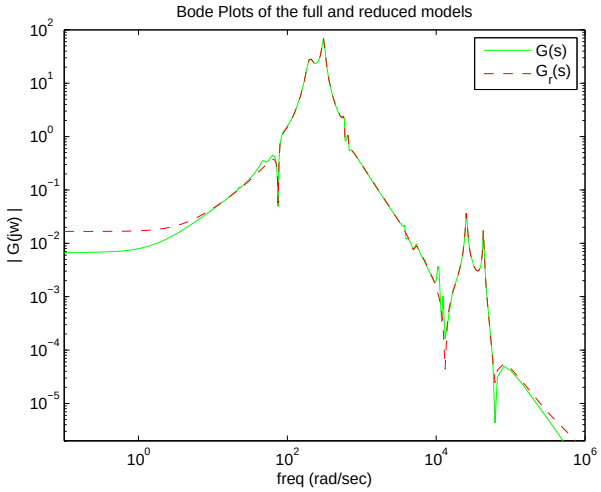

Figure 2.2: Reduction of order 120 to order 16

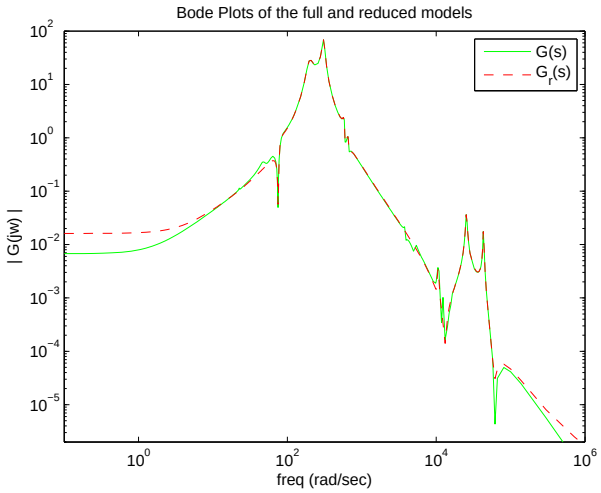

Figure 2.3: Reduction of order 120 to order 18

Chapter 3

Weighted-$\mathcal{H}_2$ model reduction

As already mentioned earlier, in some applications it is desirable to use a weight while reducing the order of the original system. In this chapter we will first extend the $\mathcal{H}_2$ norm to the weighted case by introducing the weighted-$\mathcal{H}_2$ inner product, and thus introducing the weighted-$\mathcal{H}_2$ model reduction problem. Moreover, we will discuss the controller reduction problem since this leads to a weighted model reduction problem. Finally, we review Halevi's approach proposed in [27] and introduce new optimality conditions.

3.1 The weighted-$\mathcal{H}_2$ norm

Assume $G(s), H(s)$ and $W(s)$ are transfer functions associated with real stable SISO dynamical systems. Then we can define the weighted-$\mathcal{H}_2$ inner product as

$$\begin{aligned}\langle G \cdot W,\ H \cdot W \rangle_{\mathcal{H}_2} &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \overline{G(\imath\omega)W(\imath\omega)} W(\imath\omega) H(\imath\omega)\, d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} G(-\imath\omega) W(-\imath\omega) W(\imath\omega) H(\imath\omega)\, d\omega,\end{aligned}$$

with a norm defined as

$$\|G \cdot W\|_{\mathcal{H}_2} \stackrel{\text{def}}{=} \left(\frac{1}{2\pi} \int_{-\infty}^{+\infty} |W(\imath\omega) G(\imath\omega)|^2 \, d\omega \right)^{1/2}. \tag{3.1.1}$$

This means, if G is the original system of order n and G_r is the reduced system of order r, with $n \gg r$, then

$$J_w := \|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |W(\imath\omega) \cdot (G(\imath\omega) - G_r(\imath\omega))|^2 \, d\omega. \tag{3.1.2}$$

Since the goal is to reduce $\|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2$, in the next chapter we will derive a new error expression for $J_w = \|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2$ and we will discuss our approach for minimizing this error expression, i.e. we want to solve the optimal weighted-$\mathcal{H}_2$ model reduction problem:

$$\|(G - G_r) \cdot W\|_{\mathcal{H}_2} = \min_{\substack{\dim(\widetilde{G}_r) = r \\ \widetilde{G}_r \text{: stable}}} \|(G - \widetilde{G}_r) \cdot W\|_{\mathcal{H}_2}. \tag{3.1.3}$$

The rest of this chapter will be used to introduce the controller reduction problem, to present

Halevi's approach to the frequency weighted model reduction problem which was presented in [27] and to present new optimality conditions.

3.2 The Controller Reduction Problem

The control of a system G with a controller K is being considered in this section. Assume K controls G by a feedback loop. Let $G(s)$ be an n^{th} order system with the transfer function $G(s) = C(sI - A)^{-1}B + D$, and $K(s)$ an n_κ^{th} order controller with the transfer function $K(s) = C_K(sI - A_K)^{-1}B_K + D_K$. The problem is that usually the order of the controller K is similar to the order of the the controlled system G, see [46, 51] and the references therein. Thus, in large scale settings, the order of the controller is very high. In real-time applications the accuracy becomes an issue due to ill-conditioning. Moreover the time needed to compute the output response can be a problem, and thus problematic feedback inputs can be produced. These are the main reasons for the need of reducing the controller K of order n_κ to order r. However, in many cases just requiring that the reduced controller K_r is a good approximation to K is not enough to preserve closed-loop performance. Moreover, the plant dynamics have to be considered. The solution to this problem is model reduction by frequency weighting [1, 43, 46]. The strategy of reducing a stabilizing controller $K(s)$ is to find a reduced-order controller $K_r(s)$ with the same number of unstable poles. In [1, 21, 51] we can see that, in order to ensure closed loop stability, one has to minimize

$$\|(K(s) - K_r(s)) \cdot G(s)(I + G(s)K(s))^{-1}\|_{\mathcal{H}_\infty}. \tag{3.2.1}$$

Now define $W(s) := G(s)(I + G(s)K(s))^{-1}$, and we obtain that the controller reduction

problem is an application of the weighted model reduction problem of minimizing

$$\|(K(s) - K_r(s)) \cdot W(s)\|_{\mathcal{H}_\infty}. \tag{3.2.2}$$

3.3 Halevi's approach

In this section we discuss Halevi's approach presented in [27]. The problem of order reduction with frequency weighting for the multiple input/ multiple output (MIMO) case is being considered. In [27] the optimal reduced-order model, i.e. the model G_r that minimizes $J_w := \|(G - G_r) \cdot W\|^2_{\mathcal{H}_2}$, is characterized as given below in Theorem 3.3.2. First we consider some preparations which are needed for Theorem 3.3.2.

Let $W(s)$ have the state space realization

$$W : \begin{cases} \dot{\mathbf{x}}_w(t) &= \mathbf{A}_w\mathbf{x}_w(t) + \mathbf{b}_w\eta(t) \\ y_w(t) &= \mathbf{c}_w^T\mathbf{x}_w(t) + \mathbf{d}_w\eta(t) \end{cases} \quad \text{or} \quad W(s) = \mathbf{c}_w^T(s\mathbf{I} - \mathbf{A}_w)^{-1}\mathbf{b}_w + \mathbf{d}_w. \tag{3.3.1}$$

where $\mathbf{x}_w \in \mathbb{R}^{n_w}$, $\eta \in \mathbb{R}^{m_w}$ and $y_w \in \mathbb{R}^m$. In [27], $W(s)$ is regarded as a shaping filter. This means $W(s)$ acts on the input η in order to create the input for the original system (1.2.1) and the reduced system (1.2.4). Replacing u, in (1.2.1) and (1.2.4), by y_w yields

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}(\mathbf{c}_w^T\mathbf{x}_w(t) + \mathbf{d}_w\eta(t)) \tag{3.3.2}$$

$$\dot{\mathbf{x}}_r(t) = \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{b}_r(\mathbf{c}_w^T\mathbf{x}_w(t) + \mathbf{d}_w\eta(t)). \tag{3.3.3}$$

Then, combining (3.3.1), (3.3.2) and (3.3.3) leads to

$$\dot{\tilde{\mathbf{x}}} := \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_w(t) \\ \dot{\mathbf{x}}_r(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b}\mathbf{c}_w^T & 0 \\ 0 & \mathbf{A}_w & 0 \\ 0 & \mathbf{b}_r\mathbf{c}_w^T & \mathbf{A}_r \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_w(t) \\ \mathbf{x}_r(t) \end{bmatrix} + \begin{bmatrix} \mathbf{b}\mathbf{d}_w \\ \mathbf{b}_w \\ \mathbf{b}_r\mathbf{d}_w \end{bmatrix} \eta(t). \tag{3.3.4}$$

Since $y(t) = \mathbf{c}^T\mathbf{x}(t)$ and $y_r(t) = \mathbf{c}_r^T\mathbf{x}_r(t)$ we obtain

$$\varepsilon := y(t) - y_r(t) = \mathbf{c}\mathbf{x} - \mathbf{c}_r\mathbf{x}_r = \begin{bmatrix} \mathbf{c} & 0 & -\mathbf{c}_r \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_w(t) \\ \dot{\mathbf{x}}_r(t) \end{bmatrix}. \tag{3.3.5}$$

Assume $\mathbf{d}_w \neq 0$. Suppose $rank(\mathbf{d}_w) \leq m$ and define $m_1 := rank(\mathbf{d}_w)$. Moreover, we assume that $\mathbf{d}_w$ has full row-rank since this can always be reached by an output transformation. Partition $\mathbf{b}_r$ and $\mathbf{c}_w$ as $\mathbf{b}_r$=[$\mathbf{b}_{r1}$ $\mathbf{b}_{r2}$] and $\mathbf{c}_w$=[$\mathbf{c}_{w1}^T$ $\mathbf{c}_{w2}^T$]. Using these partitions we obtain

$$\dot{\tilde{\mathbf{x}}} := \underbrace{\begin{bmatrix} \mathbf{A}_1 & 0 \\ \mathbf{b}_{r1}\mathbf{c}_{21} + \mathbf{b}_{r2}\mathbf{c}_{22} & \mathbf{A}_r \end{bmatrix}}_{=\tilde{\mathbf{A}}} \tilde{\mathbf{x}} + \underbrace{\begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_{r1}\mathbf{d}_w \end{bmatrix}}_{=\tilde{\mathbf{b}}} \eta \text{ with } \tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_w(t) \\ \mathbf{x}_r(t) \end{bmatrix} \tag{3.3.6}$$

and

$$\mathbf{A}_1 = \begin{bmatrix} \mathbf{A} & \mathbf{b}\mathbf{c}_w \\ 0 & \mathbf{A}_w \end{bmatrix}; \mathbf{b}_1 = \begin{bmatrix} \mathbf{b}\mathbf{d}_w \\ \mathbf{b}_w \end{bmatrix}; \mathbf{c}_{21} = \begin{bmatrix} 0 \\ \mathbf{c}_{w1} \end{bmatrix}; \mathbf{c}_{22} = \begin{bmatrix} 0 \\ \mathbf{c}_{w2} \end{bmatrix}. \tag{3.3.7}$$

By defining $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ appropriately we can represent (3.3.6) as $\dot{\tilde{\mathbf{x}}} = \tilde{\mathbf{A}}\tilde{\mathbf{x}} + \tilde{\mathbf{b}}\eta$. (3.3.5) and (3.3.6) define the state-space representation of the error system $(G - G_r)W$.

Following assumptions are made:

A) $\mathbf{A}_r$ is asymptotically stable

B) $(\mathbf{A}_r, \mathbf{b}_r)$ is controllable and $(\mathbf{A}_r, \mathbf{c}_r)$ is observable.

By these assumptions, we obtain that if $\tilde{\mathbf{Q}}$ solves the Lyapunov equation $\tilde{\mathbf{A}}\tilde{\mathbf{Q}}+\tilde{\mathbf{Q}}\tilde{\mathbf{A}}^T+\tilde{\mathbf{b}}\tilde{\mathbf{b}}^T=0$ we need to reduce $E := tr(\tilde{\mathbf{c}}^T\tilde{\mathbf{Q}}\tilde{\mathbf{c}})$ since $E = J_w$, see (2.1.2). Now partition $\tilde{\mathbf{Q}}$ as

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_{12} \\ \mathbf{Q}_{12}^T & \mathbf{Q}_2 \end{bmatrix} \text{ with } \mathbf{Q}_1 \in \mathbb{R}^{(n+n_w)\times(n+n_w)} \text{ and } \mathbf{Q}_2 \in \mathbb{R}^{n_r \times n_r}. \tag{3.3.8}$$

After defining $\hat{\mathbf{Q}} := \mathbf{Q}_{12}\mathbf{Q}_2^{-1}\mathbf{Q}_{12}^T$ and $\mathbf{Q} := \mathbf{Q}_1 - \hat{\mathbf{Q}}$, we make two more assumptions:

C) $\mathbf{Q}_2 > 0$

D) if $\mathbf{d}_w$ does not have full-row rank, then $\mathbf{c}_{22}\mathbf{Q}\mathbf{c}_{22}^T > 0$.

The next lemma can be found in [29]. It is the last preparation for the main result of [27], presented in Theorem 3.3.2. Proofs of Lemma 3.3.1 and Theorem 3.3.2 can be found in [29] and [27], respectively.

