Iterative Rational Krylov Algorithm for Unstable Dynamical Systems and Generalized Coprime Factorizations

Klajdi Sinani

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

Serkan Gugercin, Chair
Christopher Beattie
Mark Embree

December 18, 2015
Blacksburg, Virginia

Keywords: Model Reduction, Dynamical Systems, IRKA, Unstable Systems,
Structure-preserving algorithms

Iterative Rational Krylov Algorithm for Unstable Dynamical Systems and Generalized Coprime Factorizations

Klajdi Sinani

Abstract

Generally, large-scale dynamical systems pose tremendous computational difficulties when applied in numerical simulations. In order to overcome these challenges we use several model reduction techniques. For stable linear models these techniques work very well and provide good approximations for the full model. However, large-scale unstable systems arise in many applications. Many of the known model reduction methods are not very robust, or in some cases, may not even work if we are dealing with unstable systems. When approximating an unstable sytem by a reduced order model, accuracy is not the only concern. We also need to consider the structure of the reduced order model. Often, it is important that the number of unstable poles in the reduced system is the same as the number of unstable poles in the original system. The Iterative Rational Krylov Algorithm (IRKA) is a robust model reduction technique which is used to locally reduce stable linear dynamical systems optimally in the $\mathcal{H}_2$-norm. While we cannot guarantee that IRKA reduces an unstable model optimally, there are no numerical obstacles to the reduction of an unstable model via IRKA. In this thesis, we investigate IRKA's behavior when it is used to reduce unstable models. We also consider systems for which we cannot obtain a first order realization of the transfer function. We can use Realization-independent IRKA to obtain a reduced order model which does not preserve the structure of the original model. In this paper, we implement a structure preserving algorithm for systems with nonlinear frequency dependency.

Contents

1 Introduction **1**

2 Literature Review **2**

2.1 Model Reduction of Linear Dynamical Systems 2

2.2 Error Measures 3

2.3 Interpolatory Model Reduction 5

2.3.1 Interpolation Methods 7

2.4 Balanced Truncation 9

3 Model Reduction for Unstable Systems **13**

3.1 Optimal $\mathcal{L}_2$ Model Reduction 13

3.2 Balanced Truncation for Unstable System 18

4 Model Reduction of Unstable Systems with IRKA **20**

4.1 Pole Capturing by IRKA 21

4.2 Comparing IRKA for Unstable Systems with Other Model Reduction Techniques 31

4.3 Different Initalizations for Different Model Reduction Techniques 40

4.4 Shifted IRKA 42

5 A Structure Preserving Algorithm for Dynamical Systems with Nonlinear Frequency Dependency **45**

5.1 Loewner Matrix Approach for Interpolation and Realization-independent IRKA 46

5.2 A Structure-preserving interpolation based algorithm 48

5.3 Numerical examples: Reducing symmetric models via Structure-preserving TF-IRKA 49

5.3.1 Beam Model 49

5.3.2 Hadeler Model 51

5.3.3 Loaded String Model 53

6 Conclusions and Future Work **55**

References **56**

List of Figures

1 Bode Plots for FOM and ROM obtained via IRKA ($r = 12$) 32

2 Bode Plots for FOM and ROM obtained via $\mathcal{L}_2$ IRKA ($r = 12$) 34

3 Bode Plots for FOM and ROM obtained via balanced truncation ($r = 12$) . . 35

4 Bode Plots for FOM with equal number of stable and unstable poles, and ROM obtained via IRKA ($r = 12$) 38

5 Bode Plots for FOM with equal number of stable and unstable poles, and ROM obtained via $\mathcal{L}_2$IRKA ($r = 12$) 39

6 Bode Plots for FOM with equal number of stable and unstable poles, and ROM obtained via balanced truncation ($r = 12$) 40

7 $\mathcal{H}_\infty$ error as the shift for Shifted IRKA varies 44

8 $\mathcal{H}_2$ error as the shift for Shifted IRKA varies 45

9 Beam model reduction with Structure-preserving TF-IRKA ($r = 22$) 50

10 Beam model reduction with TF-IRKA ($r = 22$) 51

11 Hadeler model reduction with Structure-preserving TF-IRKA ($r = 12$) . . . 52

12 Hadeler model reduction with TF-IRKA ($r = 12$) 53

13 Loaded string model reduction with Structure-preserving TF-IRKA ($r = 10$) 54

List of Tables

1 FOM (2 unstable poles) vs. ROM poles ($r = 2$) 21

2 FOM (2 unstable poles) vs. ROM poles ($r = 3$) 21

3 FOM (2 unstable poles) vs. ROM poles ($r = 4$) 22

4 FOM (2 unstable poles) vs. ROM poles ($r = 5$) 22

5 FOM (3 unstable poles) vs. ROM poles ($r = 2$) 23

6 FOM (3 unstable poles) vs. ROM poles ($r = 3$) 23

7 FOM (3 unstable poles) vs. ROM poles ($r = 4$) 23

8 FOM (3 unstable poles) vs. ROM poles ($r = 5$) 24

9 FOM (3 unstable poles) vs. ROM poles ($r = 6$) 24

10 FOM (3 unstable poles) vs. ROM poles ($r = 7$) 25

11 FOM (3 unstable poles) vs. ROM poles ($r = 8$) 25

12 FOM (4 unstable poles) vs. ROM poles ($r = 2$) 26

13 FOM (4 unstable poles) vs. ROM poles ($r = 3$) 27

14 FOM (4 unstable poles) vs. ROM poles ($r = 4$) 27

15 FOM (4 unstable poles) vs. ROM poles ($r = 5$) 27

16 FOM (4 unstable poles) vs. ROM poles ($r = 6$) 28

17 FOM (4 unstable poles) vs. ROM poles ($r = 7$) 28

18 FOM (4 unstable poles) vs. ROM poles ($r = 8$) 29

19 FOM (4 unstable poles) vs. ROM poles ($r = 9$) 29

20 FOM (60 stable +60 unstable poles) vs. ROM poles ($r = 12$) 31

21 IRKAfUS error as the no. of unstable poles varies ($r = 12$) 32

22 $\mathcal{L}_2$ IRKA error as the no. of unstable poles varies ($r = 12$) 33

23 Balanced truncation error as the no. of unstable poles varies ($r = 12$) 35

24 IRKAfUS error as the no. of unstable poles varies ($r = 10$) 36

25 $\mathcal{L}_2$ IRKA error as the no. of unstable poles varies ($r = 10$) 36

26 Balanced truncation error as the no. of unstable poles varies ($r = 10$) 36

27 Comparison for FOM with equal number of stable and unstable poles ($r = 8$) 37

28 Comparison for FOM with equal number of stable and unstable poles ($r = 10$) 37

29 Comparison for FOM with equal number of stable and unstable poles ($r = 12$) 37

30 Comparison of different techniques with different initializations($r = 12$) . . . 41

31 Comparison of different techniques with different initializations($r = 10$) . . . 41

32 Comparison of different techniques with different initializations($r = 8$) 42

33 $\mathcal{L}_2$ and $\mathcal{L}_\infty$error of the approximation by Shifted IRKA 42

34 Comparison of Shifted IRKA, $\mathcal{L}_2$ IRKA, and IRKA for unstable systems ($r =$ 8) 43

35 Comparison of Shifted IRKA, $\mathcal{L}_2$ IRKA, and IRKA for unstable systems ($r =$ 10) 43

36 Comparison of Shifted IRKA, $\mathcal{L}_2$ IRKA, and IRKA for unstable systems($r = 12$) 43

37 Structure-preserving TF-IRKA vs. TF-IRKA for Beam model 50

38 Structure-preserving TF-IRKA vs. TF-IRKA for Hadeler model 52

39 Structure-preserving TF-IRKA for loaded string model 54

1 Introduction

Generally, large-scale dynamical systems pose tremendous computational difficulties when applied in numerical simultaions. In order to overcome these challenges we use several techniques such as Balanced Truncation [9, 10], Proper Orthogonal Decomposition [13], Interpolatory Methods [1, 14], and Hankel Norm Approximation [11]. For stable linear models these techniques work very well and provide good approximations for the full model. However, in many applications, we encounter large-scale unstable linear models. An example of such application is plant controller reduction [2, 16, 17, 15]. Often, we are interested in reducing the controller, which is an unstable model. In this case, we are not only interested in the accuracy of the approximation, but also in preserving the unstable structure of the system i.e. we aim to obtain a reduced order model, which has the same number of unstable poles as the original model. Some of the known model reduction methodologies are not so robust, or in some cases, may not even work if we are dealing with unstable systems. While we cannot guarantee that IRKA reduces an unstable model optimally, there are no numerical obstacles to the reduction of an unstable model via IRKA. In this thesis, we invetstigate the obtained reduced order models when unstable systems are reduced with IRKA. We also consider dynamical systems with nonlinear frequency dependency which do not have a first-order realization for the transfer function. We introduce a structure-preserving algorithm based on Realization-independent IRKA to reduce susch systems. In Chapter 2 we discuss the model reduction problem, the error measures $\mathcal{H}_2$ and $\mathcal{H}_\infty$, and model reduction techniques for stable systems such as IRKA and Balanced Truncation. In Chapter 3.6 we dicuss approximation methodologies for unstable systems such as $\mathcal{L}_2$ IRKA and Balanced truncation for unstable systems. In Chapter 4, we present results obtained by reducing unstable systems via IRKA, and compare IRKA with other model reduction techniques for unstable systems. In Chapter 5 we introduce a structure-preserving algorithm

and present a few numerical examples to illustrate the performance of this algorithm and compare it with Realization-independent IRKA. In the final chapter, we draw our conclusions based on the attained numerical results.

2 Literature Review

2.1 Model Reduction of Linear Dynamical Systems

Consider the linear dynamical system:

$$\begin{aligned} \mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad \text{with } \mathbf{x}(0) = \mathbf{0} \end{aligned} \tag{2.1}$$

where $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$, and $\mathbf{D} \in \mathbb{R}^{p\times m}$ are constant matrices. In this system $\mathbf{x}(t) \in \mathbb{R}^n$ is the internal variable, also called the state variable if the matrix $\mathbf{E}$ is invertible. The dimensiion of the system is n. $\mathbf{u}(t)$ is the input, while $\mathbf{y}(t)$ is the output. If $m = p = 1$, then the dynamical system is called single-input/single-output (SISO). If $m \neq 1$ and $p \neq 1$, the system is called multi-input/multi-output (MIMO).

When n is very large e.g. $n > 10^6$, the simulation is very costly computationally speaking. The purpose of model reduction is to replace the original model with a lower dimension model that has the form

$$\begin{aligned} \mathbf{E}_r\dot{\mathbf{x}}_r(t) &= \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{B}_r\mathbf{u}(t) \\ \mathbf{y}_r(t) &= \mathbf{C}_r\mathbf{x}_r(t) + \mathbf{D}_r\mathbf{u}(t) \quad \text{with } \mathbf{x}_r(0) = \mathbf{0} \end{aligned} \tag{2.2}$$

where $\mathbf{A}_r, \mathbf{E}_r \in \mathbb{R}^{r\times r}$, $\mathbf{B}_r \in \mathbb{R}^{r\times m}$, $\mathbf{C}_r \in \mathbb{R}^{p\times r}$, and $\mathbf{D}_r \in \mathbb{R}^{p\times m}$ with $r \ll n$, and such that the outputs of the reduced system are good approximations of the corresponding true outputs

over a wide range of outputs. In order to quantify the model reduction error, we use the frequency domain representation.

Let $\hat{\mathbf{y}}(s)$, $\hat{\mathbf{y}}_r(s)$, and $\hat{\mathbf{u}}(s)$ be the Laplace transforms of $\mathbf{y}(t)$, $\mathbf{y}_r(t)$ and $\mathbf{u}(t)$. After taking the Laplace transforms of the original model (2.1) and the reduced model (2.2) we obtain

$$\hat{\mathbf{y}}(s) = (\mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D})\hat{\mathbf{u}}(s)$$
$$\hat{\mathbf{y}}_r(s) = (\mathbf{C}_r(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{B}_r + \mathbf{D}_r)\hat{\mathbf{u}}(s).$$

The mappings $\hat{\mathbf{u}} \mapsto \hat{\mathbf{y}}$ and $\hat{\mathbf{u}} \mapsto \hat{\mathbf{y}}_r$ are called transfer functions and they are denoted:

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$
$$\mathbf{H}_r(s) = \mathbf{C}_r(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{B}_r + \mathbf{D}_r$$

There are robust methods such as the Iterative Rational Krylov Algorithm (IRKA) [1, 14], Balanced Truncation [9, 10], Hankel Norm Approximation [11] etc. which we can use to approximate these large scale systems by a reduced order model. The main methods that we examine in this paper are Iterative Rational Krylov Algorithm (IRKA) and Balanced Truncation

2.2 Error Measures

When we approximate a large scale dynamical system by a reduced order model, we need to compute the approximation error. Thus, we need appropriate error measures. The error analysis for linear dynamical systems will be conducted in the frequency domain, although the difference between the outputs $\hat{\mathbf{y}}(s)$ and $\hat{\mathbf{y}}_r(s)$ is directly linked to the difference between the full and reduced transfer functions. So we measure how close $\mathbf{H}_r(s)$ is to $\mathbf{H}(s)$ using the

$\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms.

The $\mathcal{H}_2$ norm is defined as

$$\|\mathbf{H}(s)\|_{\mathcal{H}_2} := \left(\frac{1}{2\pi}\int_{-\infty}^{\infty} \|\mathbf{H}(i\omega)\|_F^2\right)^{1/2}$$

where $\|\cdot\|_F$ represents the Frobenius norm. The $\mathcal{H}_2$ norm relates to the L_∞ norm of $\mathbf{y}(t)$ in the time domain in the following manner:

$$\|\mathbf{y}\|_{L_\infty} = \sup_{t>0} \|\mathbf{y}(t)\|_\infty \leq \|\mathbf{H}\|_{\mathcal{H}_2} \|\mathbf{u}\|_{L_2}.$$

In model reduction, we aim to minimize the error between the full and the reduced problem so we are interested in the following:

$$\|\mathbf{y} - \mathbf{y}_r\|_{L_\infty} \leq \|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{H}_2} \|\mathbf{u}\|_{L_2}.$$

As mentioned above, in addition to the $\mathcal{H}_2$ norm, we use the $\mathcal{H}_\infty$ norm to measure the error. The $\mathcal{H}_\infty$ norm is defined as:

$$\|\mathbf{H}\|_{\mathcal{H}_\infty} = \sup_{\omega\in\mathbb{R}} \|\mathbf{H}(i\omega)\|_2$$

where $\|\cdot\|_2$ denotes the 2-norm of a matrix. The $\mathcal{H}_\infty$ norm of a dynamical system is directly related to the L_2 induced operator norm of the operator which maps $\mathbf{u}$ into $\mathbf{y}$. We have,

$$\|\mathbf{H}\|_{\mathcal{H}_\infty} = \sup_{\mathbf{u}\in L_2} \|\mathbf{y}\|_{L_2} = \sup_{\mathbf{u}\in L_2} \left(\int_0^\infty \|\mathbf{y}(t)\|_2^2\, dt\right)^{1/2} \text{ for all } \mathbf{u} \text{ such that } \|\mathbf{u}\|_{L_2} = 1.$$

Hence, for the model reduction problem we have,

$$\|\mathbf{y} - \mathbf{y}_r\|_{L_2} \leq \|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{H}_\infty} \|\mathbf{u}\|_{L_2} .$$

Therefore, if we want to minimze the output error in the L_∞ norm, an $\mathcal{H}_2$-based model reduction technique should be used. On the other hand, if we aim to have the minimal output error in the L_2 norm, we should use an $\mathcal{H}_\infty$-based model reduction technique.