Lemma 3.3.1. *Suppose $\hat{\mathbf{Q}}$ and $\hat{\mathbf{P}}$ are $n \times n$ nonnegative definite and $rank(\hat{\mathbf{Q}}\hat{\mathbf{P}}) = n_r$. Then there exist $n_r \times n$ G, Γ and an invertible $n_r \times n$ M such that $\hat{\mathbf{Q}}\hat{\mathbf{P}} = G^T M \Gamma$ and $\Gamma G^T = I_n$. $(\hat{\mathbf{Q}}\hat{\mathbf{P}})^\sharp := G^T M^{-1} \Gamma$ is the group inverse of $\hat{\mathbf{Q}}\hat{\mathbf{P}}$ and $\tau := (\hat{\mathbf{Q}}\hat{\mathbf{P}})(\hat{\mathbf{Q}}\hat{\mathbf{P}})^\sharp = G^T\Gamma$ is an oblique projection.*

Theorem 3.3.2. *Suppose $\mathbf{A}_r, \mathbf{b}_r$ and $\mathbf{c}_r$ satisfy A,B,C,D and minimize E. Then they are given by*

$$\mathbf{A}_r = \Gamma(A_1 - \mathbf{Q}_a \Sigma_w \mathbf{c}_{21}) \nu_\perp G^T \tag{3.3.9}$$

$$\mathbf{b}_r = \Gamma[\mathbf{Q}_a \Sigma_w (\mathbf{A}_1 - \mathbf{Q}_a \Sigma_w \mathbf{c}_{21}) \mathbf{c}_{22}^*] \tag{3.3.10}$$

$$\mathbf{c}_r = \mathbf{c}_1 \nu_\perp G^T \tag{3.3.11}$$

where

$$\Sigma_w = (\mathbf{d}_w \mathbf{d}_w^T)^{-1} \; ; \; \mathbf{Q}_a = \mathbf{Q}_1 \mathbf{c}_{21}^T + \mathbf{b}_1 \mathbf{d}_w^T$$

$$\tau = (\hat{\mathbf{Q}}\hat{\mathbf{P}})(\hat{\mathbf{Q}}\hat{\mathbf{P}})^\sharp \; ; \; \tau_\perp = I_{n+n_w} - \tau$$

$$\mathbf{c}_{22}^* = \mathbf{Q}\mathbf{c}_{22}^T(\mathbf{c}_{22}\mathbf{Q}\mathbf{c}_{22}^T)^{-1}$$

$$\nu = \mathbf{c}_{22}^*\mathbf{c}_{22} = \mathbf{Q}\mathbf{c}_{22}^T(\mathbf{c}_{22}\mathbf{Q}\mathbf{c}_{22}^T)^{-1}\mathbf{c}_{22} \; ; \; \nu_\perp = I_{n+n_w} - \nu$$

and $\mathbf{Q}, \hat{\mathbf{Q}}$ *and* $\hat{\mathbf{P}}$ *are nonnegative definite matrices satisfying*

$$\begin{aligned}[\mathbf{A}_1 - \tau(\mathbf{A}_1 - \mathbf{Q}_a\Sigma_w\mathbf{c}_{21})\nu]\mathbf{Q} \\ +\mathbf{Q}[\mathbf{A}_1 - \tau(\mathbf{A}_1 - \mathbf{Q}_a\Sigma_w\mathbf{c}_{21})\nu]^T \\ -\mathbf{Q}_a\Sigma_w\mathbf{Q}_a^T + \tau_\perp\mathbf{Q}_a\Sigma\mathbf{Q}_a^T\tau_\perp^T + \mathbf{b}_1\mathbf{b}_1^T = 0\end{aligned} \tag{3.3.12}$$

$$\begin{aligned}\mathbf{A}_1\hat{\mathbf{Q}} + \hat{\mathbf{Q}}\mathbf{A}_1^T + \tau(\mathbf{A}_1 - \mathbf{Q}_a\Sigma_w\mathbf{c}_{21})\nu\mathbf{Q} \\ +\mathbf{Q}\nu^T(\mathbf{A}_1 - \mathbf{Q}a\Sigma_w\mathbf{c}_{21})^T\tau^T + \mathbf{Q}_a\Sigma_w\mathbf{Q}_a^T \\ -\tau_\perp\mathbf{Q}_a\Sigma_w\mathbf{Q}_a^T\tau_\perp^T = 0\end{aligned} \tag{3.3.13}$$

$$\begin{aligned}\hat{\mathbf{P}}(\mathbf{A}_1 - \mathbf{Q}_a\Sigma_w\mathbf{c}_{21})\nu_\perp + \nu_\perp^T(\mathbf{A}_1 - \mathbf{Q}_a\Sigma_w\mathbf{c}_{21})^T\hat{\mathbf{P}} \\ +\nu_\perp^T\mathbf{c}_1^T\mathbf{c}_1\nu_\perp - \tau_\perp^T\nu_\perp^T\mathbf{c}_1^T\mathbf{c}_1\nu_\perp\tau_\perp = 0\end{aligned} \tag{3.3.14}$$

with

$$rank(\hat{\mathbf{Q}}) = rank(\hat{\mathbf{P}}) = rank(\hat{\mathbf{Q}}\hat{\mathbf{P}}) = n_r$$

Based on Theorem 3.3.2, Halevi [27] proposed the following algorithm:

Algorithm 3.3.1. Halevi's algorithm:

1. *Initialize* $\mathbf{Q}^{(0)}, \nu^{(0)}$ *and* $\tau^{(0)}$
2. *Solve (3.3.13) for* $\mathbf{Q}^{(i)}$ *as a regular Riccati equation where the terms* $\tau(\mathbf{A}_1 - \mathbf{Q}_a\Sigma_w\mathbf{c}_{21})\nu$ *and* $\tau_\perp\mathbf{Q}_a\Sigma_w\mathbf{Q}_a^T\tau_\perp^T$ *are evaluated from the* $(i-1)st$ *iteration*
3. *Solve (3.3.13) and (3.3.14) for* $\hat{\mathbf{Q}}^{(i)}$ *and* $\hat{\mathbf{P}}^{(i)}$*, respectively*
4. *Update* $\nu^{(i)}$ *and* $\tau^{(i)}$
5. *If* $\mathbf{Q}^{(i)}$,, $\hat{\mathbf{Q}}^{(i)}$ *and* $\hat{\mathbf{P}}^{(i)}$ *did not converge, go to 2.*

As we can see, the algorithm requires solving a couple of Riccati equations of large dimension at each step. Therefore, this algorithm is infeasible for large-scale dynamical systems. We will propose an effective algorithm for the solution of the weighted model reduction problem. Our algorithm is an iterative Krylov subspace algorithm, i.e. no solution of large Riccati equations is required.

3.4 Error expression for the weighted-$\mathcal{H}_2$ norm

Since $\|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2$ is the quantity we would like to keep as small as possible, our goal in this section is to derive a formula for $\|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2$. Knowing the components of this error will tell us which parts have to be manipulated in order to reduce the error. This error expression will be used in Chapter 4, to propose a Krylov-based numerical algorithm for solving the weighted-$\mathcal{H}_2$ problem. The following lemma makes the weighted-$\mathcal{H}_2$ inner product computationally more accessible.

Lemma 3.4.1. *Suppose the following:*

i) G(s) has poles at $\lambda_1, \ldots, \lambda_n$

ii) H(s) has poles at $\mu_1, \ldots, \mu_m$

iii) W(s) has poles at $\gamma_1, \ldots, \gamma_p$

where the poles of $H(s)$ *and* $W(s)$ *are assumed to be distinct. Moreover assume that the poles of* $G(s), H(s)$ *and* $W(s)$ *are contained in the open left halfplane. Then:*

$$\begin{aligned}\langle G \cdot W,\ H \cdot W \rangle_{\mathcal{H}_2} = &\sum_{k=1}^{m} res[G(-s)W(-s)W(s)H(s), \mu_k] \\ &+ \sum_{k=1}^{p} res[G(-s)W(-s)W(s)H(s), \gamma_k]\end{aligned}$$

In particular:

If μ_k *is a simple pole of* $H(s)$ *then*

$$res[G(-s)W(-s)W(s)H(s), \mu_k] = G(-\mu_k)W(-\mu_k)W(\mu_k) \cdot res[H(s), \mu_k].$$

If γ_k *is a simple pole of* $W(s)$ *then*

$$res[G(-s)W(-s)W(s)H(s), \gamma_k] = G(-\gamma_k)W(-\gamma_k)H(\gamma_k) \cdot res[W(s), \gamma_k].$$

If μ_k is a double pole of $H(s)$ then

$$res[G(-s)W(-s)W(s)H(s),\mu_k]=G(-\mu_k)W(-\mu_k)W(\mu_k)\cdot res[H(s),\mu_k] \\ -(G(-s)W(-s)W(s))'(\mu_k)\cdot h_1(\mu_k)$$

where

$$h_1(\mu_k)=\lim_{s\to\mu_k}((s-\mu_k)^2H(s)).$$

If γ_k is a double pole of $W(s)$ then

$$res[G(-s)W(-s)W(s)H(s),\gamma_k]=G(-\gamma_k)W(-\gamma_k)H(\gamma_k)\cdot res[W(s),\gamma_k] \\ -(G(-s)W(-s)H(s))'(\gamma_k)\cdot h_2(\gamma_k)$$

where

$$h_2(\gamma_k)=\lim_{s\to\gamma_k}((s-\gamma_k)^2W(s)).$$

Proof. The function $G(-s)W(-s)W(s)H(s)$ has singularities at $-\lambda_1,\dots-\lambda_n,-\gamma_1,\dots,-\gamma_p$, $\gamma_1,\dots,\gamma_p,\mu_1,\dots,\mu_m$. For any $R>0$, define the semicircular contour in the left halfplane:

$$\Gamma_R=\{z\,|z=\imath\omega \text{ with } \omega\in[-R,R]\}\cup\left\{z\,\middle|\,z=Re^{\imath\theta} \text{ with } \theta\in[\frac{\pi}{2},\frac{3\pi}{2}]\right\}.$$

For R sufficiently large, Γ_R bounds a region that contains all the system poles of $W(s)H(s)$. The Residue Theorem, which can be found in [13], yields

$$\begin{aligned}\langle G \cdot W,\ H \cdot W \rangle_{\mathcal{H}_2} &= \frac{1}{2\pi} \int_{-\infty}^{+\infty} G(-\imath\omega)W(-\imath\omega)W(\imath\omega)H(\imath\omega)\, d\omega \\ &= \lim_{R\to\infty} \frac{1}{2\pi\imath} \int_{\Gamma_R} G(-s)W(-s)W(s)H(s)\ ds \\ &= \sum_{k=1}^{m} \mathsf{res}[G(-s)W(-s)W(s)H(s), \mu_k] \\ &+ \sum_{k=1}^{p} \mathsf{res}[G(-s)W(-s)W(s)H(s), \gamma_k].\end{aligned}$$

If μ_k and γ_k are simple poles for $H(s)$ and $W(s)$, respectively, then they are also simple poles for $G(-s)W(-s)W(s)H(s)$; hence

$$\begin{aligned}\mathsf{res}[G(-s)W(-s)W(s)H(s),\ \mu_k] &= \lim_{s\to\mu_k} [(s-\mu_k)G(-s)W(-s)W(s)H(s)] \\ &= G(-\mu_k)W(-\mu_k)W(\mu_k) \lim_{s\to\mu_k} ((s-\mu_k)H(s)).\end{aligned}$$

and

$$\begin{aligned}\mathsf{res}[G(-s)W(-s)W(s)H(s),\ \gamma_k] &= \lim_{s\to\gamma_k} [(s-\gamma_k)G(-s)W(-s)W(s)H(s)] \\ &= G(-\gamma_k)W(-\gamma_k)H(\gamma_k) \lim_{s\to\gamma_k} ((s-\gamma_k)W(s)).\end{aligned}$$

If μ_k is a double pole for $H(s)$, then it is also a double pole for $G(-s)W(-s)W(s)H(s)$ and

$$\begin{aligned}
\mathsf{res}[G(-s)W(-s)W(s)H(s),\,\mu_k] &= \lim_{s\to\mu_k}\frac{\partial}{\partial s}[(s-\mu_k)^2G(-s)W(-s)W(s)H(s)] \\
&= \lim_{s\to\mu_k}[G(-s)W(-s)W(s)\frac{\partial}{\partial s}((s-\mu_k)^2H(s)) \\
&+ (G(-s)W(-s)W(s))'(s-\mu_k)^2H(s)] \\
&= G(-\mu_k)W(-\mu_k)W(\mu_k)\lim_{s\to\mu_k}\frac{\partial}{\partial s}((s-\mu_k)^2H(s)) \\
&+ [-G'(-\mu_k)W(-\mu_k)W(\mu_k) - G(-\mu_k)W'(-\mu_k)W(\mu_k) \\
&+ G(-\mu_k)W(-\mu_k)W'(\mu_k)]\lim_{s\to\mu_k}((s-\mu_k)^2H(s)).
\end{aligned}$$

If γ_k is a double pole for $W(s)$, then it is also a double pole for $G(-s)W(-s)W(s)H(s)$ and

$$\begin{aligned}
\mathsf{res}[G(-s)W(-s)W(s)H(s),\,\gamma_k] &= \lim_{s\to\gamma_k}\frac{\partial}{\partial s}[(s-\gamma_k)^2G(-s)W(-s)W(s)H(s)] \\
&= \lim_{s\to\gamma_k}[G(-s)W(-s)H(s)\frac{\partial}{\partial s}((s-\gamma_k)^2W(s)) \\
&+ (G(-s)W(-s)H(s))'(s-\gamma_k)^2W(s)] \\
&= G(-\gamma_k)W(-\gamma_k)H(\gamma_k)\lim_{s\to\gamma_k}\frac{\partial}{\partial s}((s-\gamma_k)^2W(s)) \\
&+ [-G'(-\gamma_k)W(-\gamma_k)H(\gamma_k) - G(-\gamma_k)W'(-\gamma_k)H(\gamma_k) \\
&+ G(-\gamma_k)W(-\gamma_k)H'(\gamma_k)]\lim_{s\to\gamma_k}((s-\gamma_k)^2W(s)).
\end{aligned}$$

□

Lemma 3.4.1 yields the following corollary:

Corollary 3.4.2. *If $G(s)$ has simple poles at $\lambda_1, \lambda_2, \ldots, \lambda_n$ and $W(s)$ has simple poles at $\gamma_1, \gamma_2, \ldots, \gamma_p$, then*

$$\begin{aligned} \|G \cdot W\|_{\mathcal{H}_2}^2 &= \sum_{k=1}^{n} G(-\lambda_k)W(-\lambda_k)W(\lambda_k) \cdot \mathit{res}[G(s), \lambda_k] \\ &+ \sum_{k=1}^{p} G(-\gamma_k)W(-\gamma_k)G(\gamma_k) \cdot \mathit{res}[W(s), \gamma_k] \end{aligned} \tag{3.4.1}$$

With Corollary 3.4.2, we obtain the following theorem:

Theorem 3.4.3. *If the original system $G(s)$ has simple poles at $\lambda_1, \lambda_2, \ldots, \lambda_n$, the reduced order system $G_r(s)$ has simple poles at $\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_r$ and the weight $W(s)$ has simple poles at $\gamma_1, \gamma_2, \ldots, \gamma_p$, then the weighted-$\mathcal{H}_2$ error $J_w = \|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2$ is*

$$\begin{aligned} J_w &= \sum_{k=1}^{n} (G(-\lambda_k) - G_r(-\lambda_k))W(-\lambda_k)W(\lambda_k) \cdot \Phi_k \\ &+ \sum_{k=1}^{r} (G_r(-\hat{\lambda}_k) - G(-\hat{\lambda}_k))W(-\hat{\lambda}_k)W(\hat{\lambda}_k) \cdot \hat{\Phi}_k \\ &+ \sum_{k=1}^{p} (G(-\gamma_k) - G_r(-\gamma_k))W(-\gamma_k)(G(\gamma_k) - G_r(\gamma_k)) \cdot \Psi_k \end{aligned} \tag{3.4.2}$$

where Φ_k*:=* $\mathsf{res}[G(s), \lambda_k]$*,* $\hat{\Phi}_k$*:=* $\mathsf{res}[G_r(s), \hat{\lambda}_k]$ *and* Ψ_k*:=* $\mathsf{res}[W(s), \gamma_k]$*.*

Proof. First, notice that the function $G-G_r$ has simple poles at $\lambda_1, \lambda_2, \ldots, \lambda_n, \hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_r$. Then Corollary 3.4.2 yields

$$\begin{aligned} J_w &= \|(G-G_r)\cdot W\|_{\mathcal{H}_2}^2 \\ &= \sum_{k=1}^{n}(G(-\lambda_k)-G_r(-\lambda_k))W(-\lambda_k)W(\lambda_k)\cdot \mathsf{res}[G(s)-G_r(s),\lambda_k] \\ &+ \sum_{k=1}^{r}(G(-\hat{\lambda}_k)-G_r(-\hat{\lambda}_k))W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot \mathsf{res}[G(s)-G_r(s),\hat{\lambda}_k] \\ &+ \sum_{k=1}^{p}(G(-\gamma_k)-G_r(-\gamma_k))W(-\gamma_k)(G(\gamma_k)-G_r(\gamma_k))\cdot \mathsf{res}[W(s),\gamma_k]. \end{aligned}$$

With the partial fraction expansions $G(s)=\sum_{k=1}^{n}\frac{\Phi_k}{s-\lambda_k}$ and $G_r(s)=\sum_{k=1}^{r}\frac{\hat{\Phi}_k}{s-\hat{\lambda}_k}$, we obtain, for $G-G_r$, the partial fraction expansion $G-G_r{=}\sum_{k=1}^{n=r}\frac{\tilde{\Phi}_k}{s-\tilde{\lambda}_k}$,

$$\text{where } \tilde{\Phi}_k=\begin{cases}\Phi_k, & k=1,...,n\\ -\hat{\Phi}_{k-n}, & k=n+1,...,n+r\end{cases} \text{ and } \tilde{\lambda}_k=\begin{cases}\lambda_k, & k=1,...,n\\ \hat{\lambda}_{k-n}, & k=n+1,...,n+r\end{cases}$$

This yields

$$\mathsf{res}[G(s)-G_r(s),\lambda_k]=\Phi_k=\mathsf{res}[G(s),\lambda_k] \text{ and } \mathsf{res}[G(s)-G_r(s),\hat{\lambda}_k]=-\hat{\Phi}_k=-\mathsf{res}[G(s),\lambda_k].$$

□

Recall that we obtained the Meier-Luenberger conditions (2.1.6) by setting $\frac{\partial J}{\partial \hat{\Phi}_m}$=0 and $\frac{\partial J}{\partial \hat{\lambda}_m}$=0. Since these conditions are necessary conditions for G_r being a minimizer for the error expression $J = \|(G - G_r)\|^2_{\mathcal{H}_2}$, we would like to know the equivalent conditions for the weighted case. However, we will see that they are more involved than (2.1.6).

Theorem 3.4.4. *If the original system $G(s)$ has simple poles at $\lambda_1, \lambda_2, \ldots, \lambda_n$, the reduced order system $G_r(s)$ has simple poles at $\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_r$ and the weight $W(s)$ has simple poles at $\gamma_1, \gamma_2, \ldots, \gamma_p$, then the derivatives of J_w with respect to $\hat{\Phi}_m$ and $\hat{\lambda}_m$ are given by:*

$$\begin{aligned}
\frac{\partial J_w}{\partial \hat{\Phi}_m} &= \sum_{k=1}^{n} [\frac{1}{\lambda_k + \hat{\lambda}_m}] W(-\lambda_k) W(\lambda_k) \cdot \Phi_k \\
&+ \sum_{k=1}^{r} [-\frac{1}{\hat{\lambda}_k + \hat{\lambda}_m}] W(-\hat{\lambda}_k) W(\hat{\lambda}_k) \cdot \hat{\Phi}_k \\
&+ [G_r(-\hat{\lambda}_m) - G(-\hat{\lambda}_m))] W(-\hat{\lambda}_m) W(\hat{\lambda}_m) \\
&+ \sum_{k=1}^{p} [\frac{1}{\gamma_k + \hat{\lambda}_m}] [G(\gamma_k) - G_r(\gamma_k)] W(-\gamma_k) \cdot \Psi_k \\
&+ \sum_{k=1}^{p} [G(-\gamma_k) - G_r(-\gamma_k)] [\frac{-1}{\gamma_k - \hat{\lambda}_m}] W(-\gamma_k) \cdot \Psi_k
\end{aligned} \tag{3.4.3}$$

$$
\begin{aligned}
\frac{\partial J_w}{\partial \hat{\lambda}_m} &= \sum_{k=1}^{n}[-\frac{\hat{\Phi}_m}{(\lambda_k+\hat{\lambda}_m)^2}]W(-\lambda_k)W(\lambda_k)\cdot\Phi_k \\
&+ \sum_{k=1,k\neq m}^{r}[\frac{\hat{\Phi}_m}{(\lambda_k+\hat{\lambda}_m)^2}]W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k \\
&+ [\sum_{k=1,k\neq m}^{r}\frac{\hat{\Phi}_k}{(\hat{\lambda}_m+\hat{\lambda}_k)^2}+\frac{\hat{\Phi}_m}{2\hat{\lambda}_m^2}-\sum_{k=1}^{n}\frac{\Phi_k}{(\hat{\lambda}_m+\lambda_k)^2}]W(-\hat{\lambda}_m)W(\hat{\lambda}_m)\cdot\hat{\Phi}_m \\
&+ \sum_{k=1}^{p}[G_r(-\hat{\lambda}_m)-G(-\hat{\lambda}_m)][\frac{\Psi_k}{(\hat{\lambda}_m+\gamma_k)^2}]W(\hat{\lambda}_m)\cdot\hat{\Phi}_m \\
&+ \sum_{k=1}^{p}[G_r(-\hat{\lambda}_m)-G(-\hat{\lambda}_m)]W(-\hat{\lambda}_m)[\frac{-\Psi_k}{(\hat{\lambda}_m-\gamma_k)^2}]\cdot\hat{\Phi}_m \\
&+ \sum_{k=1}^{p}[\frac{-\hat{\Phi}_m}{(\gamma_k+\hat{\lambda}_m)^2}][G(\gamma_k)-G_r(\gamma_k)]W(-\gamma_k)\cdot\Psi_k \\
&+ \sum_{k=1}^{p}[G(-\gamma_k)-G_r(-\gamma_k)][\frac{-\hat{\Phi}_m}{(\gamma_k-\hat{\lambda}_m)^2}]W(-\gamma_k)\cdot\Psi_k \qquad (3.4.4)
\end{aligned}
$$

where $\Phi_k, \hat{\Phi}_k$ and Ψ_k are as previously defined.

Proof. Let the partial fraction expansions of $G(s), G_r(s)$ and $W(s)$ be $G(s)=\sum_{k=1}^{n}\frac{\Phi_k}{s-\lambda_k}$, $G_r(s)=\sum_{k=1}^{r}\frac{\hat{\Phi}_k}{s-\hat{\lambda}_k}$ and $W(s)=\sum_{k=1}^{p}\frac{\Psi_k}{s-\gamma_k}$. Then, in addition to the derivatives in Theorem 2.1.2, we obtain following derivatives:

i) $\frac{\partial}{\partial\hat{\lambda}_m}G_r(-\gamma_j) = \frac{\hat{\Phi}_m}{(\gamma_j+\hat{\lambda}_m)^2}$

ii) $\frac{\partial}{\partial\hat{\lambda}_m}G_r(\gamma_j) = \frac{\hat{\Phi}_m}{(\gamma_j-\hat{\lambda}_m)^2}$

iii) $\frac{\partial}{\partial\hat{\lambda}_m}W(-\hat{\lambda}_j) = \begin{cases} \sum_{k=1}^{p}\frac{\Psi_k}{(\hat{\lambda}_m+\gamma_k)^2}, & j=m \\ 0, & j\neq m \end{cases}$

iv) $\frac{\partial}{\partial\hat{\lambda}_m}W(\hat{\lambda}_j) = \begin{cases} \sum_{k=1}^{p}\frac{-\Psi_k}{(\hat{\lambda}_m-\gamma_k)^2}, & j=m \\ 0, & j\neq m \end{cases}$

First we discuss $\frac{\partial J_w}{\partial\hat{\Phi}_m}$. We define

$$\begin{aligned} J_{w_1} &:= \sum_{k=1}^{n}(G(-\lambda_k)-G_r(-\lambda_k))W(-\lambda_k)W(\lambda_k)\cdot\Phi_k \\ J_{w_2} &:= \sum_{k=1}^{r}(G_r(-\hat{\lambda}_k)-G(-\hat{\lambda}_k))W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k \\ J_{w_3} &:= \sum_{k=1}^{p}(G(-\gamma_k)-G_r(-\gamma_k))W(-\gamma_k)(G(\gamma_k)-G_r(\gamma_k))\cdot\Psi_k, \end{aligned}$$

which leads to $\frac{\partial J_w}{\partial\hat{\Phi}_m} = \frac{\partial J_{w_1}}{\partial\hat{\Phi}_m} + \frac{\partial J_{w_2}}{\partial\hat{\Phi}_m} + \frac{\partial J_{w_3}}{\partial\hat{\Phi}_m}$. We get the following derivatives:

$$\frac{\partial J_{w_1}}{\partial\hat{\Phi}_m} = \frac{\partial}{\partial\hat{\Phi}_m}[\sum_{k=1}^{n}(G(-\lambda_k)-G_r(-\lambda_k))W(-\lambda_k)W(\lambda_k)\cdot\Phi_k]$$

$$= \sum_{k=1}^{n}(-\frac{\partial G_r(-\lambda_k)}{\partial \hat{\Phi}_m})W(-\lambda_k)W(\lambda_k)\cdot\Phi_k$$

$$= \sum_{k=1}^{n}[\frac{1}{\lambda_k+\hat{\lambda}_m}]W(-\lambda_k)W(\lambda_k)\cdot\Phi_k$$

$$\frac{\partial J_{w_2}}{\partial \hat{\Phi}_m} = \frac{\partial}{\partial \hat{\Phi}_m}[\sum_{k=1}^{r}(G_r(-\hat{\lambda}_k)-G(-\hat{\lambda}_k))W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k]$$