2.3 Interpolatory Model Reduction

When we reduce the order of a dynamical system we are essentially approximating the full order system with a reduced order model. In this section, for the sake of simplicity, we assume the original model is single input/single output (SISO). Nevertheless, in this thesis all the results pertaining to SISO systems can be extended to the mullti input/ multi output (MIMO) case. Generally, we use interpolation to approximate complex functions. Recall:

$$\mathbf{H}(s) = \mathbf{c}^T(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{b} + \mathbf{D}$$

is a rational function of degree n and

$$\mathbf{H}_r(s) = \mathbf{c}_r^T(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{b}_r + \mathbf{D}_r$$

is a rational function of degree r. In order to interpolate we need a set of interpolation points $\{s_i\}_i^r \subset \mathbb{C}$ and construct $\mathbf{H}_r$ such that

$$\mathbf{H}_r(s_i) = \mathbf{H}(s_i),$$
$$\mathbf{H}_r'(s_i) = \mathbf{H}'(s_i) \text{ for } i = 1, 2, 3, ..., r.$$

Thus, $\mathbf{H}_r(s)$ is the reduced order model that interpolates the original model. But how do we construct $\mathbf{H}_r$? We achieve this by projection. Given $\{s_i\}_i^r$ we construct $\mathbf{W} \in \mathbb{R}^{n\times r}$ and $\mathbf{V} \in \mathbb{R}^{n\times r}$ in the following manner:

$$\mathbf{V} = [(s_1\mathbf{E} - \mathbf{A})^{-1}\mathbf{b} \cdots (s_r\mathbf{E} - \mathbf{A})^{-1}\mathbf{b}]$$

and

$$\mathbf{W}^T = \begin{bmatrix} \mathbf{c}^T(s_1\mathbf{E} - \mathbf{A})^{-1} \\ \vdots \\ \mathbf{c}^T(s_r\mathbf{E} - \mathbf{A})^{-1} \end{bmatrix}.$$

Using $\mathbf{E}_r = \mathbf{W}^T\mathbf{E}\mathbf{V}$, $\mathbf{A}_r = \mathbf{W}^T\mathbf{A}\mathbf{V}$, $\mathbf{c}_r = \mathbf{c}\mathbf{V}$, $\mathbf{b}_r = \mathbf{W}^T\mathbf{b}$ and $\mathbf{D}_r = \mathbf{D}$ we obtain a reduced order model $\mathbf{H}_r$ that satisfies the Hermite interpolation conditions, i.e.,

$$\mathbf{H}(s_i) = \mathbf{H}_r(s_i)$$
$$\mathbf{H}'(s_i) = \mathbf{H}'_r(s_i)$$

for $i = 1, 2, 3, ..., r$.

However, our goal is to construct an optimal reduced model with respect to some norm. We are interested in optimality in the $\mathcal{H}_2$ norm. In other words, if we have a full-order dynamical system $\mathbf{H}(s)$, we want to construct a reduced-order model $\mathbf{H}_r(s)$ such that

$$\|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{H}_2} \leq \left\|\mathbf{H} - \hat{\mathbf{H}}_r\right\|_{\mathcal{H}_2},$$

where $\hat{\mathbf{H}}_r$ is any dynamical system of dimension r.

2.3.1 Interpolation Methods

The previous section showed how to construct a reduced model that satisfies the interpolation conditions given a set of initial shifts. However, we have no information whether the obtained reduced-order model is optimal. In this section we will describe how to obtain optimality, at least locally, in the $\mathcal{H}_2$ norm. Recall the optimization problem we are considering: If $\mathbf{H}(s)$ is the transfer function for a large dynamical system, find a new reduced model $\mathbf{H}_r(s)$, which minimizes the $\mathcal{H}_2$ error. In other words, find $\mathbf{H}_r$ such that

$$\|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{H}_2} = \min_{dim(\hat{\mathbf{H}}_r)=r} \left\|\mathbf{H} - \hat{\mathbf{H}}_r\right\|_{\mathcal{H}_2}.$$

We suppose $\mathbf{E}$ is nonsingular, which means $\mathbf{H}(s) = \mathbf{c}^T(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{b} + \mathbf{D}$ is the trasnfer function corresponding to a dynamical system of ODEs. In order to bound the $\mathcal{H}_2$ error norm $\|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{H}_2}$, we have to assume $\mathbf{D} = \mathbf{D}_r$. For simplicity, let $\mathbf{D} = \mathbf{D}_r = \mathbf{0}$. We write the pole-residue expansion of $\mathbf{H}_r(s)$:

$$\mathbf{H}_r(s) = \sum_{i=1}^{r} \frac{\phi}{s - \lambda_i}$$

where λ_i, l_i, r_i are the poles of the reduced system and the corresponding left and right residue directions respectively. This pole-residue expansion can be computed easily through the general eigenvalue decomposition for $s\mathbf{E}_r - \mathbf{A}_r$. The eigenvalue decomposition is relatively cheap since the size of the reduced system is relatively small. The next theorem reveals the necessary conditions for $\mathcal{H}_2$ optimality.

Theorem 2.1. *Let $\mathbf{H}_r(s)$ be the best r^{th} order rational approximation of $\mathbf{H}$ with respect to*

the $\mathcal{H}_2$ norm. Then

$$\mathbf{H}(-\lambda_k) = \mathbf{H}_r(-\lambda_k),$$
$$\mathbf{H}'(-\lambda_k) = \mathbf{H}'_r(-\lambda_k)$$

for $k = 1, 2, ..., r$ where λ_k denotes the poles of the reduced system.

Thus, in order to satisfy the first-order conditions for optimality in the $\mathcal{H}_2$ norm, we need to interpolate at the mirror images of the poles of the reduced model. However, since we do not have any knowledge of the reduced system poles, the model reduction algorithm must be iterative. Building upon Theorem 2.1, the Iterative Rational Krylov Algorithm (IRKA) was developed [14]. IRKA produces a reduced model that satisfies the first-order optimality conditions in the $\mathcal{H}_2$ norm. Indeed, in the SISO case, it is guaranteed to yield at least a locally optimal reduced order model [22]. Picking a set of initial interpolation points, we can compute $\mathbf{V}$ and $\mathbf{W}$ as described above, and eventually we obtain a reduced order model. Then we compute the pole residue expansion of the reduced model. After computing the pole residue expansion, we use the mirror images of the poles of the reduced system as interpolation points and repeat the process to obtain another reduced model. We continue until the convergence condition is satisfied.

Next, we provide a sketch of IRKA.

Sketch of IRKA

- Pick an r-fold intial shift set selection that is closed under conjugation
- $\mathbf{V} = [(\sigma_1\mathbf{E} - \mathbf{A})^{-1})\mathbf{b}$ ... $(\sigma_r\mathbf{E} - \mathbf{A})^{-1})\mathbf{b}]$
- $\mathbf{W} = [(\sigma_1\mathbf{E} - \mathbf{A})^{-T}\mathbf{c}^T$ ... $[(\sigma_r\mathbf{E} - \mathbf{A})^{-T}\mathbf{c}^T]$

- while (not converged)
  - $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r,\ \mathbf{E}_r = \mathbf{W}_r^T\mathbf{E}\mathbf{V}_r,\ \mathbf{b}_r = \mathbf{W}_r^T\mathbf{b}$, and $\mathbf{c}_r = \mathbf{c}\mathbf{V}_r$
  - Compute a pole-residue expansion of $\mathbf{H}_r(s)$:

$$\mathbf{H}_r(s) = \mathbf{c}_r^T(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{b}_r = \sum_{i=1}^{r} \frac{\phi}{s - \lambda_i}$$

  - $\sigma_i \leftarrow -\lambda_i$, for $i = 1,\ \ldots\ , r$
  - $\mathbf{V} = [(\sigma_1\mathbf{E} - \mathbf{A})^{-1}\mathbf{b}\ \ldots\ (\sigma_r\mathbf{E} - \mathbf{A})^{-1}\mathbf{b}]$
  - $\mathbf{W} = [(\sigma_1\mathbf{E} - \mathbf{A})^{-T}\mathbf{c}^T\ \ldots\ [(\sigma_r\mathbf{E} - \mathbf{A})^{-T}\mathbf{c}^T]$
- $\mathbf{A}_r = \mathbf{W}^T\mathbf{A}\mathbf{V},\ \mathbf{E}_r = \mathbf{W}^T\mathbf{E}\mathbf{V}$, $\mathbf{B}_r = \mathbf{W}^T\mathbf{b}$, and $\mathbf{C}_r = \mathbf{c}\mathbf{V}$

For more details about IRKA check [1].

2.4 Balanced Truncation

Consider the following dynamical system:

$$\mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)$$

Then we write:

$$\boldsymbol{\Sigma} := \left[\begin{array}{c|c} (\mathbf{A}, \mathbf{E}) & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right]$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{n\times m}, \mathbf{C} \in \mathbb{R}^{p\times n},$ and $\mathbf{D} \in \mathbb{R}^{p\times m}$. Reducing the order of a dynamical system requires the elimination of some of the state variables. Model reduction via IRKA yields an optimal reduced order model in the $\mathcal{H}_2$ norm. On the other hand, balanced truncation is not optimal in any norm, but it has a very good error bound with respect to the $\mathcal{H}_\infty$ norm. Approximation by balanced truncation is achieved by eliminating the states that are hard to reach and hard to observe. In order to classify which states are hard to reach and to observe we use the reachability and observability gramians. We can compute these gramians by solving the following Lyapunov equations for $\mathcal{P}$ and $\mathcal{Q}$:

$$\mathbf{A}\mathcal{P}\mathbf{E}^T + \mathbf{E}\mathcal{P}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = 0, \qquad \mathbf{A}^T\mathcal{Q}\mathbf{E} + \mathbf{E}^T\mathcal{Q}\mathbf{A} + \mathbf{C}^T\mathbf{C} = 0 \tag{2.3}$$

where $\mathcal{P} \in \mathbb{R}^{n\times n}$ and $\mathcal{Q} \in \mathbb{R}^{n\times n}$ are the reachability and observability gramians respectively. The Lyapunov equations above can be solved using the Bartels-Stewart algorithm [19]. If $\mathbf{A}$ is asymptotically stable, the solutions $\mathcal{P}, \mathcal{Q}$ to the Lyapunov equations are symmetric positive semi-definite. The Bartels-Stewart algorithm computes a Schur decomposition, hence the number of arithmetic operations is $\mathcal{O}(n^3)$ and the storage required is $\mathcal{O}(n^2)$. Thus, balanced truncation is too expensive for large-scale systems if we solve the Lyapunove equations (2.3) with the Bartels-Stewart algorithm. We can reduce the cost of balanced truncation if we solve the Lyapunov equations using Krylov methods [20].

We mentioned above balanced truncation eliminates the states which are hard to reach and hard to observe. But what if the states which are difficult to reach are easy to observe or

vice-versa? Note we can transform the gramians using a nonsingular matrix T:

$$\hat{\mathcal{P}} = T\mathcal{P}T^T,\ \hat{\mathcal{Q}} = T^{-T}\mathcal{Q}T^{-1}.$$

In order to guarantee the states that are hard to reach are simultaneously hard to observe we need to find an invertible state transformation T that yields $\hat{\mathcal{P}} = \hat{\mathcal{Q}}$ in the transformed basis. If $\hat{\mathcal{P}} = \hat{\mathcal{Q}}$, we say that the reachable, observable and stable system $\mathbf{\Sigma}$ is balanced. $\mathbf{\Sigma}$ is principal axis balanced if $\hat{\mathcal{P}} = \hat{\mathcal{Q}} = \text{diag}(\sigma_1, ..., \sigma_n)$ for $\sigma_i = \sqrt{\lambda_i(\mathcal{P}\mathcal{Q})}$ where λ_i's denote the eigenvalues of $\mathcal{P}\mathcal{Q}$. The values σ_i are knows as the Hankel singular values of the system. Before computing the balancing transformation T we need the Cholesky factor $\mathbf{U}$ of $\mathcal{P}$ and the eigendecomposition of $\mathbf{U}^T\mathcal{Q}\mathbf{U}$:

$$\mathcal{P} = \mathbf{U}\mathbf{U}^T, \quad \mathbf{U}^T\mathcal{Q}\mathbf{U} = \mathbf{K}\mathbf{G}^2\mathbf{K}^T$$

Lemma 2.1. ***Balancing transformation***. *Given the reachable, observable and stable system*

$$\left[\begin{array}{c|c} (\mathbf{A}, \mathbf{E}) & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right]$$

and the corresponding gramians $\mathcal{P}$ and $\mathcal{Q}$ a principal axis balancing transformation is given as follows:

$$T = \mathbf{G}^{1/2}\mathbf{K}^T\mathbf{U}^{-1} \text{ and } T^{-1} = \mathbf{U}\mathbf{K}\mathbf{G}^{-1/2}.$$

In the balanced basis the states which are hard to observe are also hard to reach and they correspond to small Hankel singular values. Let's assume the system

$$\left[\begin{array}{c|c} (\hat{\mathbf{A}}, \hat{\mathbf{E}}) & \hat{\mathbf{B}} \\ \hline \hat{\mathbf{C}} & \hat{\mathbf{D}} \end{array}\right]$$

is balanced. Consider the following matrix partitions:

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}, \hat{\mathbf{E}} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}, \hat{\mathbf{B}} = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix}, \hat{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_1 \, \mathbf{C}_2 \end{bmatrix}, \mathbf{G} = \begin{bmatrix} \mathbf{G}_{11} & 0 \\ 0 & \mathbf{G}_{22} \end{bmatrix}$$

for $\mathbf{G} = \text{diag}(\sigma_1, ..., \sigma_n) = \mathcal{P} = \mathcal{Q}$, $\mathbf{G}_1 = (\sigma_1, ..., \sigma_r), \mathbf{G}_2 = (\sigma_{r+1}, ..., \sigma_N)$, where σ_i denotes the i-th Hankel singular values of the system for $i = 1, 2, ..., n$. Note $\mathbf{A}_{11} \in \mathbb{R}^{r\times r}, \mathbf{E}_{11} \in \mathbb{R}^{r\times r}, \mathbf{G}_{11} \in \mathbb{R}^{r\times r}, \mathbf{B}_1 \in \mathbb{R}^{r\times m}$, and $\mathbf{C} \in \mathbb{R}^{p\times r}$. Then the system

$$\boldsymbol{\Sigma}_r := \left[\begin{array}{c|c} (\mathbf{A}_{11}, \mathbf{E}_{11}) & \mathbf{B}_1 \\ \hline \mathbf{C}_1 & \mathbf{D} \end{array}\right]$$

is a reduced order model obtained by balanced truncation. This model is asymptotically stable, minimal and satisfies

$$\|\boldsymbol{\Sigma} - \boldsymbol{\Sigma}_r\|_{\mathcal{H}_\infty} \leq 2(\sigma_{r+1} + ... + \sigma_n).$$

The equality is achieved if $\mathbf{G}_{22} = \sigma_n$[10]. The balancing method described above is numerically inefficient and ill-conditioned. For this reason, when we implement balanced truncation, square-root balancing is preferred. Next, we provide a description of the square-root balancing method. First, we Compute the Cholesky factorizations of the Gramians:

$$\mathcal{P} = \mathbf{U}\mathbf{U}^T, \quad \mathcal{Q} = \mathbf{L}\mathbf{L}^T$$

Then, we compute the singular value decomposition $\mathbf{U}^T\mathbf{E}\mathbf{L} = \mathbf{W}\Sigma\mathbf{V}^T$. Let $\mathbf{W}_r, \mathbf{V}_r$ be the matrices consisting of the first r columns of $\mathbf{W}, \mathbf{V}$, respectively, and Σ_r be the diagonal matrix with the largest r singular values on the diagonal. If we define $\mathbf{Z}_r = \mathbf{U}\mathbf{W}_r\Sigma_r^{-1/2}$ and $\mathbf{Y}_r = \mathbf{L}\mathbf{V}_r\Sigma_r^{-1/2}$, we can compute $\mathbf{E}_r = \mathbf{Y}_r^T\mathbf{E}\mathbf{Z}_r$, $\mathbf{A}_r = \mathbf{Y}_r^T\mathbf{A}\mathbf{Z}_r$, $\mathbf{B}_r = \mathbf{Y}_r^T\mathbf{B}$, $\mathbf{C}_r = \mathbf{C}\mathbf{Z}_r$; hence,

obtain the reduced model.