$$= [\sum_{k=1}^{r}(\frac{\partial G_r(-\hat{\lambda}_k)}{\partial \hat{\Phi}_m})W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k] \tag{3.4.5}$$

$$+ [G_r(-\hat{\lambda}_m)-G(-\hat{\lambda}_m)]W(-\hat{\lambda}_m)W(\hat{\lambda}_m)$$

$$= [\sum_{k=1}^{r}[\frac{-1}{\hat{\lambda}_k+\hat{\lambda}_m}]W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k] \tag{3.4.6}$$

$$+ [G_r(-\hat{\lambda}_m)-G(-\hat{\lambda}_m)]W(-\hat{\lambda}_m)W(\hat{\lambda}_m)$$

$$\frac{\partial J_{w_3}}{\partial \hat{\Phi}_m} = \frac{\partial}{\partial \hat{\Phi}_m}[\sum_{k=1}^{p}(G(-\gamma_k)-G_r(-\gamma_k))W(-\gamma_k)(G(\gamma_k)-G_r(\gamma_k))\cdot\Psi_k]$$

$$= \sum_{k=1}^{p}[(-\frac{\partial G_r(-\gamma_k)}{\partial \hat{\Phi}_m})W(-\gamma_k)(G(\gamma_k)-G_r(\gamma_k))\cdot\Psi_k$$

$$+ \quad (G(-\gamma_k) - G_r(-\gamma_k))W(-\gamma_k)(-\frac{\partial G_r(\gamma_k)}{\partial \hat{\Phi}_m}) \cdot \Psi_k]$$

$$= \quad \sum_{k=1}^{p}[\frac{1}{\gamma_k + \hat{\lambda}_m}]W(-\gamma_k)(G(\gamma_k) - G_r(\gamma_k)) \cdot \Psi_k$$

$$+ \quad \sum_{k=1}^{p}[G(-\gamma_k) - G_r(-\gamma_k)]W(-\gamma_k)[\frac{-1}{\gamma_k - \hat{\lambda}_m}] \cdot \Psi_k.$$

Next we discuss $\frac{\partial J_w}{\partial \hat{\lambda}_m}$.

$$\frac{\partial J_{w_1}}{\partial \hat{\lambda}_m} = \frac{\partial}{\partial \hat{\lambda}_m}[\sum_{k=1}^{n}(G(-\lambda_k) - G_r(-\lambda_k))W(-\lambda_k)W(\lambda_k) \cdot \Phi_k]$$

$$= \sum_{k=1}^{n}(-\frac{\partial G_r(-\lambda_k)}{\partial \hat{\lambda}_m})W(-\lambda_k)W(\lambda_k) \cdot \Phi_k$$

$$= \sum_{k=1}^{n}[\frac{-\hat{\Phi}_m}{(\lambda_k + \hat{\lambda}_m)^2}]W(-\lambda_k)W(\lambda_k) \cdot \Phi_k$$

$$\frac{\partial J_{w_2}}{\partial \hat{\lambda}_m} = \frac{\partial}{\partial \hat{\lambda}_m}[\sum_{k=1}^{r}(G_r(-\hat{\lambda}_k) - G(-\hat{\lambda}_k))W(-\hat{\lambda}_k)W(\hat{\lambda}_k) \cdot \hat{\Phi}_k]$$

$$
\begin{aligned}
&= \sum_{k=1}^{r}\left(\frac{\partial}{\partial\hat{\lambda}_m}(G_r(-\hat{\lambda}_k)-G(-\hat{\lambda}_k))\right)W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k \qquad (3.4.7)\\
&+ \sum_{k=1}^{r}(G_r(-\hat{\lambda}_k)-G(-\hat{\lambda}_k))\left(\frac{\partial}{\partial\hat{\lambda}_m}W(-\hat{\lambda}_k)\right)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k\\
&+ \sum_{k=1}^{r}(G_r(-\hat{\lambda}_k)-G(-\hat{\lambda}_k))W(-\hat{\lambda}_k)\left(\frac{\partial}{\partial\hat{\lambda}_m}W(\hat{\lambda}_k)\right)\cdot\hat{\Phi}_k\\
&= \sum_{k=1,k\neq m}^{r}\left[\frac{\hat{\Phi}_m}{(\hat{\lambda}_k+\hat{\lambda}_m)^2}\right]W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k\\
&+ \left[\sum_{k=1,k\neq m}^{r}\frac{\hat{\Phi}_k}{(\hat{\lambda}_m+\hat{\lambda}_k)^2}+\frac{\hat{\Phi}_m}{2\hat{\lambda}_m^2}-\sum_{k=1}^{n}\frac{\Phi_k}{(\hat{\lambda}_m+\lambda_k)^2}\right]W(-\hat{\lambda}_m)W(\hat{\lambda}_m)\cdot\hat{\Phi}_m\\
&+ \sum_{k=1}^{p}[G_r(-\hat{\lambda}_m)-G(-\hat{\lambda}_m)]\left[\frac{\Psi_k}{(\hat{\lambda}_m+\gamma_k)^2}\right]W(\hat{\lambda}_m)\cdot\hat{\Phi}_m\\
&+ \sum_{k=1}^{p}[G_r(-\hat{\lambda}_m)-G(-\hat{\lambda}_m)]W(-\hat{\lambda}_m)\left[\frac{-\Psi_k}{(\hat{\lambda}_m-\gamma_k)^2}\right]\cdot\hat{\Phi}_m
\end{aligned}
$$

$$
\frac{\partial J_{w_3}}{\partial\hat{\lambda}_m} = \frac{\partial}{\partial\hat{\lambda}_m}\left[\sum_{k=1}^{p}(G(-\gamma_k)-G_r(-\gamma_k))W(-\gamma_k)(G(\gamma_k)-G_r(\gamma_k))\cdot\Psi_k\right]
$$

$$
\begin{aligned}
&= \sum_{k=1}^{p}(-\frac{\partial}{\partial\hat{\lambda}_m}(-G_r(-\gamma_k)))W(-\gamma_k)(G(\gamma_k)-G_r(\gamma_k))\cdot\Psi_k \\
&+ \sum_{k=1}^{p}(G(-\gamma_k)-G_r(-\gamma_k))W(-\gamma_k)(-\frac{\partial}{\partial\hat{\lambda}_m}(G_r(\gamma_k)))\cdot\Psi_k \\
&= \sum_{k=1}^{p}[\frac{-\hat{\Phi}_m}{(\gamma_k+\hat{\lambda}_m)^2}][G(\gamma_k)-G_r(\gamma_k)]W(-\gamma_k)\cdot\Psi_k \\
&+ \sum_{k=1}^{p}[G(-\gamma_k)-G_r(-\gamma_k)][\frac{-\hat{\Phi}_m}{(\gamma_k-\hat{\lambda}_m)^2}]W(-\gamma_k)\cdot\Psi_k
\end{aligned}
$$

Since $\frac{\partial J_w}{\partial\hat{\Phi}_m}=\frac{\partial J_{w_1}}{\partial\hat{\Phi}_m}+\frac{\partial J_{w_2}}{\partial\hat{\Phi}_m}+\frac{\partial J_{w_3}}{\partial\hat{\Phi}_m}$ and $\frac{\partial J_w}{\partial\hat{\lambda}_m}=\frac{\partial J_{w_1}}{\partial\hat{\lambda}_m}+\frac{\partial J_{w_2}}{\partial\hat{\lambda}_m}+\frac{\partial J_{w_3}}{\partial\hat{\lambda}_m}$, we obtain the derivatives $\frac{\partial J_w}{\partial\hat{\Phi}_m}$ and $\frac{\partial J_w}{\partial\hat{\lambda}_m}$ as in (3.4.3) and (3.4.4). □

Considering the expressions $\frac{\partial J_w}{\partial\hat{\Phi}_m}$ and $\frac{\partial J_w}{\partial\hat{\lambda}_m}$, and setting them equal to zero, we realize that the interpolation conditions for the weighted case are much harder to derive than the Meier-Luenberger conditions (2.1.6) for the unweighted case. The Meier-Luenberger conditions were used in the Iterative Rational Krylov Algorithm (IRKA) proposed in [26]. IRKA constructs a reduced order system G_r which interpolates G at the mirror images of the poles of G_r. The motivation for this is that in this case the Meier-Luenberger conditions are satisfied and G_r is a local minimizer for the error expression J. These necessary optimality conditions for the weighted case would help to construct a reduced system system as in the unweighted case. Even though, as already mentioned, for the weighted case the general optimality conditions are much harder to derive than for the unweighted, the problem can be solved numerically. The problem becomes a root finding problem for the function f defined as

$$f : \mathbb{R}^{2r} \to \mathbb{R}^{2r} \text{ with } f(q) := f(\hat{\lambda}_1, \ldots, \hat{\lambda}_r, \hat{\Phi}_1, \ldots, \hat{\Phi}_r) := \nabla J_w = \begin{pmatrix} \frac{\partial J_w}{\partial \hat{\lambda}_1} \\ \vdots \\ \frac{\partial J_w}{\partial \hat{\lambda}_r} \\ \frac{\partial J_w}{\partial \hat{\Phi}_1} \\ \vdots \\ \frac{\partial J_w}{\partial \hat{\Phi}_r} \end{pmatrix}$$

Then, since $G_r(s) = \sum_{k=1}^{r} \frac{\hat{\Phi}_k}{s-\hat{\lambda}_k}$, the solution q of $f(q) = 0$ yields a reduced order system for which the necessary conditions for weighted-$\mathcal{H}_2$ optimality, namely $\frac{\partial J_w}{\partial \hat{\Phi}_m} = \frac{\partial J_w}{\partial \hat{\lambda}_m}$=0, hold, where $\frac{\partial J_w}{\partial \hat{\Phi}_m}$ and $\frac{\partial J_w}{\partial \hat{\lambda}_m}$ are as given in Theorem 3.4.4. We suggest the following numerical solution to the problem:

Algorithm 3.4.1. A numerical solution for the optimal weighted-$\mathcal{H}_2$ necessary conditions:

i) make an initial guess for $q = [\hat{\lambda}_1, \ldots, \hat{\lambda}_r, \hat{\Phi}_1, \ldots, \hat{\Phi}_r]$

ii) solve $f(q) = 0$*, for example by using Newton-Raphson*

iii) construct the reduced order system $G_r(s) = \sum_{k=1}^{r} \frac{\hat{\Phi}_k}{s-\hat{\lambda}_k}$ *for which the necessary conditions for* $\mathcal{H}_2$ *optimality hold*

Notice that with a Newton method, we need to compute the derivatives of f, i.e. $\frac{\partial f}{\partial q} = \frac{\partial f_i}{\partial q_j}$ for $i, j = 1, ..., 2r$. This means we need to compute the derivatives $\frac{\partial J_w}{\partial \hat{\Phi}_m}$ (3.4.3) and $\frac{\partial J_w}{\partial \hat{\lambda}_m}$ (3.4.4). Computing these derivatives will require computing the complete eigendecomposition of G

and W. Moreover we need to compute the corresponding residues of G and W. If the weight is of similarly high dimension as the original system, this becomes even a harder task. The conclusion is that, unlike in the unweighted case [26], the solution becomes very expensive. In Chapter 4 we will present an algorithm for the weighted case which is similarly effective as IRKA [26] for the unweighted case.

3.5 Structured orthogonality optimality conditions

In [26] new conditions for unweighted-$\mathcal{H}_2$ optimality have been presented. These new conditions occur as structured orthogonality conditions. The authors of [26] considered the optimal $\mathcal{H}_2$ model reduction problem (2.1.4). Using their structured orthogonality conditions they showed that if $G_r(s)$ has simple poles at $\hat{\lambda}_1, \ldots, \hat{\lambda}_r$ and $G_r(s)$ is a local minimizer of dimension r for the optimal $\mathcal{H}_2$ model reduction problem then $G_r(s)$ interpolates both $G(s)$ and its derivative at $-\hat{\lambda}_i, i = 1, ..., r$, i.e.

$$G_r(-\widetilde{\lambda}_i) = G(-\widetilde{\lambda}_i) \qquad \text{and} \qquad G_r'(-\widetilde{\lambda}_i) = G'(-\widetilde{\lambda}_i), \quad \text{for } i = 1, \ldots, r. \tag{3.5.1}$$

For the weighted-$\mathcal{H}_2$ model reduction problem interpolation conditions similar to (3.5.1) are unknown. In the next theorem, new optimality conditions for the weighted-$\mathcal{H}_2$ model reduction problem are presented. They are are similar to the structured orthogonality conditions for the unweighted case which were presented in [26].

Theorem 3.5.1. *Let* $\lambda_1, \lambda_2, \ldots, \lambda_n, \hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_r \subset \mathbb{C}$ *and* $\gamma_1, \gamma_2, \ldots, \gamma_p \subset \mathbb{C}$ *be distinct*

points in the open left half plane. Assume $\lambda_1, \lambda_2, \ldots, \lambda_n$ are the poles of G, $\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_r$ are the poles of the reduced system G_r, and $\gamma_1, \gamma_2, \ldots, \gamma_p$ are the poles of the weighting system W, respectively. Define $\mathcal{M}(\boldsymbol{\lambda})$ to be the set of all proper rational functions that have simple poles exactly at $\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_r, \gamma_1, \gamma_2, \ldots, \gamma_p$. Then G_r with $G_r \cdot W \in \mathcal{M}(\boldsymbol{\lambda})$ solves

$$\|(G - G_r) \cdot W\|_{\mathcal{H}_2} = \min_{\widetilde{G}_r \cdot W \in \mathcal{M}(\boldsymbol{\lambda})} \|(G - \widetilde{G}_r) \cdot W\|_{\mathcal{H}_2} \tag{3.5.2}$$

if and only if

$$\langle (G - G_r) \cdot W,\, H \cdot W \rangle_{\mathcal{H}_2} = 0 \qquad \textit{for all } \; H \qquad \textit{with } \; H \cdot W \in \mathcal{M}(\boldsymbol{\lambda}).$$

Furthermore, the solution exists and is unique.

Proof. Since $\mathcal{M}(\boldsymbol{\lambda})$ is a closed subspace of $\mathcal{H}_2$ the equivalence follows from the classic projection theorem in Hilbert spaces, which can be found in [40]. □

Later we will use the following inner product property [40]:

$$\langle (G(s) - G_r(s)) \cdot W(s),\, H(s) \cdot W(s) \rangle_{\mathcal{H}_2} = \langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\, H(s) \rangle_{\mathcal{H}_2}$$

Now let $W \neq 0$ be a weighting system with poles at $\gamma_1, \gamma_2, \ldots, \gamma_p$. A reduced order system G_r is a local minimizer for

$$\|(G - G_r) \cdot W\|_{\mathcal{H}_2} = \min_{\substack{\dim(\widetilde{G}_r) = r \\ \widetilde{G}_r stable}} \|(G - \widetilde{G}_r) \cdot W\|_{\mathcal{H}_2} \tag{3.5.3}$$

if for $\epsilon > 0$ sufficiently small,

$$\|(G - G_r) \cdot W\|_{\mathcal{H}_2} \leq \|(G - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2},$$

for all stable dynamical systems , $\widetilde{G}_r^{(\varepsilon)}$ with $\dim(\widetilde{G}_r^{(\varepsilon)})$=r and $\|(G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2} \leq C\,\varepsilon$,where C is a constant that may depend on the particular family $\widetilde{G}_r^{(\varepsilon)}$ considered.

Theorem 3.5.2. *If G_r is a local minimizer as described and has simple poles then*

$$\langle (G - G_r) \cdot W,\ (G_r \cdot H_1 + H_2) \cdot W \rangle_{\mathcal{H}_2} = 0 \tag{3.5.4}$$

for all real dynamical systems, H_1 and H_2, having the same poles with the same multiplicities as G_r.

Proof. If H_1=0 then

$$\langle (G - G_r) \cdot W,\ H_2 \cdot W \rangle_{\mathcal{H}_2} = 0$$

follows by Theorem 3.5.1. This means it suffices to show that

$$\langle (G - G_r) \cdot W,\ G_r \cdot H \cdot W \rangle_{\mathcal{H}_2} = 0$$

for all H having the same poles with the same multiplicities as G_r. This is equivalent to showing

$$\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ G_r(s) \cdot H(s) \rangle_{\mathcal{H}_2} = 0$$

for all H having the same poles with the same multiplicities as G_r.

Suppose that $\{\widetilde{G}_r^{(\varepsilon)}\}_{\varepsilon>0}$ is a family of real stable dynamical systems with $\dim(\widetilde{G}_r^{(\varepsilon)}) = r$ and $\|(G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2} < C\varepsilon$ for some constant $C > 0$. Then for all $\varepsilon > 0$ sufficiently small,

$$\begin{aligned}
\|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2 &\leq \|(G - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2}^2 \\
&\leq \|(G - G_r) \cdot W + (G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2}^2 \\
&\leq \|(G - G_r) \cdot W\|_{\mathcal{H}_2}^2 + 2\left\langle (G - G_r) \cdot W,\ (G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W \right\rangle_{\mathcal{H}_2} \\
&+ \|(G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2}^2
\end{aligned}$$

For $\varepsilon > 0$ sufficiently small this implies,

$$0 \leq 2 \left\langle (G - G_r) \cdot W,\ (G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W \right\rangle_{\mathcal{H}_2} + \|(G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2}^2$$

Suppose $\hat{\lambda}_1,\ \hat{\lambda}_2,\ \ldots\ \hat{\lambda}_r$ are the poles of G_r, and they are ordered so that the first m_R are real and the next m_C are in the upper half plane. Write $\hat{\lambda}_i = \alpha_i + \imath\beta_i$. We can represent any real rational function, which has the same poles as $G_r(s)$, as

$$H(s) = \sum_{i=1}^{m_R} \frac{\gamma_i}{s - \hat{\lambda}_i} + \sum_{i=m_R+1}^{m_R+m_C} \frac{\rho_i(s - \alpha_i) + \tau_i}{(s - \alpha_i)^2 + \beta_i^2},$$

with arbitrary real valued choices for γ_i, ρ_i, and τ_i.

Now suppose $\hat{\lambda}$ is a real pole for G_r and that

$$\left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \frac{G_r(s)}{s - \hat{\lambda}} \right\rangle_{\mathcal{H}_2} \neq 0. \tag{3.5.5}$$

Let $G_r(s) = \dfrac{p_{r-1}(s)}{(s - \hat{\lambda})\, q_{r-1}(s)}$, where $p_{r-1},\ q_{r-1} \in \mathcal{P}_{r-1}$ are real polynomials, and define

$$\widetilde{G}_r^{(\varepsilon)}(s) = \frac{p_{r-1}(s)}{\left[s - \hat{\lambda} - (\pm\varepsilon)\right]\, q_{r-1}(s)},$$

such that the sign of $\pm\varepsilon$ matches the sign of $\left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \dfrac{G_r(s)}{s - \hat{\lambda}} \right\rangle_{\mathcal{H}_2}$.

We obtain

$$\widetilde{G}_r^{(\varepsilon)}(s) = \frac{p_{r-1}(s)}{\left[s - \hat{\lambda} - (\pm\varepsilon)\right] q_{r-1}(s)}$$

$$= \frac{p_{r-1}(s)}{q_{r-1}(s)[s-\hat{\lambda}][1 - \frac{\pm\varepsilon}{s-\hat{\lambda}}]}$$

$$= \frac{p_{r-1}(s)}{q_{r-1}(s)[s-\hat{\lambda}]} \cdot \frac{1}{[1 - \frac{\pm\varepsilon}{s-\hat{\lambda}}]}$$

$$= \frac{p_{r-1}(s)}{q_{r-1}(s)[s-\hat{\lambda}]} \cdot \sum_{k=0}^{\infty} (\frac{\pm\varepsilon}{s-\hat{\lambda}})^k$$

$$= \frac{p_{r-1}(s)}{q_{r-1}(s)[s-\hat{\lambda}]} \cdot (1 + \frac{\pm\varepsilon}{s-\hat{\lambda}} + \mathcal{O}(\varepsilon^2))$$

$$= G_r(s) \pm \varepsilon \frac{p_{r-1}(s)}{(s-\hat{\lambda})^2\, q_{r-1}(s)} + \mathcal{O}(\varepsilon^2).$$

This yields $G_r(s) - \widetilde{G}_r^{(\varepsilon)}(s) = \mp\varepsilon \dfrac{G_r(s)}{s-\hat{\lambda}} + \mathcal{O}(\varepsilon^2)$ and

$$\left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ G_r - \widetilde{G}_r^{(\varepsilon)} \right\rangle_{\mathcal{H}_2} =$$
$$-\varepsilon \left| \left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \frac{G_r(s)}{s-\lambda} \right\rangle_{\mathcal{H}_2} \right| + \mathcal{O}(\varepsilon^2).$$

For $\varepsilon \to 0$ following equation

$$0 \leq 2 \left\langle (G - G_r) \cdot W,\ (G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W \right\rangle_{\mathcal{H}_2} + \|(G_r - \widetilde{G}_r^{(\varepsilon)}) \cdot W\|_{\mathcal{H}_2}^2$$

implies
$0 < \left| \left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \dfrac{G_r(s)}{s - \hat{\lambda}} \right\rangle_{\mathcal{H}_2} \right| \leq C\varepsilon$ for some constant C. However, this contradicts

$$\left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \frac{G_r(s)}{s - \hat{\lambda}} \right\rangle_{\mathcal{H}_2} \neq 0. \tag{3.5.6}$$

Now suppose that $\hat{\lambda} = \alpha + \imath\beta$ is a pole for G_r, where the imaginary part is nonzero, $\beta \neq 0$, thus is one of a conjugate pair of poles for G_r. Suppose further that

$$\left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \frac{G_r(s)}{(s-\alpha)^2 + \beta^2} \right\rangle_{\mathcal{H}_2} \neq 0 \quad \text{and}$$
$$\left\langle (G(s) - G_r(s)) \cdot W(s) \cdot W(-s),\ \frac{(s-\alpha)\, G_r(s)}{(s-\alpha)^2 + \beta^2} \right\rangle_{\mathcal{H}_2} \neq 0.$$

Let $G_r(s) = \dfrac{p_{r-1}(s)}{[(s-\alpha)^2 + \beta^2]\, q_{r-2}(s)}$, where $p_{r-1} \in \mathcal{P}_{r-1}$ and $q_{r-2} \in \mathcal{P}_{r-2}$ are real polynomials. Exactly the same arguments as in the previous case lead to the remaining assertions. In particular,

$$\text{to show} \quad \left\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ \frac{G_r(s)}{(s-\alpha)^2+\beta^2}\right\rangle_{\mathcal{H}_2}=0,$$

$$\text{consider} \quad \widetilde{G}_r^{(\varepsilon)}(s)\ =\frac{p_{r-1}(s)}{[(s-\alpha)^2+\beta^2-(\pm\varepsilon)]\,q_{r-2}(s)};$$

$$\text{to show} \quad \left\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ \frac{(s-\alpha)\,G_r(s)}{(s-\alpha)^2+\beta^2}\right\rangle_{\mathcal{H}_2}=0,$$

$$\text{consider} \quad \widetilde{G}_r^{(\varepsilon)}(s)\ =\frac{p_{r-1}(s)}{[(s-\alpha-(\pm\varepsilon))^2+\beta^2]\,q_{r-2}(s)}.$$

For G_r being a locally optimal $\mathcal{H}_2$ reduced order model follows

$$\begin{aligned}\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ G_r\cdot H_1+H_2\rangle_{\mathcal{H}_2}=&\\ \sum_{i=1}^{m_R}\gamma_i\left\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ \frac{G_r(s)}{s-\hat{\lambda}_i}\right\rangle_{\mathcal{H}_2}&\\ +\sum_{i=m_R+1}^{m_R+m_C}\rho_i\left\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ \frac{(s-\alpha_i)\,G_r(s)}{(s-\alpha_i)^2+\beta_i^2}\right\rangle_{\mathcal{H}_2}&\\ +\sum_{i=m_R+1}^{m_R+m_C}\tau_i\left\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ \frac{G_r(s)}{(s-\alpha_i)^2+\beta_i^2}\right\rangle_{\mathcal{H}_2}&\\ +\langle (G(s)-G_r(s))\cdot W(s)\cdot W(-s),\ H_2(s)\rangle_{\mathcal{H}_2}=0,&\end{aligned}$$

and the conclusion follows.

□

Chapter 4

An IRKA for the weighted-$\mathcal{H}_2$ model reduction problem

4.1 Motivation for the proposed algorithm

For the weighted-$\mathcal{H}_2$ error we derived the following formula:

$$\begin{aligned} J_w &= \sum_{k=1}^{n}(G(-\lambda_k) - G_r(-\lambda_k))W(-\lambda_k)W(\lambda_k)\cdot\Phi_k \\ &+ \sum_{k=1}^{r}(G_r(-\hat{\lambda}_k) - G(-\hat{\lambda}_k))W(-\hat{\lambda}_k)W(\hat{\lambda}_k)\cdot\hat{\Phi}_k \\ &+ \sum_{k=1}^{p}(G(-\gamma_k) - G_r(-\gamma_k))W(-\gamma_k)(G(\gamma_k) - G_r(\gamma_k))\cdot\Psi_k. \end{aligned} \tag{4.1.1}$$

Moreover, for the unweighted-$\mathcal{H}_2$ model reduction problem, we have derived the error ex-

pression J (2.1.5). Remember that the error was due to the mismatch of G and G_r at the mirror images of the full order poles $\lambda_1, ..., \lambda_n$ and the reduced order poles $\hat{\lambda}_1, ..., \hat{\lambda}_r$. As we can see, for the weighted case the error is due to the mismatch of G and G_r at the mirror images of the full order poles $\lambda_1, ..., \lambda_n$, the reduced order poles $\hat{\lambda}_1, ..., \hat{\lambda}_r$ and the poles of the weight $\gamma_1, ..., \gamma_p$. The idea now is to modify IRKA, presented in [26], in order to get a reduced order system G_r such that the residues of largest absolute value are cancelled in the error expression J_w (4.1.1). In other words, we use parts of IRKA in order to construct a reduced order system G_r that interpolates the full order system G at the mirror images of λ_k and γ_k, where λ_k and γ_k are the poles of G and W which belong to the highest residues Φ_k and Ψ_k, respectively. This will be the first step in our algorithm. In the next step we use IRKA to force $G_r(-\hat{\lambda}_k) = G(-\hat{\lambda}_k)$. However, the difference is that with W-IRKA we only force $G_r(-\hat{\lambda}_k) = G(-\hat{\lambda}_k)$, whereas IRKA forces in addition $G_r'(-\hat{\lambda}_k) = G'(-\hat{\lambda}_k)$ for $k = 1, ..., r$.