3 Model Reduction for Unstable Systems

The algorithms we have discussed so far, IRKA and Balanced Truncation, are used to reduce linear stable systems. However, unstable systems arise in many applications. Consider the linear dynamical system:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) \end{aligned} \tag{3.1}$$

where $\mathbf{A}, \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$ are constant matrices with $p = m = 1$. Assume this system has poles that lie to the right of the imaginary axis i.e., the system is unstable. Throughout this section we assume there are no poles on the imaginary axis. Our goal is to reduce the system and obtain a new model with order $r << n$:

$$\begin{aligned} \dot{\mathbf{x}}_r(t) &= \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{B}_r\mathbf{u}(t) \\ \mathbf{y}_r(t) &= \mathbf{C}_r\mathbf{x}_r(t) \end{aligned} \tag{3.2}$$

The model reduction techniques discussed above can be modified so that we can use these techniques to reduce unstable systems.

3.1 Optimal $\mathcal{L}_2$ Model Reduction

Before we describe the iteratively corrected rational Krylov algorithms, we are going to discuss briefly $\mathcal{L}_2$ systems and the $\mathcal{L}_2$ norm. Let $\mathcal{L}_2^n(\mathbb{R})$ be the set of vector-valued functions

with finite "energy" on $\mathbb{R}$:

$$\mathcal{L}_2^n(\mathbb{R}) = \{\mathbf{x}(t) \in \mathbb{R}^n : \int_{-\infty}^{\infty} \|\mathbf{x}(t)\|^2 \, dt < \infty\}.$$

Define $\mathcal{A} : \mathcal{L}_2^n(\mathbb{R}) \mapsto \mathcal{L}_2^n(\mathbb{R})$ as $\mathcal{A}\mathbf{x} = \dot{\mathbf{x}} - \mathbf{A}\mathbf{x}$ on all vector-valued functions $\mathbf{x}(t) \in \mathcal{L}_2^n(\mathbb{R})$ with absolutely continuous components and derivative $\dot{\mathbf{x}} \in \mathcal{L}_2^n(\mathbb{R})$. If $\mathbf{A}$ has eigenvalues that lie on the imaginary axis, then there is $\mathbf{f} \in \mathcal{L}_2^n(\mathbb{R})$ such that $\mathcal{A}\mathbf{x} = \mathbf{f}$ does not have a solution in $\mathcal{L}_2^n(\mathbb{R})$. On the other hand, if $\mathbf{A}$ has no purely imaginary eigenvalues, $\mathcal{A}\mathbf{x} = \mathbf{f}$ has a unique solution in $\mathcal{L}_2^n(\mathbb{R})$. Using (3.1) and the defintion of $\mathcal{A}$, we obtain the following input-output mapping, which is also a convolution operator

$$\mathbf{y}(t) = [\mathbf{C}\mathcal{A}^{-1}\mathbf{B}]\mathbf{u}(t) = \int_{-\infty}^{\infty} h(t-\tau)\mathbf{u}(\tau)d\tau \tag{3.3}$$

Following the discussion in [2], if the eigenvalues of $\mathbf{A}$ lie to the left of the imaginary axis we have

$$[\mathcal{A}^{-1}\mathbf{f}](t) = \mathbf{x}(t) = \int_{-\infty}^{t} e^{\mathbf{A}(t-\tau)}\mathbf{f}(\tau)d\tau,$$
$$\mathbf{y}(t) = [\mathbf{C}\mathcal{A}^{-1}\mathbf{B}\mathbf{u}](t) = \int_{-\infty}^{\infty} h(t-\tau)\mathbf{u}(\tau)d\tau$$

and

$$h(t) = \begin{cases} \mathbf{C}e^{\mathbf{A}t}\mathbf{B} & t \geq 0 \\ 0 & t < 0 \end{cases} \tag{3.4}$$

On the other hand, if the eigenvalues of $\mathbf{A}$ have positive real parts, then

$$[\mathcal{A}^{-1}\mathbf{f}](t) = \mathbf{x}(t) = -\int_t^{-\infty} e^{\mathbf{A}(t-\tau)}\mathbf{f}(\tau)d\tau,$$
$$\mathbf{y}(t) = [\mathbf{C}\mathcal{A}^{-1}\mathbf{B}\mathbf{u}](t) = \int_{-\infty}^{\infty} h(t-\tau)\mathbf{u}(\tau)d\tau$$

and

$$h(t) = \begin{cases} 0 & t \geq 0 \\ -\mathbf{C}e^{\mathbf{A}t}\mathbf{B} & t < 0 \end{cases} \tag{3.5}$$

If the system is unstable i.e. the eigenvalues of $\mathbf{A}$ lie both to the left and to right of the imaginary axis, we separate the system into two parts, where one is stable and the other is antistable. Let $\mathbf{X}^+$ be a basis for $\mathcal{U}^+$ and $\mathbf{X}^-$ for $\mathcal{U}^-$ where $\mathcal{U}^+$ and $\mathcal{U}^-$ are invariant subspaces of $\mathbf{A}$ corresponding to stable and antistable eigenvalues, respectively. In other words, $\mathcal{U}^+ = \text{Ran}(\mathbf{X}^+)$ and $\mathcal{U}^- = \text{Ran}(\mathbf{X}^-)$. Since $\dim(\mathcal{U}^+)+\dim(\mathcal{U}^-) = n$, the matrix $\mathbf{X} = [\mathbf{X}^+\mathbf{X}^-]$ has rank n, hence it's nonsingular. Thus, we can write

$$\mathbf{A}[\mathbf{X}^+\mathbf{X}^-] = [\mathbf{X}^+\mathbf{X}^-]\begin{bmatrix} \mathbf{M}^+ & 0 \\ 0 & \mathbf{M}^- \end{bmatrix}$$

where $\mathbf{M}^+$ is stable and $\mathbf{M}^-$ is antistable. If we let $\mathbf{Y} = (\mathbf{X}^{-1})^T = [\mathbf{Y}^+\mathbf{Y}^-]$, then,

$$\mathbf{\Pi}^+ = \mathbf{X}^+(\mathbf{Y}^+)^T \text{ and } \mathbf{\Pi}^- = \mathbf{X}^-(\mathbf{Y}^-)^T$$

are the stable and antistable projectors for $\mathbf{A}$. These spectral projectors enable us to separate a linear unstable system into its stable and antistable components. In this case we have

$$\mathbf{y}(t) = [\mathbf{C}\mathcal{A}^{-1}\mathbf{B}\mathbf{u}](t) = \int_{-\infty}^{\infty} h(t-\tau)\mathbf{u}(\tau)d\tau$$

where

$$h(t) = \begin{cases} \mathbf{C}e^{\mathbf{A}t}\mathbf{\Pi}^{+}\mathbf{B} & t \geq 0 \\ -\mathbf{C}e^{\mathbf{A}t}\mathbf{\Pi}^{-}\mathbf{B} & t < 0 \end{cases}. \tag{3.6}$$

As we can see, it is possible to separate an unstable system into its stable and antistable components. Since a stable system can be reduced via IRKA relatively easy, [2] applied IRKA to the stable subsystem and the negative of the antistable component of the unstable system. After reducing each component separately, we negate the component which corresponds to the antistable part of the system so that we obtain a reduced antistable component, we put the components together. Hence, we obtain a reduced unstable system. Since model reduction is an approximation problem, we need to measure the error. In this case, we will use the frequency domain representation. Recall in section 2.1 we obtained the frequency domain representation of the dynamical systems by taking the Lalplace transform of the time representations (2.1) and (2.2). However, the existence of the Laplace transform of $\mathbf{u}$ is not guaranteed if $\mathbf{u} \in \mathcal{L}_2^n(\mathbb{R})$. Thus, we apply a Fourier transform to (3.6) and obtain

$$\hat{\mathbf{y}}(\omega) = \mathbf{C}(i\omega\mathbf{I} - \mathbf{A})^{-1}\mathbf{\Pi}^{+}\mathbf{B}\hat{\mathbf{u}}(\omega) + \mathbf{C}(i\omega\mathbf{I} - \mathbf{A})^{-1}\mathbf{\Pi}^{-}\mathbf{B}\hat{\mathbf{u}}(\omega) = \mathbf{C}(i\omega\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}(\omega)$$

where $\hat{\mathbf{y}}(\omega)$ and $\hat{\mathbf{u}}(\omega)$ are the Fourier transforms of $\mathbf{y}(t)$ and $\mathbf{u}(t)$, respectively. Then,

$$\mathbf{H}^{+}(i\omega)\hat{\mathbf{u}}(\omega) + \mathbf{H}^{-}(i\omega)\hat{\mathbf{u}}(\omega) = \mathbf{H}(i\omega)\hat{\mathbf{u}}(\omega)$$

where $\mathbf{H}$ is the total transfer function, $\mathbf{H}^{+}$ the transfer function of the stable part and $\mathbf{H}^{-}$ the transfer function of the antistable part. Let $\mathcal{L}_2(i\mathbb{R})$ be the Hilbert space whose elements are the meromorphic functions $\mathbf{G}(s)$ such that $\int_{-\infty}^{\infty} |\mathbf{G}(i\omega)|^2 d\omega$ is finite. Note that the transfer functions of the stable and the antistable parts of the dynamical systems are contained in $\mathcal{L}_2(i\mathbb{R})$. Then, for any two functions $\mathbf{G}, \mathbf{H}$ in $\mathcal{L}_2(i\mathbb{R})$ that represent real dynamical systems,

the inner product is defined as

$$\langle \mathbf{G}, \mathbf{H} \rangle = \int_{-\infty}^{\infty} \mathbf{G}(-i\omega)\mathbf{H}(i\omega)d\omega.$$

As a result, the $\mathcal{L}_2(i\mathbb{R})$ norm of $\mathbf{H}$ is

$$\|\mathbf{H}\|_{\mathcal{L}_2} = \left(\int_{-\infty}^{\infty} |\mathbf{H}(i\omega)|^2 d\omega \right)^{1/2} \tag{3.7}$$

We know $\mathcal{L}_2(i\mathbb{R})$ can be written as a direct sum : $\mathcal{L}_2(i\mathbb{R}) = \mathcal{H}_2(\mathbb{C}^-) \oplus \mathcal{H}_2(\mathbb{C}^+)$. If $\mathbf{H}_r^+$ denotes the stable reduced subsystem, $\mathbf{H}_r^-$ denotes the antistable reduced subsystem, and $\mathbf{H}_r$ denotes the total reduced unstable system, then we have

$$\|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{L}_2}^2 = \left\|\mathbf{H}^+ - \mathbf{H}_r^+\right\|_{\mathcal{H}_2(\mathbb{C}^+)}^2 + \left\|\mathbf{H}^- - \mathbf{H}_r^-\right\|_{\mathcal{H}_2(\mathbb{C}^-)}^2.$$

Thus, in order to compute the $\mathcal{L}_2$ error in optimal $\mathcal{L}_2$ model reduction, we need to compute the $\mathcal{H}_2$ error obtained during the reduction of the stable and antistable components. Below is a sketch of $\mathcal{L}_2$ IRKA

Sketch of $\mathcal{L}_2$ IRKA

- Decompose $\mathbf{H}$ into minimal stable and antistable systems.
- Make an initial shift selection closed under conjugation and ordered as follows $\{\sigma_1, \cdots, \sigma_k\} \subset \mathbb{C}^+$ and $\{\sigma_{k+1}, \cdots, \sigma_r\} \subset \mathbb{C}^-$
- Negate the antistable subsystem.
- Reduce each subsystem via IRKA

- Negate the reduced subsystem corresponding to the antistable subsystem
- Add the reduced stable and antistable systems.

For a detailed description of $\mathcal{L}_2$ IRKA see [2].

3.2 Balanced Truncation for Unstable System

A balanced realization and Gramian based model reduction methods may be used to reduce the order of a possibly unstable dynamical system. Let's write (3.1) as

$$\boldsymbol{\Sigma} := \left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & 0 \end{array}\right]$$

In order to get a balanced realization of the system, we need to compute the reachability and observability gramians $\mathcal{P}$ and $\mathcal{Q}$. For stable systems we obtain the gramians $\mathcal{P}$ and $\mathcal{Q}$ by solving the Lyapunov equations (2.3). However, if $\mathbf{A}$ has eigenvalues whose real part is positive, the Lyapunov equations (2.3) may not have a solution at all. Hence, we need to redefine the reachability and observability Gramians. Suppose $\mathbf{T}$ is a transformation such that

$$\left[\begin{array}{c|c} \mathbf{TAT}^{-1} & \mathbf{TB} \\ \hline \mathbf{CT}^{-1} & 0 \end{array}\right] = \left[\begin{array}{cc|c} \mathbf{A}_1 & 0 & \mathbf{B}_1 \\ 0 & \mathbf{A}_2 & \mathbf{B}_2 \\ \hline \mathbf{C}_1 & \mathbf{C}_2 & 0 \end{array}\right]$$

where $\mathbf{A}_1$ is stable and $\mathbf{A}_2$ is antistable. Let $\mathcal{P}_1, \mathcal{P}_2, \mathcal{Q}_1, \mathcal{Q}_2 \geq 0$ be solutions to the Lyapunov equations:

$$\begin{aligned}
\mathbf{A}_1\mathcal{P}_1 + \mathcal{P}_1\mathbf{A}_1^T + \mathbf{B}_1\mathbf{B}_1^T &= 0, \\
\mathbf{A}_1^T\mathcal{Q}_1 + \mathcal{Q}_1\mathbf{A}_1 + \mathbf{C}_1^T\mathbf{C}_1 &= 0, \\
(-\mathbf{A}_2)\mathcal{P}_2 + \mathcal{P}_1(-\mathbf{A}_2)^T + \mathbf{B}_2\mathbf{B}_2^T &= 0, \\
(-\mathbf{A}_2)^T\mathcal{Q}_2 + \mathcal{Q}_2(-\mathbf{A}_2) + \mathbf{C}_2^T\mathbf{C}_2 &= 0.
\end{aligned}$$

Zhou et al., 1999 showed $\mathcal{P}$ and $\mathcal{Q}$ can be computed as

$$\mathcal{P} = \mathbf{T}^{-1} \begin{bmatrix} \mathcal{P}_1 & 0 \\ 0 & \mathcal{P}_2 \end{bmatrix} \mathbf{T}^{-T}$$

$$\mathcal{Q} = \mathbf{T}^{-1} \begin{bmatrix} \mathcal{Q}_1 & 0 \\ 0 & \mathcal{Q}_2 \end{bmatrix} \mathbf{T}^{-T}$$

Recall the generalized Hankel singular values are defined as $\sigma_i = \sqrt{\lambda_i(\mathcal{P}\mathcal{Q})}$ such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$. In other words, the generalized Hankel singular values of an unstable system can be computed via the Hankel singular values of its stable and antistable components. After this point, we follow the same steps as in balanced truncation for stable systems. This method eliminates the states associated with the smallest Hankel singular values without making a distinction between the values associated with the stable subsystem and those associated with the antistable part. The only criterion for the elimination is the magnitude of the Hankel singular value.