4.2 The proposed algorithm

Having given the motivation behind W-IRKA, we present the algorithm we propose. This algorithm reduces, for a given weighting system W, the original system G of order n to a reduced order model G_r of order r, such that $\|(G - G_r) \cdot W\|_{\mathcal{H}_2}$ is small.

Algorithm 4.2.1. W-IRKA:

1. *Fix a convergence tolerance tol and make an initial selection of σ_i, for $i = 1, \ldots, r$ in such a way that $\sigma_1, ..., \sigma_\vartheta$ are the poles of the original system G which belong to the ϑ highest residues and $\sigma_{\vartheta+1}, ..., \sigma_{\vartheta+\rho}$ are the poles of the weight W which belong to the ρ highest residues ($r=\vartheta + \rho$).*

2. *Choose* $\mathbf{V}_r$ *and* $\mathbf{W}_r$ *so that* $Ran(\mathbf{V}_r) = span\left\{(\sigma_1\mathbf{I}-\mathbf{A})^{-1}\mathbf{b},\ \cdots,\ (\sigma_r\mathbf{I}-\mathbf{A})^{-1}\mathbf{b}\right\}$, $Ran(\mathbf{W}_r) = span\left\{(\sigma_1\mathbf{I}-\mathbf{A}^T)^{-1}\mathbf{c},\ \cdots,\ (\sigma_r\mathbf{I}-\mathbf{A}^T)^{-1}\mathbf{c}\right\}$, *and* $\mathbf{W}_r^T\mathbf{V}_r = \mathbf{I}$.

3. $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r$

4. *Assign* $\sigma_i \longleftarrow -\lambda_i(\mathbf{A}_r)$ *for* $i = 1, \ldots, r$

5. *while (relative change in* $\{\sigma_i\} >$ *tol)*

   (a) *Update* $\mathbf{V}_r$ *so that* $Ran(\mathbf{V}_r) = span\left\{(\sigma_1\mathbf{I}-\mathbf{A})^{-1}\mathbf{b},\ \cdots,\ (\sigma_r\mathbf{I}-\mathbf{A})^{-1}\mathbf{b}\right\}$ *and* $\mathbf{W}_r^T\mathbf{V}_r = \mathbf{I}$

   (b) $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r$,

   (c) *Assign* $\sigma_i \longleftarrow -\lambda_i(\mathbf{A}_r)$ *for* $i = 1, \ldots, r$.

6. $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r$, $\mathbf{b}_r = \mathbf{W}_r^T\mathbf{b}$, $\mathbf{c}_r^T = \mathbf{c}^T\mathbf{V}_r$

Upon convergence W-IRKA constructs a reduced system G_r such that $G_r(-\hat{\lambda}_i) = G(-\hat{\lambda}_i)$ for $i = 1..., r$, where $\hat{\lambda}_1, ..., \hat{\lambda}_r$ are the poles of G_r. Moreover, W-IRKA forces $G_r(-\lambda_j) = G(-\lambda_j)$ for $j = 1, ..., \vartheta$ and $G_r(-\gamma_k) = G(-\gamma_k)$ for $k = 1, ..., \rho$, where λ_j and γ_k are the poles of G and W which belong to the dominant residues Φ_j and Ψ_k, respectively. As for the unweighted case with IRKA, we obtain a numerically effective rational Krylov algorithm. W-IRKA uses the dominant poles as initialization. However, computing the full eigenvalue decomposition is very expensive. Instead, we suggest to compute the ϑ and ρ dominant poles of G and W with the *dominant pole algorithm* proposed by Rommes [41]. This algorithm allows computing the dominant poles in a numerically effective way. In the appendix we present the matlab code for W-IRKA. Similarly as IRKA [26], W-IRKA for weighted model reduction converged quickly for most cases.

4.3 Numerical examples

In this section, we illustrate the performance of the proposed algorithm with three different dynamical systems. In every example ϑ will always denote the number of ϑ dominant residues for G and ρ will always denote the number of ρ dominant residues for G_r, respectively. Moreover, we present the weighted errors we obtained with W-IRKA and compare the errors obtained with W-IRKA to the errors obtained with *weighted balanced truncation* [5, 22]. All three examples come from a controller reduction problem as explained in Section 3.2. In each case, for a given plant, a controller is designed to dampen the oscillations in the impulse response. Then, W-IRKA is used to obtain a reduced order controller by solving the underlying weighted-$\mathcal{H}_2$ problem as in (3.2.2).

4.3.1 Reduction of building model controller 1

In the first example we will reduce a controller which is designed to dampen the oscillations for a building model. We present the absolute value of the 10 highest residues of K and W, which will be used to initialize W-IRKA. Φ_{42}^N will denote the vector with the 10 highest normalized residues of K, and Ψ_{42}^N will denote the vector with the 10 highest normalized residues of W. In this example we use W-IRKA in order to reduce a controller of order 42.

The normalized residue vectors are:

$$\Phi_{42}^N = \begin{bmatrix} 1.0000 \\ 0.8406 \\ 0.8406 \\ 0.1544 \\ 0.1544 \\ 0.1150 \\ 0.1150 \\ 0.0652 \\ 0.0652 \\ 0.0419 \end{bmatrix} \qquad \Psi_{42}^N = \begin{bmatrix} 1.0000 \\ 1.0000 \\ 0.7230 \\ 0.7230 \\ 0.6943 \\ 0.6943 \\ 0.5085 \\ 0.5085 \\ 0.3804 \\ 0.3804 \end{bmatrix}$$

Below, in Tables 4.1-4.3, we show the weighted-$\mathcal{H}_2$ error norms for different reduction orders r, using the proposed Algorithm 4.2.1.

residue combinations	ϑ=9, ρ=0	ϑ=7, ρ=2	ϑ=3, ρ=6	ϑ=1, ρ=8	ϑ=0, ρ=10
weighted-$\mathcal{H}_2$ error	4.4003	1.1117	0.9785	0.7199	0.2049

Table 4.1: Reduction of order 42 to order 9/10

residue combinations	ϑ=11, ρ=0	ϑ=9, ρ=2	ϑ=5, ρ=6	ϑ=3, ρ=8	ϑ=1, ρ=10	ϑ=0, ρ=10
weighted-$\mathcal{H}_2$ error	4.5144	0.8661	0.9335	0.2242	0.1882	0.2049

Table 4.2: Reduction of order 42 to order 10/11

residue comb.	ϑ=18, ρ=0	ϑ=12, ρ=6	ϑ=9, ρ=10	ϑ=7, ρ=12	ϑ=3, ρ=16	ϑ=1, ρ=18
weighted $\mathcal{H}_2$-error	1.7762	0.4891	0.0707	0.0702	0.0883	0.01875

Table 4.3: Reduction of order 42 to order 18/19

Considering the normalized residue vectors Φ_{42}^N and Ψ_{42}^N, we observe that after the third

component of Φ_{42}^N the values drop significantly, whereas the values of Ψ_{42}^N are still dominant. The Tables 4.1-4.3 show that the weighted-$\mathcal{H}_2$ error decreases as we take more dominant poles of W instead of G, as initialization. The conclusion is that there is a connection between the drop in the normalized vectors and the obtained weighted-$\mathcal{H}_2$ errors. Due to the drop in the normalized vectors we suggested choosing ϑ=3. The Tables 4.1-4.3 illustrate that there are better choices for ϑ. However, for every reduction order r we obtain a much better result with ϑ=3 than with choosing the complete initialization to be poles of K, i.e. ρ=0. Moreover, the result for ϑ=3 is close to the minimum case.

In Figure 4.1 and Figure 4.2 we show the bode plots of the transfer functions $K(s)$ and $K_r(s)$ for r=11 and r=17, respectively, where $K_r(s)$ was constructed with the proposed algorithm. As both figures illustrate, the reduced model $K_r(s)$ matches $K(s)$ well.

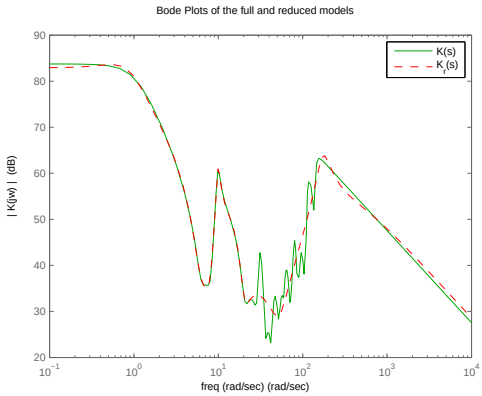

Figure 4.1: Reduction to order 11 with ϑ=3, ρ=8

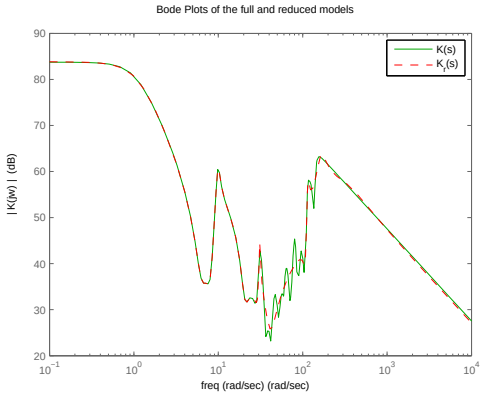

Figure 4.2: Reduction to order 17 with ϑ=7, ρ=10

Comparison to Weighted Balanced Truncation

Table 4.4 presents the weighted-$\mathcal{H}_2$ errors resulting from weighted balanced truncation (WBT) and W-IRKA.

reduction order r	9	10	11	18	19
WBT $\mathcal{H}_2$-error	1.3266	1.0307	0.32061	0.15209	0.12969
best W-IRKA error	0.7199	0.2049	0.2242	0.4891	0.0702

Table 4.4: Comparison W-IRKA ($\mathcal{H}_2$) to WBT ($\mathcal{H}_2$), n=42

Evidently, the proposed algorithm outperforms weighted balanced truncation for virtually every chosen reduction order r. Table 4.5 presents the weighted-$\mathcal{H}_\infty$ errors resulting from W-IRKA and weighted balanced truncation (WBT). In this comparison W-IRKA outperforms weighted balanced truncation for every r.

reduction order r	9	10	11	18	19
WBT $\mathcal{H}_\infty$-error	1.0675	0.8232	0.1553	0.0683	0.04935
best W-IRKA error	0.9375	0.0956	0.0964	0.0321	0.0246

Table 4.5: Comparison W-IRKA ($\mathcal{H}_\infty$) to WBT ($\mathcal{H}_\infty$), n=42

4.3.2 Reduction of building model controller 2

The second example is from [6]. Here we reduce a controller that is designed for the building model for the Los Angeles University Hospital. The building has 8 floors with 3 degrees of freedom (displacements in x and y direction, and rotation). Both, the controller and the plant, have order 48. We present the absolute value of the 10 highest residues of K and W. Φ_{48}^N will denote the vector with the 10 highest normalized residues of K, and Ψ_{48}^N will denote the vector with the 10 highest normalized residues of W.

The normalized residue vectors are:

$$\Phi_{48}^N = \begin{bmatrix} 1.0000 \\ 1.0000 \\ 0.0286 \\ 0.0286 \\ 0.0088 \\ 0.0088 \\ 0.0080 \\ 0.0080 \\ 0.0060 \\ 0.0060 \end{bmatrix} \qquad \Psi_{48}^N = \begin{bmatrix} 1.0000 \\ 1.0000 \\ 0.8416 \\ 0.8416 \\ 0.3935 \\ 0.3935 \\ 0.2646 \\ 0.2646 \\ 0.0951 \\ 0.0951 \end{bmatrix}$$

Below, in Tables 4.6-4.8, we show weighted-$\mathcal{H}_2$ error norms for different r values using the

proposed Algorithm.

residue combinations	ϑ=10, ρ=0	ϑ=8, ρ=2	ϑ=6, ρ=4	ϑ=4, ρ=6	ϑ=2, ρ=8
weighted-$\mathcal{H}_2$ error	1.3911	1.0841	1.027	0.7068	0.6677

Table 4.6: Reduction of order 48 to order 10

residue combinations	ϑ=12, ρ=2	ϑ=10, ρ=4	ϑ=8, ρ=6	ϑ=6, ρ=8
weighted-$\mathcal{H}_2$ error	1.343	0.6477	0.304	0.1538
residue combinations	ϑ=4, ρ=10	ϑ=2, ρ=12	ϑ=0, ρ=14	
weighted-$\mathcal{H}_2$ error	0.1425	0.1351	0.2224	

Table 4.7: Reduction of order 48 to order 14

residue combinations	ϑ=18, ρ=0	ϑ=16, ρ=2	ϑ=14, ρ=4	ϑ=12, ρ=6	ϑ=10, ρ=8
weighted-$\mathcal{H}_2$ error	1.4526	1.2016	0.3495	0.5972	0.2273
residue combinations	ϑ=8, ρ=10	ϑ=6, ρ=12	ϑ=4, ρ=14	ϑ=2, ρ=16	ϑ=0, ρ=18
weighted-$\mathcal{H}_2$ error	0.1052	0.1084	0.1037	0.1028	0.102

Table 4.8: Reduction of order 48 to order 18

Considering the normalized residue vectors Φ_{48}^N and Ψ_{48}^N, we observe that after the second component of Φ_{48}^N the values drop significantly, whereas the values of Ψ_{48}^N are still dominant. The Tables 4.6-4.8 show that the weighted-$\mathcal{H}_2$ error decreases as we take more dominant poles of W instead of K, as initialization. Moreover we observe that choosing ϑ=2 (2 dominant residues of K) is a very good choice for the initialization. Table 4.7 illustrates the error is decreasing as we take more dominant poles of W instead of K. Moreover, we observe that the error again increases as we choose ϑ <2. The conclusion is that there is a connection between the drop in the normalized vector Φ_{48}^N after the second component and the obtained weighted-$\mathcal{H}_2$ errors. It seems to be a good choice to choose ϑ and ρ accordingly to the drops in the normalized residue vectors.