4 Model Reduction of Unstable Systems with IRKA

As we discussed in Chapter 2, IRKA yields a reduced model satisfying the first-order necessary conditions for $\mathcal{H}_2$ optimality. And for SISO systems, upon convergence, it guarantees at least a locally optimal reduced model. Even though the stability of the full system is a requirement for the optimality of the reduced system, from a computational perspective, the original system does not have to be stable in order to be reduced via IRKA. As long as a dynamical system does not have any poles in the imaginary axis, IRKA can be used to reduce such system. In this section we investigate IRKA's behavior when it is applied directly to an unstable system i.e. we do not separate the unstable system into its stable and antistable components. Instead, we reduce the unstable system via IRKA the same way we reduce a stable system. Recall that for the approximation of certain unstable systems, in addition to accuracy, we want the reduced order model to have the same number of unstable poles as the full model e.g. controller reduction. Obviously, when approximating unstable systems, we do not expect IRKA to yield the same level of accuracy as with stable systems; however, for systems with a small number of unstable poles, we observe that IRKA not only converges to a reduced order model that has the same number of poles as the original model, but also captures the value of the unstable poles. Throughout this section we reduce unstable finite element models for an Euler-Bernoulli cantilever beam. These systems were generated via a Matlab program. We pick the initial shifts to be mirror images of the poles of the system. After running several trials with unstable systems, we note that if we use enough shifts, IRKA converges and also captures the unstable poles. In fact, we investigated the results of IRKA for several cases.

4.1 Pole Capturing by IRKA

In this section, we examine IRKA's behavior when the full system has few unstable poles. First, we consider a system with 400 stable poles and only two unstable poles. Tables 1 through 4 show the results we obtained after reducing the aforementioned system via IRKA. If we approximate the full model with a reduced order model of size $r = 2$, we notice the reduced model has one stable and one unstable pole, as is illustrated in Table 1. When the full model is reduced to a model of order $r = 3$, one of the unstable poles is captured. This is shown in Table 2. In Table 3 we see that after applying IRKA with $r = 4$ we obtain a reduced order model with two unstable poles which are very close to the unstable poles of the original system. If we increase the size of the reduced order model i.e. let $r = 5$ the poles of the original system are captured even more accurately as we can see in Table 4.

Table 1: FOM (2 unstable poles) vs. ROM poles ($r = 2$)

FOM poles	ROM poles
1.0000×10^{-1}	9.4726×10^{-3}
1.0000×10^{-2}	$-\ 9.9733\times10^{-2}$

Table 2: FOM (2 unstable poles) vs. ROM poles ($r = 3$)

FOM poles	ROM poles
1.0000×10^{-1}	1.0009×10^{-2}
1.0000×10^{-2}	-1.0711×10^{-2}
-8.0280×10^{3}	-9.9729×10^{-1}

Table 3: FOM (2 unstable poles) vs. ROM poles ($r = 4$)

FOM poles	ROM poles
1.000000×10^{-1}	9.999997×10^{-2}
1.000000×10^{-2}	9.998589×10^{-3}
-5.607697×10^{3}	-1.252094×10^{-2}
-8.027997×10^{3}	-9.968973×10^{-2}

Table 4: FOM (2 unstable poles) vs. ROM poles ($r = 5$)

FOM poles	ROM poles
1.000000×10^{-1}	9.999998×10^{-2}
1.000000×10^{-2}	9.999560×10^{-3}
-5.594920×10^{3}	-2.967848×10^{-4}
-5.607697×10^{3}	-1.272002×10^{-2}
-8.027997×10^{3}	-9.968876×10^{-2}

For a system with three unstable poles and 400 stable poles we notice a similar patern. In Table 5 we see the results of the approximation when the size of the reduced order model is $r = 2$. The reduced system has one stable and one unstable pole. For $r = 3$, the reduced order model has two unstable poles and a stable one, as is illustrated in Table 6. In Tables 7 and 8, we have the results of model reduction with $r = 4$ and $r = 5$, respectively. In these cases, we do not observe any patterns arising in the poles of the reduced model. Even though the reduced system of order $r = 6$ has three unstable poles, the same number of unstable poles in the full model, only one of the poles of the original model is captured fairly accurately. See Table 9 for the values of the rest of the poles of the reduced model of order

$r = 6$. In Table 10, we notice the unstable poles of the reduced system are getting closer to the unstable poles of the original system. If we implement IRKA with eight intepolation points, the unstable poles are captured accurately, as we can see in Table 10.

Table 5: FOM (3 unstable poles) vs. ROM poles ($r = 2$)

FOM poles	ROM poles
2.0000	-3.7120
1.0000×10^{-2}	7.7740×10^{-3}

Table 6: FOM (3 unstable poles) vs. ROM poles ($r = 3$)

FOM poles	ROM poles
1.0000×10^{1}	7.9467
2.0000	9.9989×10^{-3}
1.0000×10^{-2}	-1.2451×10^{-2}

Table 7: FOM (3 unstable poles) vs. ROM poles ($r = 4$)

FOM poles	ROM poles
1.0000×10^{1}	9.9520
2.0000	2.2951
1.0000×10^{-2}	9.9988×10^{-3}
-8.0280×10^{3}	-1.2475×10^{-2}

Table 8: FOM (3 unstable poles) vs. ROM poles ($r = 5$)

FOM poles	ROM poles
1.0000×10^{1}	7.4594
2.0000	9.9998×10^{-3}
1.0000×10^{-2}	-2.8741×10^{-4}
-5.6077×10^{3}	-1.2647×10^{-2}
-8.0280×10^{3}	-2.3438×10^{1}

Table 9: FOM (3 unstable poles) vs. ROM poles ($r = 6$)

FOM poles	ROM poles
1.0000×10^{1}	7.7571
2.0000	1.3304×10^{-1}
1.0000×10^{-2}	9.9998×10^{-3}
-5.5949×10^{3}	-2.8760×10^{-4}
-5.6077×10^{3}	-1.2648×10^{-2}
-8.0280×10^{3}	-9.0912

Table 10: FOM (3 unstable poles) vs. ROM poles ($r = 7$)

FOM poles	ROM poles
1.0000×10^{1}	9.2969
2.0000	1.6110
1.000000×10^{-2}	1.000002×10^{-2}
-5.5737×10^{3}	-2.7959×10^{-4}
-5.5949×10^{3}	-1.2566×10^{-2}
-5.6077×10^{3}	-1.1960×10^{-1}
-8.0280×10^{3}	-3.1682×10^{-1}

Table 11: FOM (3 unstable poles) vs. ROM poles ($r = 8$)

FOM poles	ROM poles
1.000000×10^{1}	1.004201×10^{1}
2.000000	2.003875
1.000000×10^{-2}	1.000004×10^{-2}
-5.544320×10^{3}	-2.786650×10^{-4}
-5.573738×10^{3}	-1.255170×10^{-2}
-5.594920×10^{3}	-7.423360×10^{-2}
-5.607700×10^{3}	$-9.068348\times10^{-1} - 3.404822\times10^{-1}$i
-8.027997×10^{3}	$-9.068348\times10^{-1} + 3.404822\times10^{-1}$i

The numerical results for reducing an unstable system with 4 unstable poles are presented in the tables 12 through 19. We reduced the model using only two interpolation points, the unstable poles were not captured at all. Actually, the reduced model had one stable and

one unstable pole even though we picked the mirror images of two unstable poles as initial shifts. Table 12 illustrates the results for $r = 2$, where r refers to the size of the reduced model. When we use three initial shifts with the same system, the behavior is similar. The reduced model has three unstable poles, but they are not captured accurately. Table 13 shows the results for $r = 3$. Even for a reduced model of size $r = 4$, the reduced model does not have the same number of unstable poles as the original model. We end up with three unstable poles and one stable pole. We conclude this by observing the results displayed in Table 14. If we use five initial shifts, we obtain a reduced system that preserves the stability. It has four stable poles and one unstable pole and we started with the mirror images of four unstable poles and one stable pole, as shown in Table 15. However, the unstable poles are not captured accurately at all. If we use six shifts, we notice reduced order model has 4 unstable poles; however, the unstable poles still are not captured very accurately. In Table 16 we see the reduced model has four unstable poles and two stable poles. Similar results are obtained for seven shifts even though we notice the unstable poles are somewhat captured in table 17. However, the accuracy could be much better. Table 18 illustrates this. When we use eight initial shifts, we can see that the stability of the system is preserved and the unstable poles are captured fairly accurately as the results presented in Table 18 show. When we use nine shifts, the accuracy of the pole capture improves further, as shown in Table 19.

Table 12: FOM (4 unstable poles) vs. ROM poles ($r = 2$)

FOM poles	ROM poles
1.0069×10^{-2} - 5.1435×10^{-2}i	1.0777×10^{-3}
1.0069×10^{-2} + 5.1435×10^{-2}i	-1.5843$\times10^{-2}$

Table 13: FOM (4 unstable poles) vs. ROM poles ($r = 3$)

FOM poles	ROM poles
1.5321×10^{-3}	7.4006×10^{-3} - 6.7000×10^{-2}i
1.0069×10^{-2} - 5.1435×10^{-2}i	7.4006×10^{-3} + 6.7000×10^{-2}i
1.0069×10^{-2} + 5.1435×10^{-2}i	1.0231×10^{-3}

Table 14: FOM (4 unstable poles) vs. ROM poles ($r = 4$)

FOM poles	ROM poles
1.5321×10^{-3}	1.0607×10^{-3}
1.8469×10^{-2}	5.7499×10^{-4} - 5.9429×10^{-2}i
1.0069×10^{-2} - 5.1435×10^{-2}i	5.7499×10^{-4} + 5.9429×10^{-2}i
1.0069×10^{-2} + 5.1435×10^{-2}i	-1.6069×10^{-3}

Table 15: FOM (4 unstable poles) vs. ROM poles ($r = 5$)

FOM poles	ROM poles
1.5321×10^{-3}	1.8932×10^{-2}
1.8469×10^{-2}	1.3618×10^{-2}- 5.2176×10^{-2}i
1.0069×10^{-2} - 5.1435×10^{-2}i	1.3618×10^{-2} + 5.2176×10^{-2}i
1.0069×10^{-2} + 5.1435×10^{-2}i	1.3726×10^{-3}
-8.0280×10^{-3}	-1.2079×10^{-2}

Table 16: FOM (4 unstable poles) vs. ROM poles ($r = 6$)

FOM poles	ROM poles
1.5321×10^{-3}	1.5279×10^{-3}
1.8469×10^{-2}	1.8606×10^{-2}
1.0069×10^{-2} - 5.1435×10^{-2}i	1.2770×10^{-2} - 5.2848×10^{-2}i
1.0069×10^{-2} + 5.1435×10^{-2}i	1.2770×10^{-2} + 5.2848×10^{-2}i
-5.6077×10^{3}	-2.9187×10^{-4}
-8.0280×10^{3}	-1.2707×10^{-2}

Table 17: FOM (4 unstable poles) vs. ROM poles ($r = 7$)

FOM poles	ROM poles
1.5321×10^{-3}	1.5349×10^{-3}
1.8469×10^{-2}	1.8431×10^{-3}
1.0069×10^{-2} - 5.1435×10^{-3}i	1.0090×10^{-3} - 5.2116×10^{-2}i
1.0069×10^{-2} + 5.143×10^{-2}i	1.0090×10^{-2} + 5.2116×10^{-2}i
-5.5949×10^{3}	- 2.7355×10^{-4}
-5.607×10^{3}	-1.2353×10^{-2}
-8.0280×10^{3}	-1.1053×10^{-1}

Table 18: FOM (4 unstable poles) vs. ROM poles ($r = 8$)

FOM poles	ROM poles
1.5321×10^{-3}	1.5324×10^{-3}
1.8469×10^{-2}	1.8467×10^{-2}
1.0069×10^{-2} - 5.1435×10^{-2}i	1.0088×10^{-2} - 5.1433×10^{-2}i
1.0069×10^{-2} + 5.1435×10^{-2}i	1.0088×10^{-2} + 5.1433×10^{-2}i
-5.5737×10^{3}	-2.8033×10^{-4}
-5.5949×10^{3}	- 1.2560×10^{-3}
- 5.607×10^{3}	-3.9537×10^{-2} - 6.7733×10^{-2}i
-8.0280×10^{-3}	- 3.9537×10^{-2} + 6.7733×10^{-2}i

Table 19: FOM (4 unstable poles) vs. ROM poles ($r = 9$)

FOM poles	ROM poles
1.84693×10^{-2}	1.84696×10^{-2}
1.00687×10^{-2} 5.143482×10^{-2}i	-1.00565×10^{-2} - 5.14279×10^{-2}i
1.00687×10^{-2} + 5.14348×10^{-2}i	1.00565×10^{-2} + 5.14279×10^{-2}i
1.5320952×10^{-3}	1.5320949×10^{-3}
-5.54432×10^{3}	-2.81072×10^{-4}
- 5.57374×10^{3}	- 1.25874×10^{-2}
-5.59492×10^{3}	-4.42412×10^{-2} - 6.018044×10^{-2}i
-5.60770×10^{3}	-4.42412×10^{-2} + 6.01804×10^{-2}i
-8.02800×10^{3}	-1.57396

As we increase the size of the reduced model, its unstable poles get closer and closer to the

unstable poles of the original model. This means the number of shifts used with IRKA is directly related to the accuracy with which the unstable poles are captured. In other words, more shifts imply better accuracy. Also, as the number of unstable poles of the original model becomes larger, the number of shifts required to accurately capture these unstable poles becomes larger as well. As we have seen in the examples above, if we have only two unstable poles, five initial shifts suffice to capture the unstable poles accurately. For a system with three unstable poles, we need to use eight initial shifts. If the full model has four unstable poles, the size of the reduced model needs to be $r = 9$ in order to capture the unstable poles closely.

If the original model has eight unstable poles, we need at least 14 initial shifts in order to capture the unstable poles. If the number of unstable poles keeps increasing, so does the number of shifts that are necessary to capture the unstable poles. If there are 12 unstable poles in the original model, we need to use at least 36 interpolation points. Also, if we have many unstable poles and only few stable ones, we notice that this behavior is mirrored, i.e. the stable poles are captured provided we use enough interpolation points. However, if we have a model that has an equal number of stable and unstable poles, e.g. let's say there are 60 stable poles and 60 unstable poles, we are unable to capture any poles. The results in the following table pertain to an unstable system with 60 stable poles and 60 antistable poles. If we use 12 shifts, after applying IRKA we obtain a reduced model that has seven unstable poles and five stable ones.

Table 20: FOM (60 stable +60 unstable poles) vs. ROM poles ($r = 12$)

FOM poles	ROM poles
1.4574×10^{1}	1.4160×10^{-4}
1.9447×10^{1}	-2.7980×10^{-4}
2.55478×10^{1}	7.0623×10^{-3}
3.3187×10^{1}	-1.0864×10^{-3}
4.2722×10^{1}	$2.9155\times10^{-3}+ 6.109\times10^{-2}$i
5.4523×10^{1}	$2.9155\times10^{-3}-6.109\times10^{-2}$i
6.8876×10^{1}	6.2504×10^{-2}
8.5758×10^{1}	$-8.7541\times10^{-2} + 6.0785\times10^{-3}$i
1.0443×10^{2}	$-8.7541\times10^{-2} - 6.0785\times10^{-3}$i
1.2284×10^{2}	1.33461×10^{-1}
1.3721×10^{2}	-2.9143×10^{-1}
2.0395×10^{2}	3.3870×10^{-1}

4.2 Comparing IRKA for Unstable Systems with Other Model Reduction Techniques

In order to compare different model reduction techniques, we reduce several generated beam models whose number of unstable poles varies. We reduce each one of these models using IRKA for unstable systems, $\mathcal{L}_2$ IRKA, and balanced truncation. For each model we use each technique three times and approximate the original system by a reduced order model of size 8, 10 and 12. All the full models we consider have 120 poles in total, but the number of the unstable poles varies from 4 to 20. In Table 21, we have recorded the $\mathcal{L}_2$ and $\mathcal{L}_\infty$ errors after

reducing the full models to a model of size $r = 12$ with IRKA. We notice the errors have a tendency to increase as the number of unstable poles of the full model becomes larger. In Figure 1 we have plotted the Bode plots of the full model, which has 4 unstable poles and the reduced model of size $r = 12$ which was obtained via IRKA.

Table 21: IRKAfUS error as the no. of unstable poles varies ($r = 12$)

No. of unstable poles	4	8	12	16	20
$\mathcal{L}_2$ Error	0.0334	0.0332	0.0306	0.0370	0.0365
$\mathcal{L}_\infty$ Error	0.0012	0.0022	0.0028	0.0031	0.0028

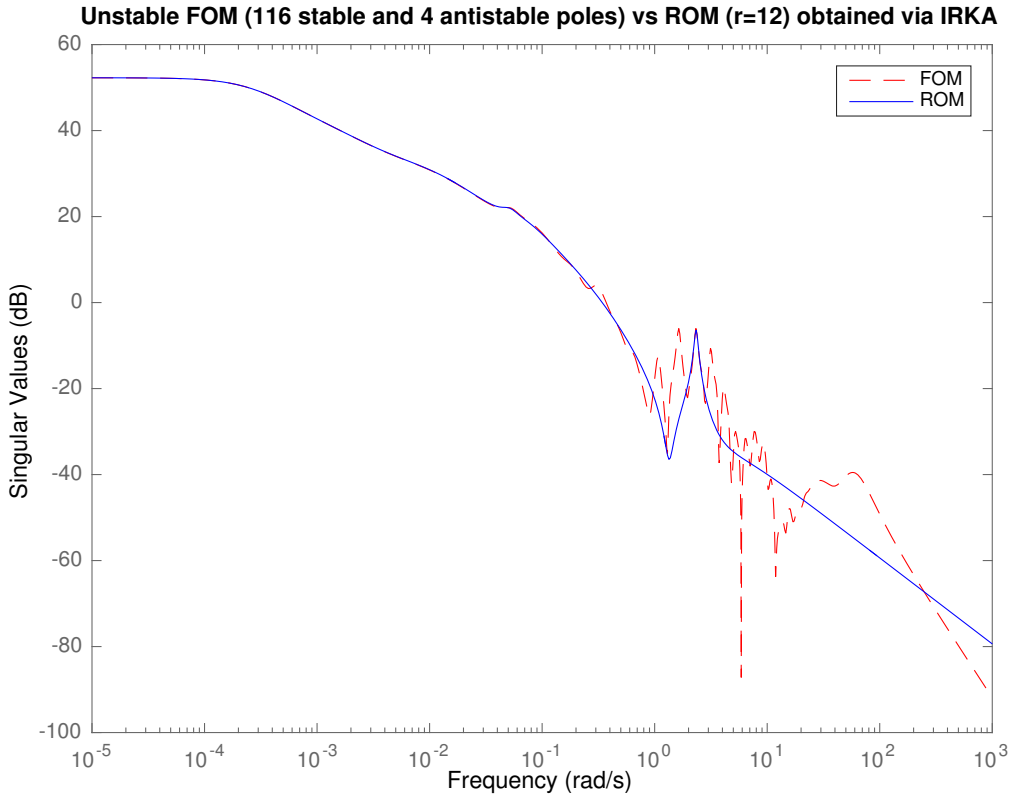

Figure 1: Bode Plots for FOM and ROM obtained via IRKA ($r = 12$)

In Table 22, we have recorded the $\mathcal{L}_2$ and $\mathcal{H}_\infty$ errors after reducing the full models to a model of size $r = 12$ with $\mathcal{L}_2$ IRKA. The magnitude of the errors increases as the number of unstable poles of the full model becomes larger. In Figure 2 we have plotted the Bode plots of the full model, which has 4 unstable poles and the reduced model of size $r = 12$, which was obtained via $\mathcal{L}_2$ IRKA.

Table 22: $\mathcal{L}_2$ IRKA error as the no. of unstable poles varies ($r = 12$)

No. of unstable poles	4	8	12	16	20
$\mathcal{L}_2$ Error	0.0532	0.0722	0.1189	0.1671	0.2192
$\mathcal{L}_\infty$ Error	0.0963	0.1259	0.1884	0.2463	0.2974

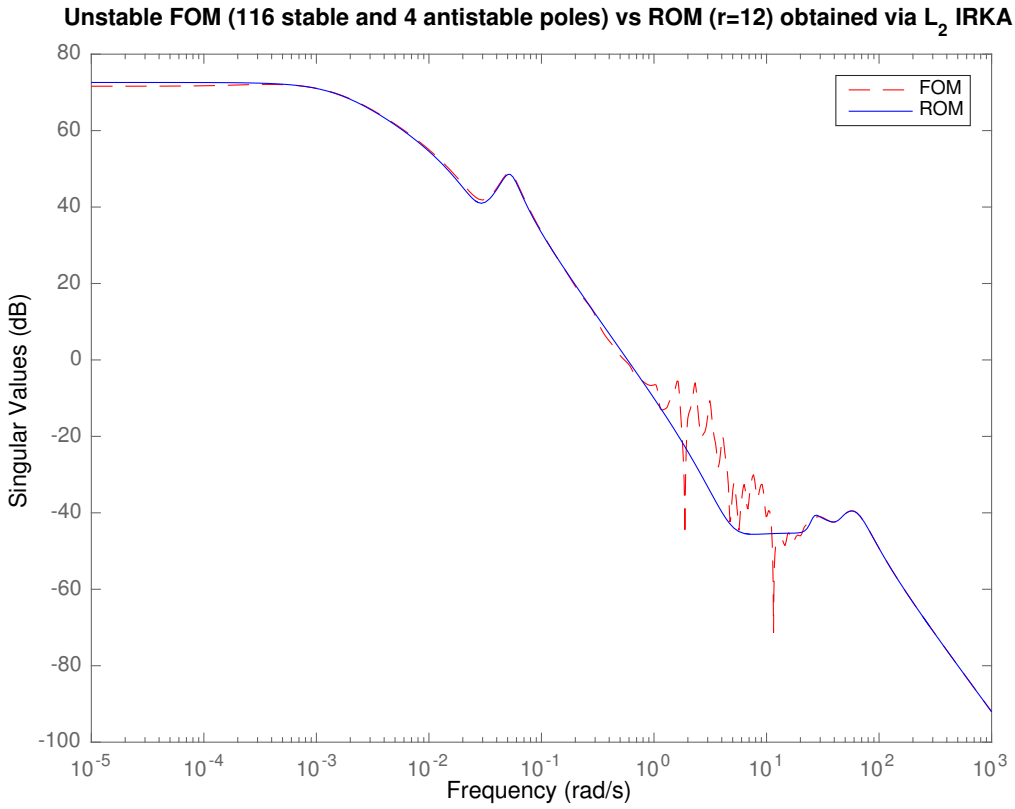

Figure 2: Bode Plots for FOM and ROM obtained via $\mathcal{L}_2$ IRKA ($r = 12$)

In Table 23, we have recorded the $\mathcal{L}_2$ and $\mathcal{L}_\infty$ errors after reducing the full models to a model of size $r = 12$ with $\mathcal{L}_2$ balanced truncation. Even with balanced truncation, the magnitude of the errors increases as the number of unstable poles of the full model becomes larger. In Figure 3 we have plotted the Bode plots of the full model, which has 4 unstable poles and the reduced model of size $r = 12$, which was obtained via balanced truncation.

Table 23: Balanced truncation error as the no. of unstable poles varies ($r = 12$)

No. of unstable poles	4	8	12	16	20
$\mathcal{L}_2$ Error	0.0021	0.0050	0.0221	0.0299	0.0326
$\mathcal{L}_\infty$ Error	1.2×10^{-4}	3.0×10^{-4}	0.0033	0.0028	0.0033

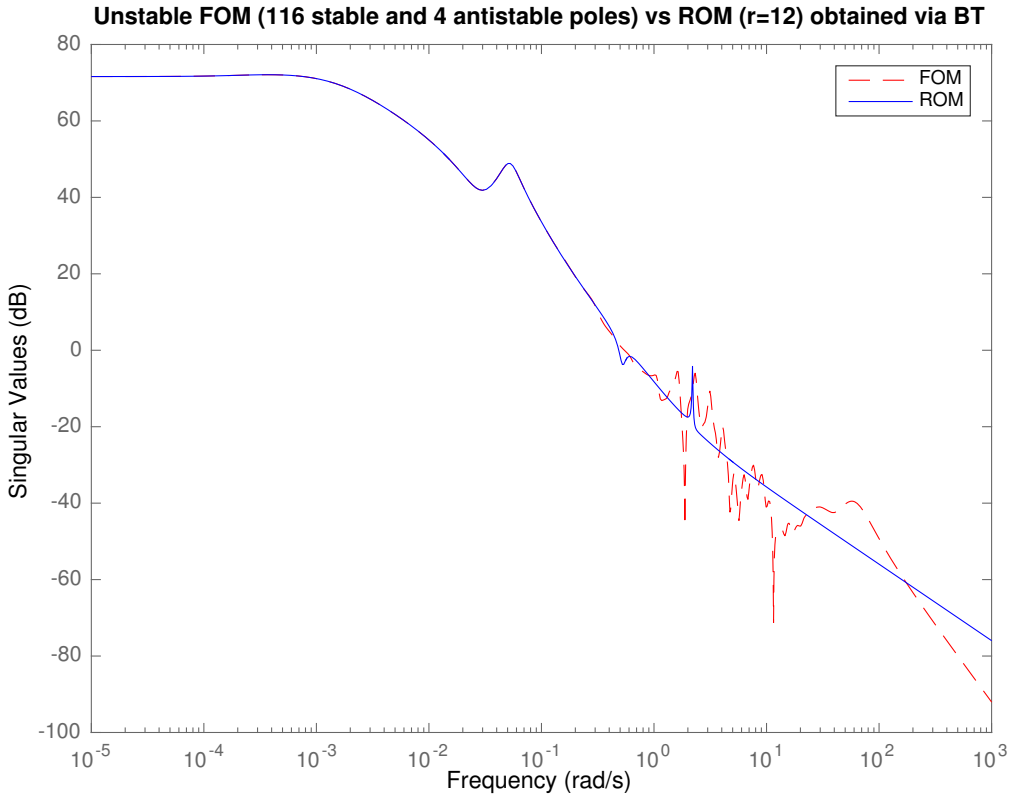

Figure 3: Bode Plots for FOM and ROM obtained via balanced truncation ($r = 12$)

In Tables 24, 25, and 26 we have recorded the $\mathcal{L}_2$ and $\mathcal{L}_\infty$ errors of the approximation by IRKA, $\mathcal{L}_2$ IRKA, and balanced truncation, respectively. In all these cases we approximated

the full model by a reduced model of order $r = 10$. In these examples, we see the same patterns we saw when we approximated the original model by a reduced model of order $r = 12$. In other words, even for $r = 10$, the relative errors are larger, if the number of the unstable poles of the original system is large.

Table 24: IRKAfUS error as the no. of unstable poles varies ($r = 10$)

No. of unstable poles	4	8	12	16	20
$\mathcal{L}_2$ Error	0.0388	0.0382	0.0389	0.0448	0.0423
$\mathcal{L}_\infty$ Error	0.0011	0.0022	0.0028	0.0029	0.0033

Table 25: $\mathcal{L}_2$ IRKA error as the no. of unstable poles varies ($r = 10$)

No. of unstable poles	4	8	12	16	20
$\mathcal{L}_2$ Error	0.0301	0.0907	0.9898	0.1675	0.2194
$\mathcal{L}_\infty$ Error	0.0575	0.1218	0.7310	0.2465	0.2971

Table 26: Balanced truncation error as the no. of unstable poles varies ($r = 10$)

No. of unstable poles	4	8	12	16	20
$\mathcal{L}_2$ Error	0.0022	0.0049	0.0287	0.0321	0.0582
$\mathcal{L}_\infty$ Error	1.2×10^{-4}	3.2×10^{-4}	0.0033	0.0031	0.0109

Next, we compare IRKA, $\mathcal{L}_2$ IRKA and balanced truncation results after reducing an unstable model with an equal number of stable and unstable poles. The $\mathcal{L}_2$ and $\mathcal{L}_\infty$ approximation errors are recorded in Tables 27, 28, and 29. In Figures 4, 5, and 6 we have plotted the Bode plots of the full models and the reduced order models obtained by IRKA, $\mathcal{L}_2$ IRKA, and

balanced truncation. We notice IRKA and balanced truncation yield small approximation errors for this model. That is not the case for $\mathcal{L}_2$ IRKA.

Table 27: Comparison for FOM with equal number of stable and unstable poles ($r = 8$)

Technique	IRKAfUS	$\mathcal{L}_2$ IRKA	BT
$\mathcal{L}_2$ Error	0.0693	0.5368	0.0883
$\mathcal{L}_\infty$ Error	0.0030	0.5613	0.0173

Table 28: Comparison for FOM with equal number of stable and unstable poles ($r = 10$)

Technique	IRKAfUS	$\mathcal{L}_2$ IRKA	BT
$\mathcal{L}_2$ Error	0.0669	0.5017	0.0883
$\mathcal{L}_\infty$ Error	0.0030	0.5187	0.0173

Table 29: Comparison for FOM with equal number of stable and unstable poles ($r = 12$)

Technique	IRKAfUS	$\mathcal{L}_2$ IRKA	BT
$\mathcal{L}_2$ Error	0.0531	0.4885	0.0717
$\mathcal{L}_\infty$ Error	0.0021	0.5160	0.0110

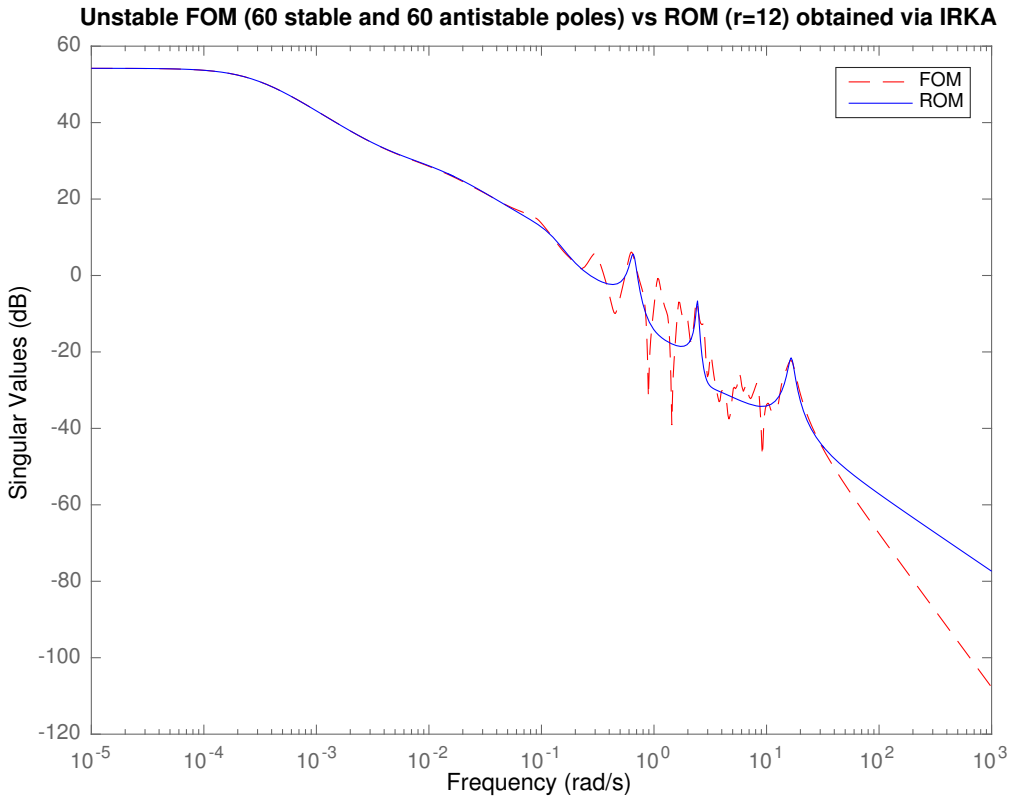

Figure 4: Bode Plots for FOM with equal number of stable and unstable poles, and ROM obtained via IRKA ($r = 12$)