In Figure 4.3 and Figure 4.4 we show the bode plots of the transfer functions $K(s)$ and $K_r(s)$ for r=14 and r=16, where $K_r(s)$ was constructed with the proposed algorithm. Once more K_r replicated K very well.

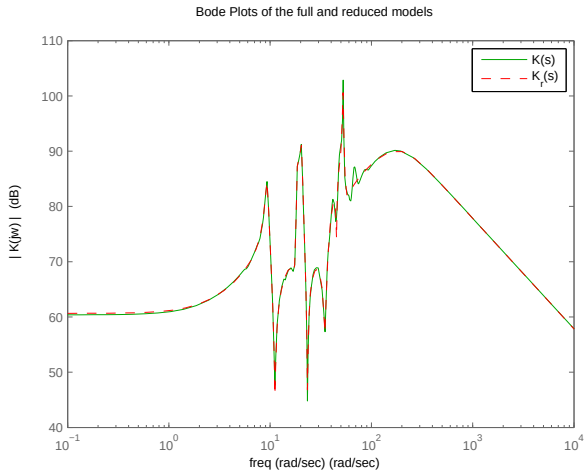

Figure 4.3: Reduction to order 14 with ϑ=4, ρ=10

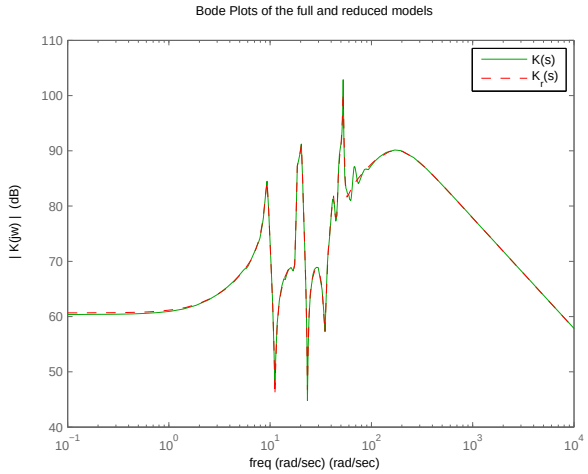

Figure 4.4: Reduction to order 16 with ϑ=4, ρ=12

Comparison to Weighted Balanced Truncation

Table 4.9 presents the weighted-$\mathcal{H}_2$ errors resulting from weighted balanced truncation (WBT) and W-IRKA.

reduction order r	8	10	12	18	20
WBT $\mathcal{H}_2$ error	1.7519	2.1080	1.1723	0.12149	0.13104
best W-IRKA error	0.7272	0.6677	0.3576	0.1028	0.075

Table 4.9: Comparison W-IRKA ($\mathcal{H}_2$) to WBT ($\mathcal{H}_2$), n=48

We observe that the proposed algorithm yields good results compared to weighted balanced truncation. In this example W-IRKA outperforms weighted balanced truncation for every chosen reduction order r. Table 4.10 presents the weighted-$\mathcal{H}_\infty$ errors resulting from W-IRKA and weighted balanced truncation (WBT). We see that even in this case W-IRKA yields better results for many r.

reduction order r	8	10	12	18	20
WBT $\mathcal{H}_\infty$ error	1.83	1.409	0.5286	0.0498	0.083
best W-IRKA error	0.8924	0.2128	0.1562	0.0663	0.0516

Table 4.10: Comparison W-IRKA ($\mathcal{H}_\infty$) to WBT ($\mathcal{H}_\infty$), n=48

4.3.3 Reduction of a CD-Player controller

In the last example we will reduce a controller for a CD-Player [26]. As in the previous examples, we present the absolute value of the 10 highest residues of K and W. Φ_{120}^N will denote the vector with the 10 highest normalized residues of K, and Ψ_{120}^N will denote the vector with the 10 highest normalized residues of W. In this example we use W-IRKA in order to reduce a controller of order 120. The normalized residue vectors are:

$$\Phi_{120}^N = \begin{bmatrix} 1.0000 \\ 0.0857 \\ 0.0183 \\ 0.0183 \\ 0.0059 \\ 0.0059 \\ 0.0013 \\ 0.0012 \\ 0.0012 \\ 0.0007 \end{bmatrix} \qquad \Psi_{120}^N = \begin{bmatrix} 1.0000 \\ 1.0000 \\ 0.6083 \\ 0.6083 \\ 0.1613 \\ 0.1613 \\ 0.1599 \\ 0.1599 \\ 0.1164 \\ 0.1164 \end{bmatrix}$$

Below, in Tables 4.11-4.13, we show weighted-$\mathcal{H}_2$ error norms for different r values using the proposed Algorithm.

residue combinations	ϑ=9, ρ=0	ϑ=6, ρ=4	ϑ=4, ρ=6	ϑ=1, ρ=8	ϑ=0, ρ=10
weighted-$\mathcal{H}_2$ error	97.1797	6.2738	19.5853	18.9714	19.508

Table 4.11: Reduction of order 120 to 9/10

residue combinations	ϑ=15, ρ=0	ϑ=13, ρ=2	ϑ=7, ρ=8	ϑ=6, ρ=10	ϑ=4, ρ=12
weighted-$\mathcal{H}_2$ error	46.0025	15.0767	3.8286	4.3065	4.3066

Table 4.12: Reduction of order 120 to 15/16

residue comb.	ϑ=25, ρ=0	ϑ=19, ρ=6	ϑ=13, ρ=12	ϑ=7, ρ=18	ϑ=1, ρ=24
weighted-$\mathcal{H}_2$ error	47.7674	4.0405	4.1169	3.6511	3.4627

Table 4.13: Reduction of order 120 to 24/25

Considering the normalized residue vectors Φ_{120}^N and Ψ_{120}^N, we observe that after the first component of Φ_{120}^N the values drop significantly, whereas the values of Ψ_{120}^N are still dominant. The Tables 4.11-4.13 show that the weighted-$\mathcal{H}_2$ error decreases as we take more dominant poles of W instead of K, as initialization. Moreover we observe that choosing ϑ=1 (the highest residue of K) is a good choice for the initialization. The conclusion is that there is a connection between the drop in the normalized vector Φ_{120}^N after the first component and the obtained weighted-$\mathcal{H}_2$ errors. It seems to be a good choice to choose the number of the dominant poles accordingly to the drop in the normalized residue vectors.

In Figures 4.5-4.7 we show the bode plots of the transfer functions $K(s)$ and $K_r(s)$ for r=15, r=19 and r=23, where $K_r(s)$ was constructed with the proposed algorithm. The figures below illustrate that K_r matches K well.

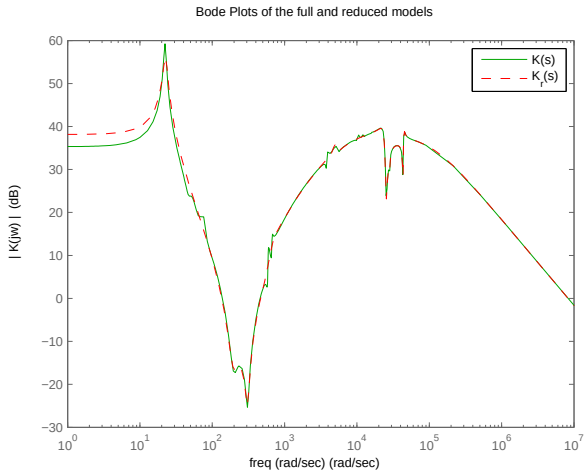

Figure 4.5: Reduction to order 15 with ϑ=7, ρ=8

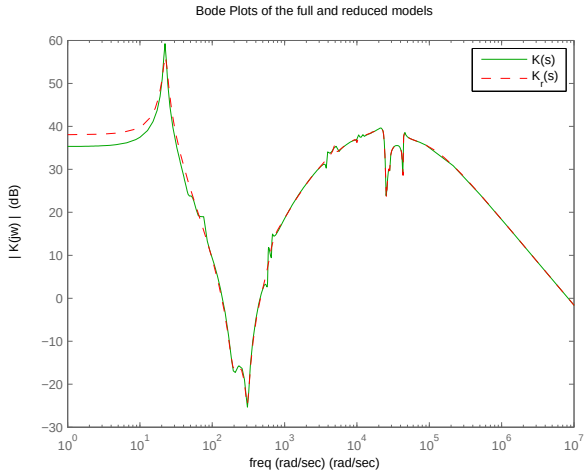

Figure 4.6: Reduction to order 19 with ϑ=7, ρ=12

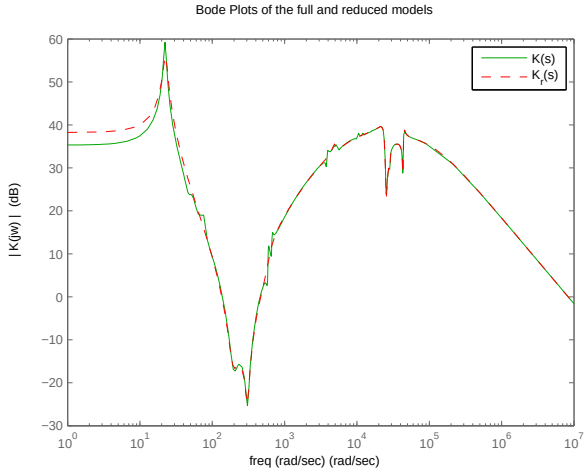

Figure 4.7: Reduction to order 23 with ϑ=9, ρ=14

Comparison to Weighted Balanced Truncation

Table 4.14 presents the weighted-$\mathcal{H}_2$ errors resulting from weighted balanced truncation (WBT) and W-IRKA.

reduction order r	10	12	15	20	23
WBT $\mathcal{H}_2$ error	63.49	62.810	38.635	3.9549	4.0486
best W-IRKA error	6.2738	5.0684	3.8286	4.2126	3.6948

Table 4.14: Comparison W-IRKA ($\mathcal{H}_2$) to WBT ($\mathcal{H}_2$), n=120

As we can see, W-IRKA outperforms weighted balanced truncation in most cases. Table 4.15 presents the weighted-$\mathcal{H}_\infty$ errors resulting from W-IRKA and weighted balanced truncation (WBT). We see that even in this case W-IRKA yields better results for many r.

reduction order r	10	12	15	20	23
WBT $\mathcal{H}_\infty$ error	21.074	12.686	4.019	0.1211	0.249
best W-IRKA error	0.8862	0.8843	0.6296	0.8622	0.6261

Table 4.15: Comparison W-IRKA ($\mathcal{H}_\infty$) to WBT ($\mathcal{H}_\infty$), n=120

Chapter 5

Final conclusions

We derived a new formula (3.4.2) for the weighted-$\mathcal{H}_2$ error $J_w = \|(G - G_r) \cdot W\|^2_{\mathcal{H}_2}$ and presented new weighted-$\mathcal{H}_2$ optimality conditions (3.5.4). In Algorithm 3.4.1 we presented a way of solving for the optimal weighted-$\mathcal{H}_2$ necessary conditions numerically. Due to the fact that this is not a numerically effective way of solving the problem, searching for a more effective way of solving the problem might be a good direction for future works.

The new formula (3.4.2) for the weighted-$\mathcal{H}_2$ error and Algorithm 2.2.1 (IRKA), which was presented in [26], were the starting points for Algorithm 4.2.1 (W-IRKA). W-IRKA constructs a reduced order system G_r which interpolates G at the mirror images of the ϑ dominant poles of G, at the mirror images of the ρ dominant poles of W and at the mirror images of the r poles of the reduced system G_r ($r = \vartheta + \rho$). W-IRKA is more computationally effective than Halevi's Algorithm [27] and weighted balanced truncation [5, 22]. In Chapter 4 we illustrated the performance of W-IRKA. We compared the $\mathcal{H}_2/\mathcal{H}_\infty$ errors obtained with W-IRKA to the $\mathcal{H}_2/\mathcal{H}_\infty$ errors obtained with weighted balanced truncation. We observed that, for virtually all reduction orders r, the $\mathcal{H}_2$ error obtained with W-IRKA was smaller

than the $\mathcal{H}_2$ error obtained with weighted balanced truncation. Moreover we observed that the $\mathcal{H}_\infty$ error obtained with W-IRKA for many r was smaller than the $\mathcal{H}_\infty$ error obtained with weighted balanced truncation.