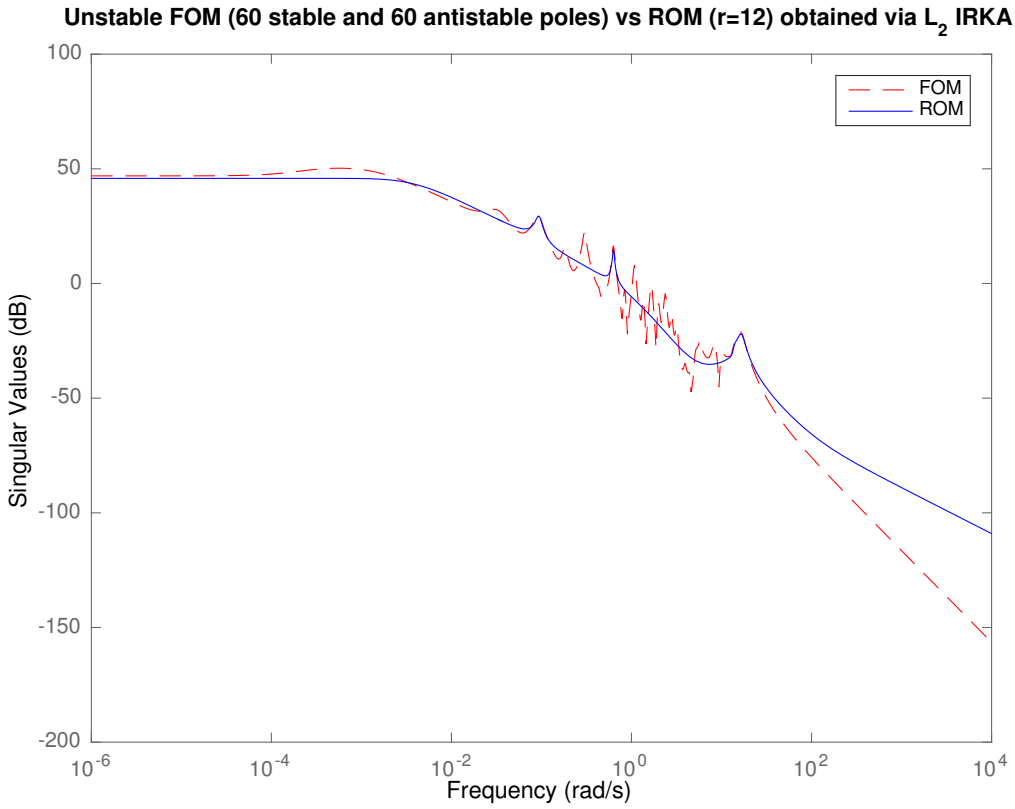

Figure 5: Bode Plots for FOM with equal number of stable and unstable poles, and ROM obtained via $\mathcal{L}_2$IRKA ($r = 12$)

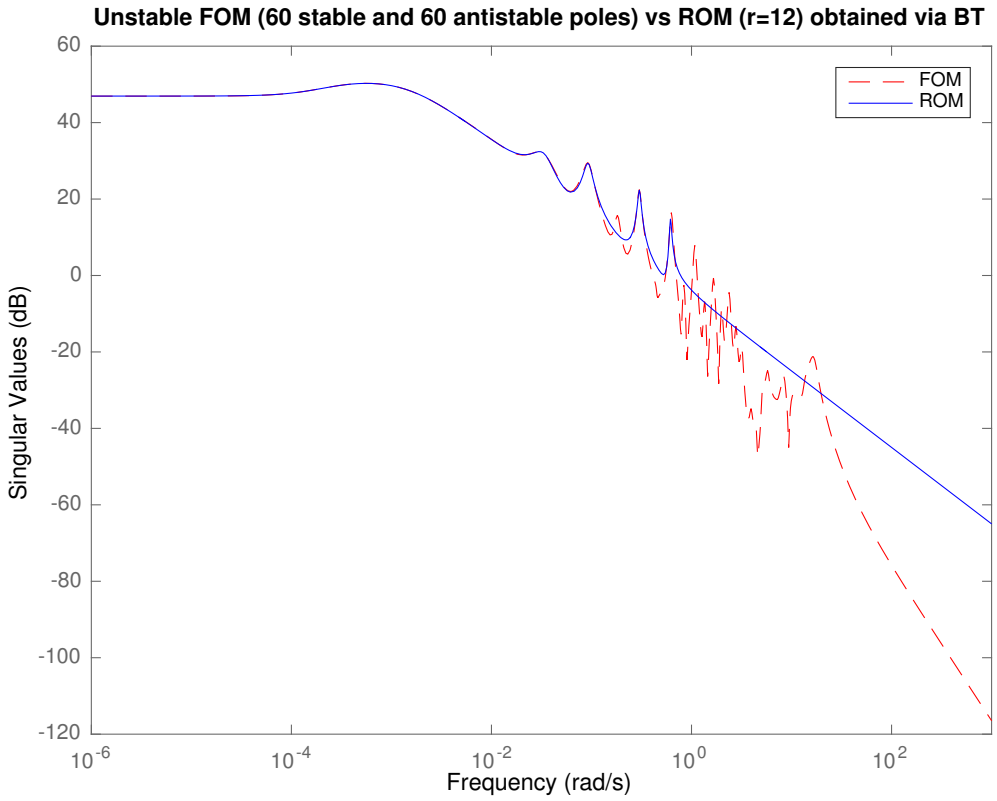

Figure 6: Bode Plots for FOM with equal number of stable and unstable poles, and ROM obtained via balanced truncation ($r = 12$)

4.3 Different Initalizations for Different Model Reduction Techniques

In this section we reduce the unstable system consisting of 60 stable and 60 unstable poles via IRKA, $\mathcal{L}_2$ IRKA, and balanced truncation. We run IRKA with two different initializations. We use a subset of the mirror images of the poles of the full system for one initialization. The other set of initial shifts consists of the reflections about the imaginary axis of the

poles of a reduced system obtained via balanced truncation. We initialized $\mathcal{L}_2$ IRKA with three different set of shifts; we use the reflections about the imaginary axis of a reduced system obtained with IRKA for unstable systems, the reflections about the imaginary axis of a reduced system obtained with balanced truncation, and a subset of the mirror images of the poles of the original system. There are no significant changes in accuracy of the approximaiton attained with IRKA for unstable systems when we use different initializations. However, it appears the intial set of shifts affects the accuracy of $\mathcal{L}_2$ IRKA notably. $\mathcal{L}_2$ IRKA yields a better approximation when it is initialized with the mirror images of the poles of a reduced system obtained either by IRKA for unstable systems or balanced truncation. The $\mathcal{L}_2$ and $\mathcal{L}_\infty$ errors for each technique with each intialization are recorded in the Tables 30 through 32.

Table 30: Comparison of different techniques with different initializations($r = 12$)

Technique	$\mathcal{L}_2$ IRKA	$\mathcal{L}_2$ IRKA	$\mathcal{L}_2$ IRKA	IRKAfUS	IRKAfUS	BT
Initialization	Poles	BT poles	IRKAfUS poles	Poles	BT poles	-
$\mathcal{L}_2$ error	0.4885	0.0457	0.0840	0.1449	0.1754	0.0717
$\mathcal{L}_\infty$ error	0.5160	0.0041	0.0072	0.0198	0.0423	0.0110

Table 31: Comparison of different techniques with different initializations($r = 10$)

Technique	$\mathcal{L}_2$ IRKA	$\mathcal{L}_2$ IRKA	$\mathcal{L}_2$ IRKA	IRKAfUS	IRKAfUS	BT
Initialization	Poles	BT poles	IRKAfUS poles	Poles	BT poles	-
$\mathcal{L}_2$ error	0.5246	0.0719	0.0840	0.1856	0.1856	0.0883
$\mathcal{L}_\infty$ error	0.5586	0.0054	0.0072	0.0424	0.0424	0.0173

Table 32: Comparison of different techniques with different initializations($r = 8$)

Technique	$\mathcal{L}_2$ IRKA	$\mathcal{L}_2$ IRKA	$\mathcal{L}_2$ IRKA	IRKAfUS	IRKAfUS	BT
Initialization	Poles	BT poles	IRKAfUS poles	Poles	BT poles	-
$\mathcal{L}_2$ error	0.6315	0.1257	0.1922	0.2076	0.1890	0.1894
$\mathcal{L}_\infty$ error	0.6376	0.0128	0.0246	0.0379	0.0345	0.0516

4.4 Shifted IRKA

An alternative approach to reduce unstable dynamical systems is to shift the unstable poles to the left of the imaginary axis [12]. This yields a stable system that can be reduced locally optimally in the $\mathcal{H}_2$ norm via IRKA. After obtaining the reduced model we shift the poles back. The success of this technique was very limited. If the magnitude of the unstable poles of the model varies, Shifted IRKA fails to approximate the original model accurately. However, if the real part of the unstable poles is between 0 and 0.001, the approximation obtained by Shifted IRKA is accurate. If we have a system where the unstable poles are between 0 and 0.001 and we shift them to the left by a number slightly larger than 0.001 we obtain the results shown in Table 33.

Table 33: $\mathcal{L}_2$ and $\mathcal{L}_\infty$error of the approximation by Shifted IRKA

r	8	10	12
$\mathcal{L}_2$ error	0.0559	0.0542	0.0355
$\mathcal{L}_\infty$ error	0.0315	0.0329	0.0265

After comparing the performance of Shifted IRKA with IRKA for unstable systems and $\mathcal{L}_2$ IRKA, we infer that for this system Shifted IRKA yields a better approximation than $\mathcal{L}_2$,

but it does not have any benefit against IRKA for unstable systems as we can see in Tables 34, 35, and 36.

Table 34: Comparison of Shifted IRKA, $\mathcal{L}_2$ IRKA, and IRKA for unstable systems ($r = 8$)

Technique	IRKAfUS	Shifted IRKA	$\mathcal{L}_2$ IRKA
$\mathcal{L}_2$ error	0.0433	0.0681	0.8161
$\mathcal{L}_\infty$ error	0.0015	0.0196	0.8760

Table 35: Comparison of Shifted IRKA, $\mathcal{L}_2$ IRKA, and IRKA for unstable systems ($r = 10$)

Technique	IRKAfUS	Shifted IRKA	$\mathcal{L}_2$ IRKA
$\mathcal{L}_2$ error	0.0435	0.0443	0.7532
$\mathcal{L}_\infty$ error	0.0015	0.0231	0.8091

Table 36: Comparison of Shifted IRKA, $\mathcal{L}_2$ IRKA, and IRKA for unstable systems($r = 12$)

Technique	IRKAfUS	Shifted IRKA	$\mathcal{L}_2$ IRKA
$\mathcal{L}_2$ error	0.0343	0.0516	0.8388
$\mathcal{L}_\infty$ error	8.3722e-04	0.0364	0.8980

We noticed that shifted IRKA depends on the magnitude of the shift. Let p denote the magnitude of the pole with the largest real part. We varied the shift from $p + 10^{-8}$ to $p + 10^{-3}$. The graphs below show that there is a shift value that yields the smallest error.

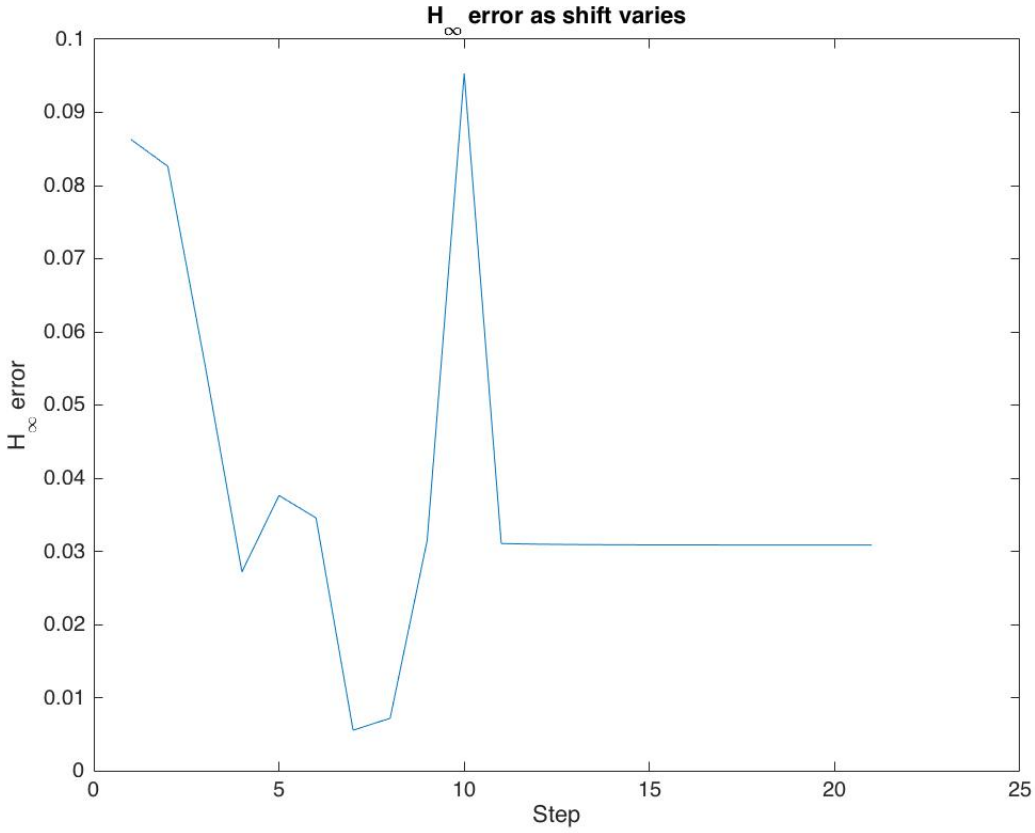

Figure 7: $\mathcal{H}_\infty$ error as the shift for Shifted IRKA varies

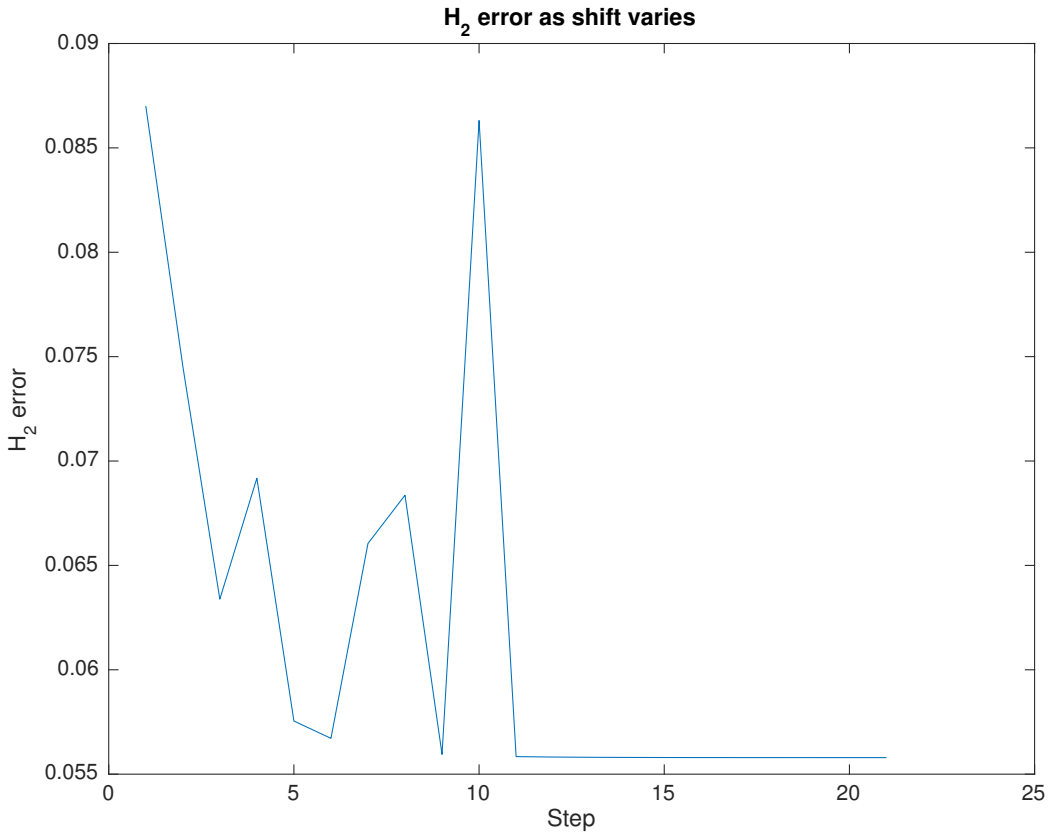

Figure 8: $\mathcal{H}_2$ error as the shift for Shifted IRKA varies

5 A Structure Preserving Algorithm for Dynamical Systems with Nonlinear Frequency Dependency

Interpolation-based methods such as IRKA require access to a standard first-order realization for the transfer function $\mathbf{H}(s)$. For this reason we cannot use the version of IRKA presented in section 2.3.1 to reduce systems for which we cannot obtain a standard frist-order realiza-

tion. Beattie and Gugercin introduced in [4] an implementation of IRKA that does not need access to any realization of the transfer function. This implementation requires only transfer function evaluations; hence, it is referred to as TF-IRKA or Realization-independent IRKA. Realization-independent IRKA yields an optimal reduced order model in the $\mathcal{H}_2$ norm; however, it does not preserve the structure of the original model. We introduce a structure-preserving algorithm which builds on TF-IRKA, and compare this algorithm with the optimal realization-independent IRKA. Again, for simplicity, we develop the framework for the SISO systems, but it can be extended to the MIMO case.

5.1 Loewner Matrix Approach for Interpolation and Realization-independent IRKA

As we have seen in section 2.3, if we have a dynamical system whose transfer function is given by a first-order realization $\mathbf{H}(s) = \mathbf{c}^T(s\mathbf{E}-\mathbf{A})^{-1}\mathbf{b}$, IRKA uses projection-based interpolatory model reduction to obtain $\mathbf{H}_r(s)$. For models whose transfer function is unknown or does not have a first-order realization, the Loewner matrix framework for interpolation is used to construct an $\mathbf{H}_r(s)$ that is a Hermite interpolant to $\mathbf{H}(s)$. The Loewner framework only requires knowledge of the transfer function $\mathbf{H}(s)$ and its derivative $\mathbf{H}'(s)$. Given a set of initial shifts $\{s_i\}_{i=1}^r$,

$$(\mathbf{E}_r)_{i,j} := \begin{cases} -\dfrac{(\mathbf{H}(s_i) - \mathbf{H}(s_j))}{s_i - s_j} & \text{if } i \neq j \\ -\mathbf{H}'(s_i) & \text{if } i = j \end{cases} \tag{5.1}$$

$$(\mathbf{A}_r)_{i,j} := \begin{cases} -\dfrac{(s_i\mathbf{H}(s_i) - s_j\mathbf{H}(s_j))}{s_i - s_j} & \text{if } i \neq j \\ -[s\mathbf{H}(s)]'|_{s=s_i} & \text{if } i = j \end{cases} \tag{5.2}$$

$$\mathbf{c}_r^T = [\mathbf{H}(s_1) \cdots \mathbf{H}(s_r)] \text{ and } \mathbf{b}_r = \begin{bmatrix} \mathbf{H}(s_1) \\ \vdots \\ \mathbf{H}(s_r) \end{bmatrix}. \tag{5.3}$$

The matrix $\mathbf{E}_r$ is known as the Loewner matrix and $\mathbf{A}_r$ is the shifted Loewner matrix. The transfer function $\mathbf{H}_r(s) = \mathbf{c}_r^T(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{b}_r$ is a Hermite interpolant of $\mathbf{H}(s)$. Realization-independent IRKA is based on the Loewner framework, and obviously it is an iterative method. At each iteration a transfer function $\mathbf{H}_r(s)$ is computed using (5.1) through (5.3). Below is a sketch of TF-IRKA.

Sketch of TF-IRKA

- Pick an r-fold intial shift set that is closed under conjugation.
- while (not converged)
  - Construct $\mathbf{A}_r, \mathbf{E}_r, \mathbf{b}_r, \mathbf{c}_r$ as in (5.1), (5.2), and (5.3)
  - Compute a pole-residue expansion of $\mathbf{H}_r(s)$:

$$\mathbf{H}_r(s) = \mathbf{c}_r^T(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{b}_r = \sum_{i=1}^{r} \frac{\phi_i}{s - \lambda_i}$$

  - $\sigma_i \leftarrow -\lambda_i$.
- Construct $\mathbf{A}_r, \mathbf{E}_r, \mathbf{b}_r, \mathbf{c}_r$ as in (5.1), (5.2), and (5.3).

Realization-independent IRKA converges to an $\mathcal{H}_2$-optimal reduced order model.

5.2 A Structure-preserving interpolation based algorithm

Even though TF-IRKA converges to an optimal reduced model in the $\mathcal{H}_2$-norm, it does not preserve the structure. Of course, we can reduce any system via interpolation, and preserve the structure. However, the reduced system obtained in this manner is not optimal. If we interpolated iteratively in order to preserve the structure, in general we would not be able to compute all the poles of the system. For example, consider a delay system. We could use TF-IRKA to obtain a reduced model that does not preserve the structure. We can also interpolate and obtain a reduced model that preserves the structure of the delay model [5]. For a reduced order delay model, we cannot compute all the system poles since there are infinitely many of them. This poses a problem with pole updating. Realization-independent IRKA is vital in overcoming this barrier. In this section we develop a structure-preserving algorithm based on TF-IRKA and IRKA. Given a set of initial shift $\{s_i\}_{i=1}^r$, and tangential directions $\{\mathbf{r}_i\}_{i=1}^r$ and $\{\mathbf{l}_i\}_{i=1}^r$, we can construct the projection matrices $\mathbf{V}$ and $\mathbf{W}$ in the same way we computed them in section 2.3.1. Using $\mathbf{V}$ and $\mathbf{W}$ we obtain reduced-size matrices corresponding to the full-size matrices in the transfer function of the original model. Then, using Realization-independent IRKA, we obtain a first-order realization of a reduced-order transfer function, for which we can compute the system poles. We use the system poles as interpolation points, and repeat the procedure. Below is a sketch of this structure-preserving algorithm we name Structure-preserving TF-IRKA for a second-order model whose transfer function is given by $\mathbf{H}(s) = \mathbf{c}^T(\mathbf{K}(s))^{-1}\mathbf{b}$.

Sketch of Structure-preserving TF-IRKA

- Pick an r-fold intial shift set selection that is closed under conjugation.
- while (not converged)

- Compute $\mathbf{V}$ and $\mathbf{W}$ the same way we compute them for IRKA
- Compute $\mathbf{K}_r(s) = \mathbf{W}^T\mathbf{K}(s)\mathbf{V}$, $\mathbf{B}_r = \mathbf{W}^T\mathbf{B}$, and $\mathbf{C}_r = \mathbf{CV}$
- Obtain $\mathbf{H}_r(s) = \mathbf{C}_r^T(s^2\mathbf{M}_r + s\mathbf{G}_r + \mathbf{K}_r)^{-1}\mathbf{B}_r$
- Compute $\mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r$ via TF-IRKA with input $\mathbf{H}_r(s)$
- Compute the eigenvalues λ_i of $\mathbf{A}_r$
- $\sigma_i \leftarrow -\lambda_i$

- Compute $\mathbf{K}_r(s) = \mathbf{W}^T\mathbf{K}(s)\mathbf{V}$, $\mathbf{B}_r = \mathbf{W}^T\mathbf{B}$, and $\mathbf{C}_r = \mathbf{CV}$

Next, we give a few numerical examples where we reduced the full model via Structure-preserving TF-IRKA. Since we cannot enforce stability in the intermediate steps while implementing Structure-preserving TF-IRKA with non-symmetric examples, for our examples we chose symmetric models.

5.3 Numerical examples: Reducing symmetric models via Structure-preserving TF-IRKA

In this section we present the results we obtain after reducing three symmetric systems via Structure-preserving TF-IRKA, and then compare the structure-preserving algorithm with TF-IRKA where possible.

5.3.1 Beam Model

The first system we consider is a second-order Cantilever Beam model with transfer function $\mathbf{H}(s) = \mathbf{c}^T(s^2\mathbf{M} + s\mathbf{G} + \mathbf{K})^{-1}\mathbf{b}$. Wyatt implemented a structure-preserving algorithm and

used it to reduce this Beam model in [21]. We reduced this model of size $n = 200$ with Structure-preserving TF-IRKA and TF-IRKA. The results are illustrated in the table and the graphs below.

Table 37: Structure-preserving TF-IRKA vs. TF-IRKA for Beam model

r	16	20	22
SP TF-IRKA $\mathcal{H}_\infty$ errorr	0.0023	0.0023	0.0021
TF-IRKA $\mathcal{H}_\infty$ error	5.1438×10^{-5}	3.1225×10^{-5}	3.1415×10^{-5}
SP TF-IRKA $\mathcal{H}_2$ error	0.0032	8.8064×10^{-4}	2.1533×10^{-4}
TF-IRKA $\mathcal{H}_2$ error	4.0076×10^{-5}	2.1640×10^{-5}	1.8514×10^{-5}

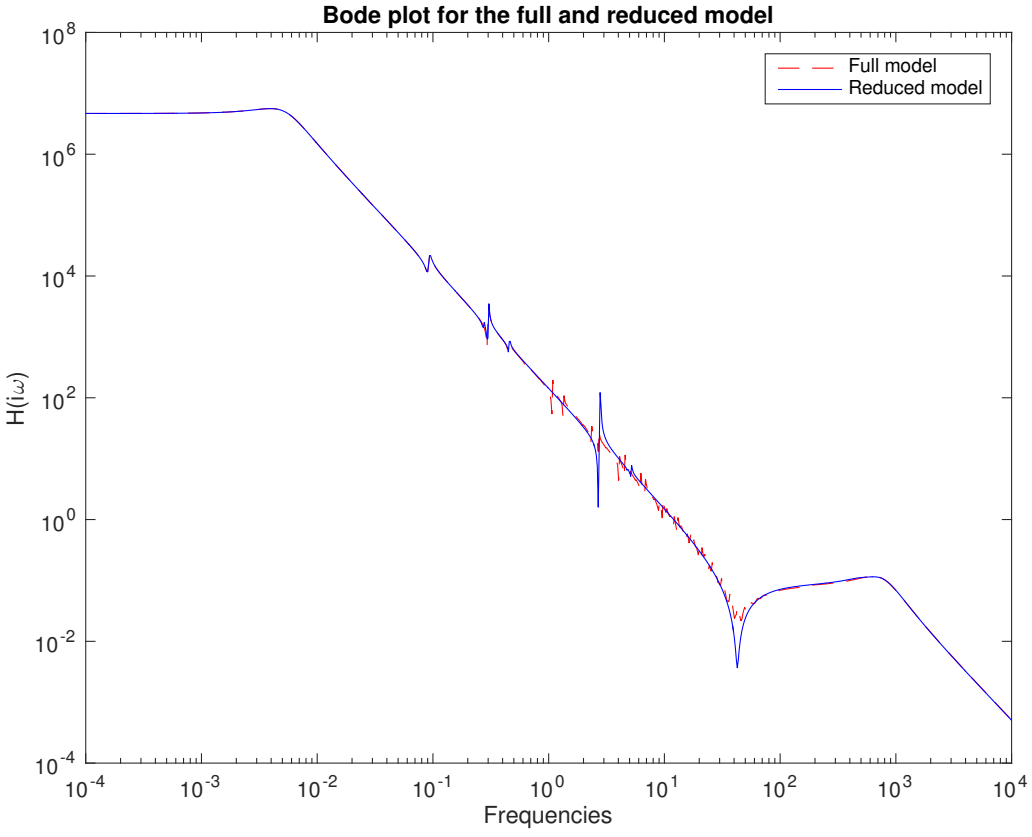

Figure 9: Beam model reduction with Structure-preserving TF-IRKA ($r = 22$)

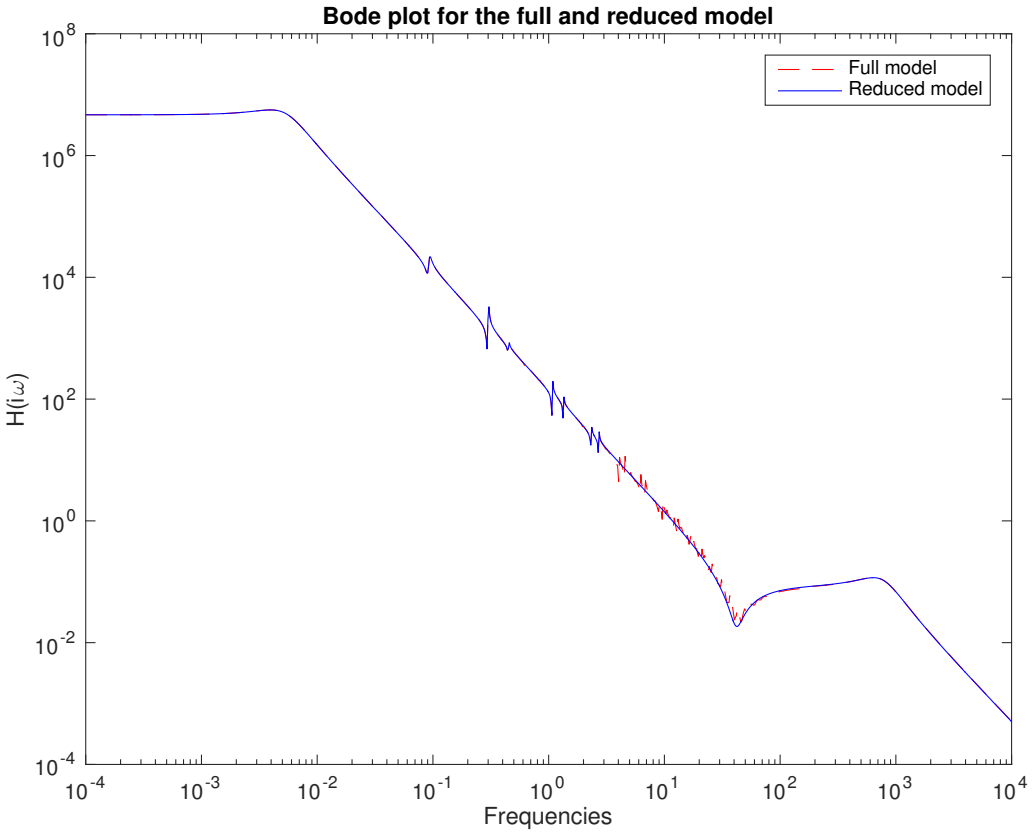

Figure 10: Beam model reduction with TF-IRKA ($r = 22$)

5.3.2 Hadeler Model

Second, we consider the Hadeler system which we obtained from the NLEVP package described in [7]. This model first was introduced as a nonlinear eigenvalue problem by Hadeler. The transfer function is given by $\mathbf{H}(s) = \mathbf{C}^T(e^s - 1)\mathbf{A}_2 + s^2\mathbf{A}_1 + \alpha\mathbf{A}_0)\mathbf{B}$ where $\mathbf{A}_2$, $\mathbf{A}_1$ are symmetric 100×100 matrices, $\mathbf{A}_0$ is the identity, and α is a scalar. In this case $\alpha = 1$.