However, we also observed that there might be better initialization strategies than the one we chose with ϑ dominant poles of the original system G and ρ dominant poles of the weight W as initialization. Finding a better initialization strategy for W-IRKA would be a possible direction for future works on the topic.

Moreover, the weighted-$\mathcal{H}_2$ optimality conditions (3.5.4) we derived in Chapter 3 can serve as a starting point for future works on finding new interpolation conditions for weighted-$\mathcal{H}_2$ optimality.

Appendix A

MATLAB CODE

Matlab code for W-IRKA:

```
clear

% load the full order system G and the weight W

[A,B,C,D] = ssdata(G);
[Aw,Bw,Cw,Dw]=ssdata(W);

% compute the highest residues with the corresponding poles of G and
% the highest residues with the corresponding poles of W by using the
% dominant pole algorithm , then choose the number x of cancelled
% residues (belonging to G) and the number y of cancelled residues
% (belonging to W)
```

```
x = ;

y = ;


% store the x+y dominant poles of G and W in Ls


Ls = ;


A = sparse(A);

B = sparse(B);

C = sparse(C);


SGM = zeros(x+y,1);


% initialization vector for IRKA


for i = 1:x+y


    SGM(i) = -Ls(i);


end


S = [SGM]


E = speye(size(A));
```

```
% sort the real and the complex numbers and construct Vr and Wr
% as  Vr = [Vre Vim] and Wr = [Wre Wim], where Vre, Vim, Wre, Wim
% are constructed as described

Wre = ;
Vre = ;

% Now costruct the reduced Ar,Br,Cr

   Ar = (Wr'*Vr)\(Wr'*A*Vr);
   Br = (Wr'*Vr)\(Wr'*B);

   Cr = (C*Vr)';

% choose tolerance

epsilon = ;
relchange = ;

% assign SGM the mirror images of the eigenvalues of Ar

SGM = -eig(Ar);

% force stability

unss = find(real(SGM)<0);
```

```
SGM(unss) = -SGM(unss);


S = [S SGM];


iterations = 0;


G_old = ss(Ar, Br, Cr', 0);


H2unw =  norm(G-G_old,2);
H2w  = norm((G-G_old)*W,2);


while epsilon<relchange,



    % sort the real and the complex numbers and construct Vr and Wr
    % as  Vr = [Vre Vim] and Wr = [Wre Wim], where Vre, Vim, Wre, Wim
    % are constructed as described


    Vr = ;
    Wr = ;


    % Construct the new reduced Ar,Br,Cr


    Ar = (Wr'*Vr)\(Wr'*A*Vr);
    Br = (Wr'*Vr)\(Wr'*B);
    Crt = C*Vr;
```

```
    G_new = ss(Ar, Br, Crt, D);
    G_err = G_old-G_new;

    G_old = G_new;
    SGM = -eig(Ar);
    S = [S SGM];
    S = sort(S);
    l = size(S,2);
    relchange = norm(S(:,l)-S(:,l-1))/norm(S(:,l));
    relchange
    iterations = iterations+1
    H2unw =  [H2unw norm(G-G_old,2)];
    H2w  = [H2w norm((G-G_old)*W,2)];


end

S

H2unw

H2w
```

Bibliography

[1] B.D.O. Anderson and Y. Liu, *Controller Reduction: Concept and Approaches*, IEEE Transactions on Automatic Control, AC-34, pp. 802-812, 1989

[2] A.C. Antoulas, *On recursiveness and related topics in linear systems*, IEEE Trans. Automatic Control, **31**, pp. 1121–1135 (1986).

[3] A.C. Antoulas and J.C. Willems, *A behavioral approach to linear exact modeling*, *IEEE Trans. Automatic Control*, **38**, pp. 1776–1802 (1993).

[4] A.C. Antoulas, *Recursive modeling of discrete-time time series*, in IMA volume on Linear Algebra for Control, P. Van Dooren and B.W. Wyman Editors, Springer Verlag, **62**, pp. 1–20 (1993).

[5] A.C. Antoulas, *Approximation of large-scale dynamical systems*, Advances in Design and Control DC-06, SIAM, Philadelphia (2005).

[6] A. C. Antoulas, D.C. Sorensen, and S. Gugercin, *A survey of model reduction methods for large scale systems*, Contemporary Mathematics, AMS Publications, **280**, pp. 193–219 (2001).

[7] J. Badia, P. Benner, R. Mayo, E. S. Quintana-Orti, G. Quintana-Orti, and J. Saak, *Parallel Order Reduction via Balanced Truncation for Optimal Cooling of Steel Profiles.*

In Euro-Par 2005 Parallel Processing; Lisbon. (eds: Cunha, Mdeiros), Springer **L**CNS 3648, pp. 857–866 (2005)

[8] L. Baratchart, M. Cardelli and M. Olivi, *Identification and rational ℓ_2 approximation: a gradient algorithm*, Automatica, **27**, pp. 413–418 (1991).

[9] P. Benner, *Solving Large-Scale Control Problems*, IEEE Control Systems Magazine, **24** (1), pp. 44–59 (2004).

[10] P. Benner and J. Saak, *Efficient numerical solution of the LQR-problem for the heat equation*, Proc. Appl. Math. Mech., **4**(1), pp. 648–649 (2004).

[11] A.E. Bryson and A. Carrier, *Second-order algorithm for optimal model order reduction*, J. Guidance Contr. Dynam., pp. 887–892 (1990).

[12] A. Bunse-Gerstner, D. Kubalinska, G. Vossen and D. Wilczek, *H_2-optimal model reduction for large scale discrete dynamical MIMO systems*, University of Bremen, ZeTeM Technical Report 07-04 (2007). (available at http://www.math.uni-bremen.de/zetem/reports/reports-liste.html#reports2007)

[13] John B. Conway, *Functions of One Complex Variable* , Springer, 2nd edition (1978)

[14] C. De Villemagne and R. Skelton, *Model reduction using a projection formulation*, International Journal of Control, **40**, pp. 2141–2169 (1987).

[15] P. Feldman and R.W. Freund, *Efficient linear circuit analysis by Padé approximation via a Lanczos method*, IEEE Trans. Computer-Aided Design, **14**, pp. 639-649 (1995).

[16] P. Fulcheri and M. Olivi, *Matrix rational $\mathcal{H}_2$ approximation: a gradient algorithm based on Schur analysis*, SIAM Journal on Control and Optimization, **36**, pp. 2103–2127 (1998).

[17] D. Gaier, *Lectures on Complex Approximation*, Birkhäuser (1987).

[18] K. Gallivan, E. Grimme, and P. Van Dooren, *A rational Lanczos algorithm for model reduction*, Numerical Algorithms, **2**(1–2), pp. 33–63 (1996).

[19] K. Gallivan, P. Van Dooren, and E. Grimme, *On some recent developments in projection-based model reduction*, in ENUMATH 97 (Heidelberg), World Sci. Publishing, River Edge, NJ, 1998, pp. 98–113 (1998).

[20] E.J. Grimme, *Krylov Projection Methods for Model Reduction*, Ph.D. Thesis, ECE Dept., University of Illinois, Urbana-Champaign (1997).

[21] S. Gugercin, A.C. Antoulas, and C.A. Beattie, *Krylov-based controller reduction for large-scale dynamical systems*

[22] S. Gugercin and A.C. Antoulas, *A survey of model reduction by balanced truncation and some new results*, International Journal of Control, Vol: 77 No: 8, pp. 748-766 (2004).

[23] S. Gugercin and A.C. Antoulas, *A comparative study of 7 model reduction algorithms*, Proc. 39th IEEE Conf. on Decision and Control, Sydney, pp. 2367–2372 (2000).

[24] S. Gugercin, *Projection methods for model reduction of large-scale dynamical systems*, Ph.D. Thesis, ECE Dept., Rice University (2002).

[25] S. Gugercin and A.C. Antoulas, *An $\mathcal{H}_2$ error expression for the Lanczos procedure*, Proc. 42nd IEEE Conf. on Decision and Control, **2**, pp. 1869–1872, (2003).

[26] S. Gugercin, A.C. Antoulas, and C.A. Beattie, *$\mathcal{H}_2$ Model Reduction for Large-Scale Linear Dynamical Systems*, SIAM Journal on Matrix Analysis and Applications, Vol. 30, Issue: 2, pp. 609-638, (2008).

[27] Y. Halevi, Frequency weighted model reduction via optimal projection, in Proc. IEEE Conf. Decision and Control, pp. 2906–2911 (1990).

[28] D. Higham and N. Higham, *Structured backward error and condition of generalized eigenvalue problems*, SIAM Journal on Matrix Analysis and Applications, **20**, pp. 493–512 (1998).

[29] D.C. Hyland and D.S. Bernstein, *The optimal projection equations for model reduction and the relationships among the methods of Wilson, Skelton, and Moore*, IEEE Trans. Automatic Control, **30**, pp. 1201–1211 (1985).

[30] Werner A. Kohler and Lee W. Johnson, *Elementary Differential Equations with Boundary Value Problems*, Addison-Wesley; 1st edition (2004)

[31] J.G. Korvink and E.B. Rudyni, *Oberwolfach Benchmark Collection*, In P. Benner, G. Golub, V. Mehrmann, and D. Sorensen, editors, Dimension Reduction of Large-Scale Systems, Lecture Notes in Computational Science and Engineering, **45**, Springer (2005).

[32] D. Kubalinska, A. Bunse-Gerstner, G. Vossen and D. Wilczek, *H_2-optimal interpolation based model reduction for large-scale systems*, Proc. of the 16th ICSS, September 4–6, 2007, Wroclaw, Poland.

[33] A. Lepschy, G.A. Mian, G. Pinato and U. Viaro, *Rational L_2 approximation: A non-gradient algorithm*, Proc. 30^{th} IEEE Conf. Decision and Control, pp. 2321–2323 (1991).

[34] L. Meier and D.G. Luenberger, *Approximation of Linear Constant Systems*, IEEE Trans. Automatic Control, **12**, pp. 585–588 (1967).

[35] B. C. Moore, *Principal Component Analysis in Linear System: Controllability, Observability and Model Reduction*, IEEE Trans. Automatic Control, **26**, pp. 17–32 (1981).

[36] C. T. Mullis and R. A. Roberts, *Synthesis of minimum roundoff noise fixed point digital filters*, IEEE Trans. on Circuits and Systems, **CAS-23**, pp. 551-562 (1976).

[37] T. Penzl, *Algorithms for model reduction of large dynamical systems*, Linear Algebra and its Applications, **4**15 (2–3), pp. 322–343 (2006).

[38] L.T. Pillage and R.A. Rohrer, *Asymptotic waveform evaluation for timing analysis*, IEEE Trans. Computer-Aided Design, **9**, pp. 352-366 (1990).

[39] V. Raghavan, R.A. Rohrer, L.T. Pillage, J.Y. Lee, J.E. Bracken, and M.M. Alaybeyi, *AWE Inspired*, Proc. IEEE Custom Integrated Circuits Conference (1993).

[40] F. Riesz and Sz.-Nagy, B., *Functional Analysis*, Ungar, New York. (1955).

[41] J. Rommes and N. Martins, *Efficient computation of transfer function dominant poles using subspace acceleration*, IEEE TPWRS 21, 3 (Aug. 2006), pp.1218-1226.

[42] A. Ruhe, *Rational Krylov algorithms for nonsymmetric eigenvalue problems II: matrix pairs*, Linear Algebra and its Applications, **197**, pp. 283-295 (1994).

[43] G. Schelfhout and B. De Moor, *A note on closed-loop balanced truncation*, IEEE Transactions on Automatic Control, AC-41, pp. 1498-1500, 1996

[44] J.T. Spanos, M.H. Milman, and D.L. Mingori, *A new algorithm for $\mathcal{L}_2$ optimal model reduction*, Automatica, **28**, pp. 897-909 (1992).

[45] P. Van Dooren, K. A. Gallivan and P.-A. Absil, *H_2 optimal model reduction of MIMO systems*, Applied Mathematics Letters (2008). In press, doi:10.1016/j.aml.2007.09.015.

[46] A. Varga and B.D.O. Anderson, *Frequency-weighted balancing related controller reduction*, Proceedings of the IFAC'2002 congress, Barcelona, Spain, 2002

[47] D.A. Wilson, *Optimum solution of model reduction problem*, in Proc. Inst. Elec. Eng., pp. 1161–1165 (1970).

[48] W-Y. Yan and J. Lam, *An approximate approach to $\mathcal{H}_2$ optimal model reduction*, IEEE Trans. Automatic Control, **44**, pp. 1341–1358 (1999).

[49] A. Yousouff and R.E. Skelton, *Covariance equivalent realizations with applications to model reduction of large-scale systems*, in Control and Dynamic Systems, C.T. Leondes ed., Academic Press, **22**, pp. 273-348 (1985).

[50] A. Yousouff, D. A. Wagie, and R. E. Skelton, *Linear system approximation via covariance equivalent realizations*, Journal of Math. Anal. and App., **196**, pp. 91–115 (1985).

[51] K. Zhou with J.C. Doyle and K. Glover, *Robust and optimal control*, Prentice Hall, 1996

[52] D. Zigic, L. Watson, and C. Beattie, *Contragredient transformations applied to the optimal projection equations.* Linear Algebra and its Applications, **188/189**, pp. 665–676 (1993).