For this model the structure-preserving algorithm seems to yield much better results than TF-IRKA. The results are illustrated below.

Table 38: Structure-preserving TF-IRKA vs. TF-IRKA for Hadeler model

r	8	10	12
SP TF-IRKA $\mathcal{H}_\infty$ errorr	0.0123	0.0166	0.0054
TF-IRKA $\mathcal{H}_\infty$ error	0.9587	0.9587	0.9595

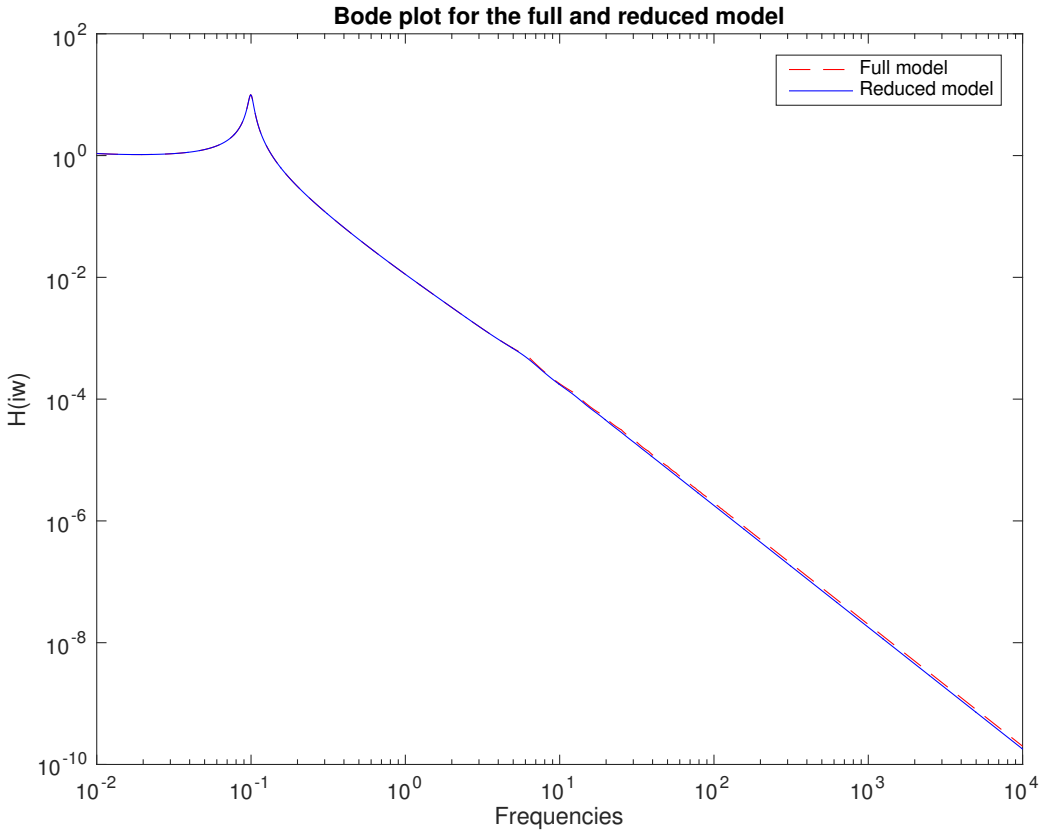

Figure 11: Hadeler model reduction with Structure-preserving TF-IRKA ($r = 12$)

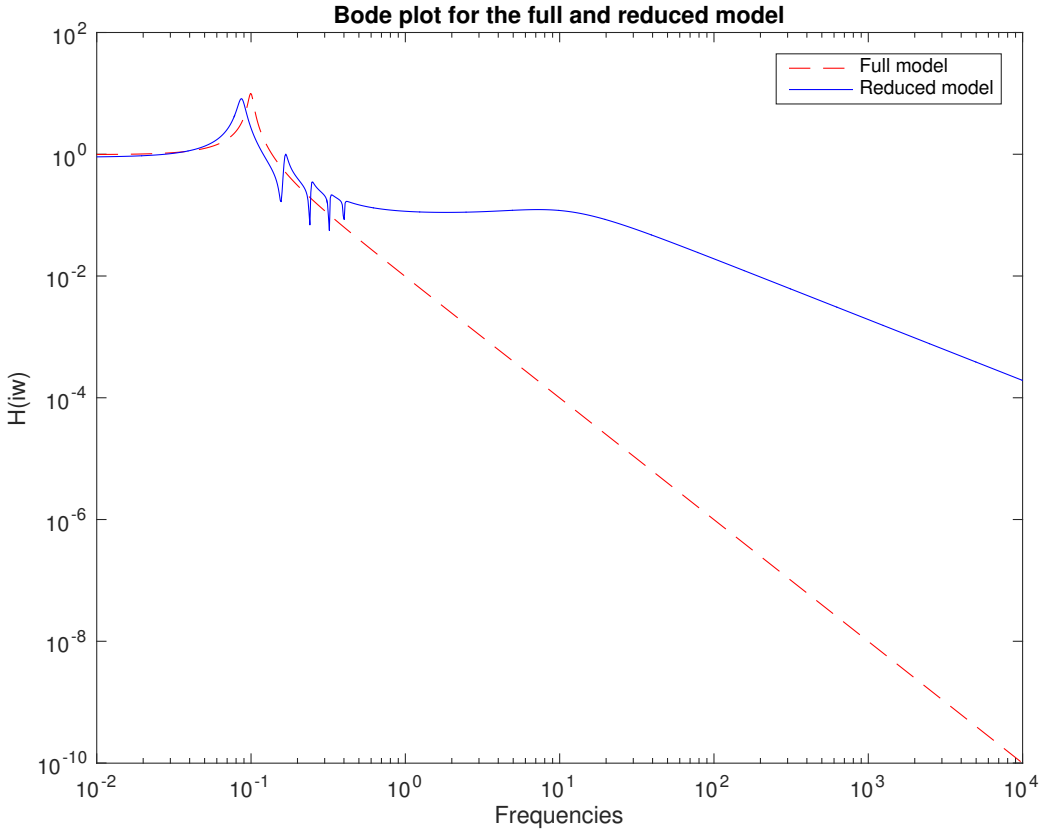

Figure 12: Hadeler model reduction with TF-IRKA ($r = 12$)

5.3.3 Loaded String Model

Finally, we consider a loaded string model with transfer function $\mathbf{H} = \mathbf{c}^T(\mathbf{A}_2 - s\mathbf{A}_1 + \frac{s}{s-\sigma}\mathbf{A}_0)^{-1}\mathbf{b}$ and size $n = 100$, also available in the NLEVP package. This model is obtained through a finite element discretization of a boundary problem which describes the vibration of a string to whom is attached an object of mass $m = 1$ by a spring with stiffness $k = 1$. For more information about this system, check [7].

Table 39: Structure-preserving TF-IRKA for loaded string model

r	8	9	10
SP TF-IRKA $\mathcal{H}_\infty$ errorr	0.0125	0.0123	0.0085

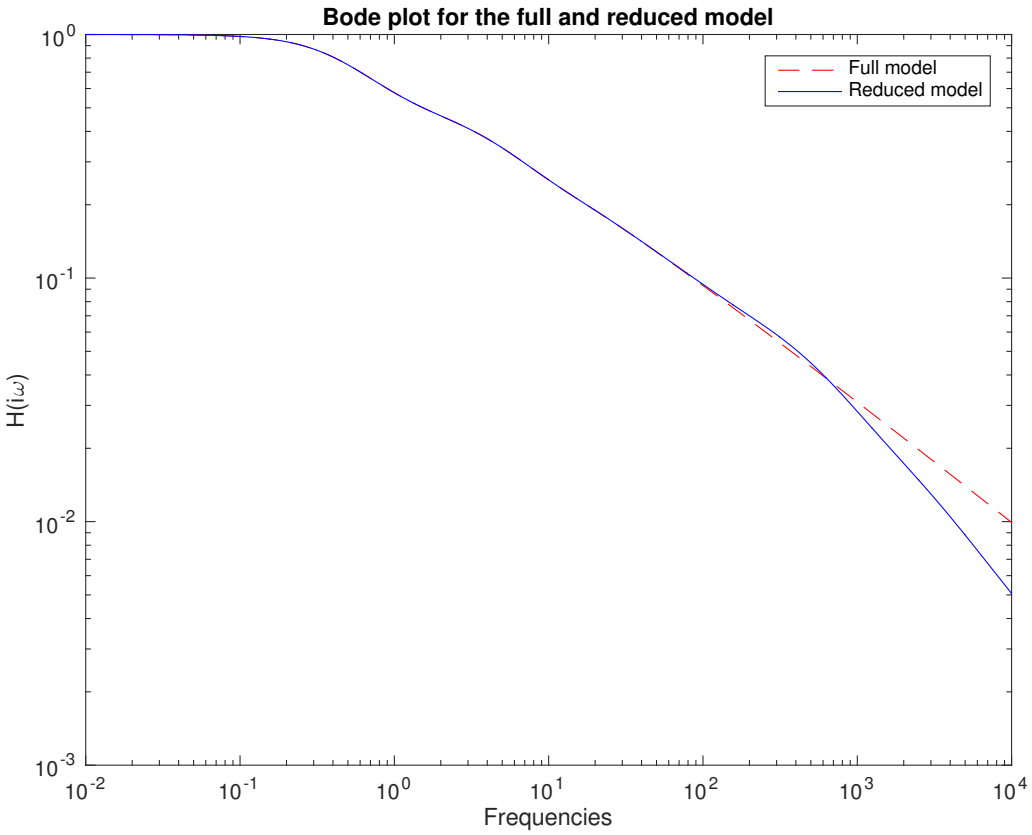

Figure 13: Loaded string model reduction with Structure-preserving TF-IRKA ($r = 10$)

For the loaded string model, the convergence of the structure-preserving algorithm is sensitive to the initialization. We could not reduce this model accurately with TF-IRKA.

6 Conclusions and Future Work

Model reduction techniques such as balanced truncation and IRKA are very reliable tools if we want to approximate large-scale linear stable dynamical systems with lower-order models. It is known that upon convergence, IRKA produces a locally $\mathcal{H}_2$-optimal reduced order model if the original system is stable and linear. For unstable systems, we cannot guarantee this. However, we notice that if we reduce an unstable system via IRKA, there are no obstacles from an algorithmic perspective. Furthermore, if the unstable system had only a few unstable poles, IRKA captures these poles. Also, IRKA's performance is comparable with $\mathcal{L}_2$ IRKA. In this paper, we also considered systems whose frequency dependency is nonlinear. Realization-independent IRKA can be used to reduce such systems, but it cannot preserve the structure. In this paper, we introduced a structure-preserving algorithm which performed really well with symmetric systems. For certain models, such as the Hadeler and the Loaded String model, the structure-preserving TF-IRKA yielded better results than the optimal TF-IRKA. In the future, we could investigate the connection between these problems and the eigenvalue problems. Also, in the intermediate step of Structure-preserving TF-IRKA, we encountered issues regarding stability. Thus, researching how to enforce stability in the intermediate step of Stucture-preserving TF-IRKA, is important if we want to implement the algorithm with non-symmetric systems.

References

[1] S. Gugercin and C. Beattie, *Model Reduction by Rational Interpolation*. Model Reduction and Approximation for Complex Systems, 2014.

[2] C. Magruder, C. Beattie, and S. Gugercin, *Rational Krylov methods for optimal $\mathcal{L}_2$ model reduction*. 49th IEEE Conference on Decision and Control, Atlanta, GA 2010.

[3] A. Varga and B. D. O.Anderson, *Accuracy-enhancing methods for balancingrelated frequency-weighted model and controller reduction.* Automatica, 39(5), 919–927, 2003

[4] C. Beattie and S. Gugercin *Realization-independent $\mathcal{H}_2$-approximation*. 51st IEEE Conference on Decision and Control, Maui, HI 2012.

[5] C. Beattie and S. Gugercin, *Interpolatory projection methods for structure-preserving model reduction*. Systems and Control Letters 58, 225-232, 2009.

[6] K. Zhou, G. Salomon, and E. Wu, *Balanced Realization and Model Reduction for Unstable Systems*. Int. J. Robust Nonlinear Control 9, 183-198, 1999.

[7] T. Betcke, N. J. Higham, V. Mehrmann, C. Shröder, and Françoise Tisseur, *NLEVP: A Collection of Nonlinear Eigenvalue Problems*. Manchester Institute for Mathematical Sciences, 2011.

[8] A.J. Mayo and A.C. Antoulas, *A framework for the solution of the generalized realization problem*. Linear Algebra and Its Applications, problem. Linear Algebra and Its Applications, 425(2-3), 634–662, 2007.

[9] C. T. Mullis and R. A. Roberts, *IEEE Transactions on Circuits and Systems* 551–561,1976.

[10] A. C. Antoulas, *Approximation of large-scale dynamical systems (Advances in Design and Control)*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2005.

[11] K. Glover, *All optimal Hankel-norm approximations of linear multivariable systems and their L^∞-error bounds*, Internat. J. Control, 39 (6), 1115– 1193, 1984.

[12] D. Kubalinska, *Optimal interpolation-based model reduction*, Bremen University, 2009.

[13] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge Monographs on Mechanics. Cambridge University Press, Cambridge, 1996.

[14] S. Gugercin, A. C. Antoulas, and C. Beattie, $\mathcal{H}_2$ *Model Reduction for Large-scale Linear Dynamical Systems*. Siam J. Matrix Anal. Appl. 30 (2), 609–638 2008.

[15] B. Anič, C. Beattie, S. Gugercin, and A. C. Antoulas, *Interpolatory weighted-$\mathcal{H}_2$ model reduction*. Automatica 49, 1275–1280, 2013.

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

[17] J.T. Borggaard, S. Gugercin, *Model Reduction for DAEs with an Application to Flow Control*. Active Flow and Combustion Control 2014, R. King editors, Springer-Verlag, Notes on Numerical Fluid Mechanics and Multidisciplinary Design, 127, 381-396, 2015.

[18] L. Meier and D.G. Luenberger, *Approximation of Linear Constant Systems*, IEE. Trans. Automat. Contr., 12, 585-588, 1967.

[19] R.H. Bartels and GW Stewart, *Algorithm 432: Solution of the matrix equation AX+ XB= C*. Communications of the ACM, 15(9), 820–826, 1972.

[20] K. Jbilou, *ADI preconditioned Krylov methods for large Lyapunov matrix equations.* Linear Algebra and its Applications 432, 2473–2485 (2010).

[21] S. Wyatt, *Issues in Interpolatory Model Reduction: Inexact Solves, Second- order Systems and DAEs.* PhD Dissertation, Virginia Tech, 2012.

[22] G. Flagg, C.A. Beattie and S. Gugercin, *Convergence of the Iterative Rational Krylov Algorithm*, Systems and Control Letters, 61(6), 688-691, 2012.