A Shifting Method for Dynamic System Model Order Reduction

by

Xu, Song

B.S., Department of Automation
Tsinghua University, 2006

Submitted to the School of Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Computation for Design and Optimization

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2007

Author .....
School of Engineering
August 15, 2007

Certified by .....
Jacob K. White
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Jaime Peraire
Professor of Aeronautics and Astronautics
Codirector, Computation for Design and Optimization Program

A Shifting Method for Dynamic System Model Order Reduction

by

Xu, Song

Submitted to the School of Engineering
on August 15, 2007, in partial fulfillment of the
requirements for the degree of
Master of Science in Computation for Design and Optimization

Abstract

Model Order Reduction (MOR) is becoming increasingly important in computational applications. At the same time, the need for more comprehensive models of systems is generating problems with increasing numbers of outputs and inputs. Classical methods, which were developed for Single-Input Single-Output (SISO) systems, generate reduced models that are too computationally inefficient for large Multiple-Input Multiple-Output (MIMO) systems.

Although many approaches exclusively designed for MIMO systems have emerged during the past decade, they cannot satisfy the overall needs for maintaining the characteristics of systems. This research investigates the reasons for the poor performances of the proposed approaches, using specific examples.

Inspired by these existing methods, this research develops a novel way to extract information from MIMO systems, by means of system transfer functions. The approach, called Shifting method, iteratively extracts time-constant shifts from the system and splits the transfer function into several simple systems referred to as contour terms that outline the system structure, and a reducible system referred to as remainder system that complement the Contour Terms. This algorithm produces a remainder system that existing approaches can reduce more effectively. This approach works particularly well for systems with either tightly clustered or well separated modes, and all the operations are $O(n)$. The choice of shifts is based on an optimization process, with Chebyshev Polynomial roots as initial guesses. This paper concludes with a demonstration of the procedure as well as related error and stability analysis.

Thesis Supervisor: Jacob K. White
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

The work of this thesis couldn't have been possible without the help and support of many people. My foremost and most sincere gratitude goes to my thesis advisor, Jacob White. He has been an extraordinary source of ideas as well as an excellent mentor. From the beginning of my time at MIT, he has been supporting and encouraging me, particularly on my idea of shifts and splits. Without his constant and insightful advices, this thesis wouldn't have been this smooth. As a persistent, rigorous and intelligent scholar, he has set an example for me to learn from.

The Computational Prototyping Group has created an interactive environment for discussion, for which I owe many thanks. Dimitry Vasilyev has offered to join my dicussion and set forth many valuable suggestions. Although the proposed joint discussion wasn't be able to be set up due to my limited time at MIT, I have still been grateful for the suggestions on dynamic system simulation and the kindness. Bradley Bond, during one talk with me, suggested that a specific system should be defined in order to guarantee the accuracy, which was an important source for me to build up my approach. Lei Zhang, my senior in CDO, has given me continuous help in my studies and research. I would also to thank other members in this group.

Without the help of CDO staff members, I couldn't have proceeded smoothly since I came to MIT. Thanks for the care and the activities organized.

Last but not least, infinite thanks should go to my parents and my wife, who have been consistently supporting me and providing me with valuable suggestions beyond the sea. Their encouragements and urgence are great stimulation for me to march on all the way here.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Model Order Reduction

Existing physical and engineered systems can be described by mathematical models useful for analysis, the most common being a dynamic system. An example of a physical system model is the Storm Water Management Model (SWMM), a dynamic rainfall-runoff simulation model used for single event or long-term (continuous) simulation of runoff quantity and quality from primarily urban areas [1]. An example of an engineered system model is the interconnected circuit used to design electronic systems. As these models became more complex, techniques are needed to reduce that complexity in a reliable fashion. [2].

For simulation purposes, dynamic systems are primarily discretized and transformed into discrete Ordinary or Partial Differential Equations (ODEs or PDEs), for instance the simulation of RC circuits. For control purposes, controllers are often designed according to and depending on original systems, in order to adjust the system's characteristics. A proportional-integral-derivative controller (PID controller) is effective and most common in vast applications. It is noticeable that either the simulation or the control process is directly affected by the complexity of the original systems. In other words, given a similar structure, it is more expensive to simulate or design a controller for a more complex system.

However, the need for increasing accuracy is driving research in generating more

and more complex models, and this poses a problem. The computational cost of simulating ever more complicated system is not adequately compensated by the increasing speed of computers. Therefore, it has been increasingly important to develop methods for computing smaller-sized approximations to original systems, in order to reduce computational complexity while retaining best accuracy. This technique has been generally named Model Order Reduction (MOR). It should also be noted that other than simplifying the original, parellelization is also an option given a computer cluster without sacrificing accuracy as MOR does. In this paper, we only focus on MOR.

1.2 Single-Input Single-Output Systems MOR

So far, the most prevalent MOR approaches are for or at least more effective for Single-Input Single-Output systems (SISOs).

1.2.1 Linear Systems

Until present, most of the efforts have been focused on developing MOR algorithms for linear SISO systems. Among them there are SVD-based methods including Hankel norm approximants [3], Truncated Balanced Realization (TBR) [4], Proper Orthogonal Decomposition (or Karhunen Loeve expansion) [5, 6] and [7], Krylov-based methods including Krylov subspace projections [8, 9] as well as SVD-Krylov mixed type [2].

1.2.2 Nonlinear Systems

For nonlinear SISO systems, MOR techniques include methods based on linearization or bilinearization of the initial system around the equilibrium point [10]-[12], methods of balanced truncation [13, 14], and algorithms using Proper Orthogonal Decomposition [15, 16]. Moreover, recently methods such as Trajectory Piecewise-Linear approximations [17] have emerged.

1.3 Multiple-Input Multiple-Output Systems MOR

In a vast array of applications, systems also contain multiple inputs and outputs, or terminals, where the established methods mentioned above become inefficient or even inapplicable. This incapability is due to the inherent property of MIMO systems: because we cannot make the assumption that the interaction between any pair of terminals is magnitude-wise insignificant, the transfer function matrix will be fully populated. Past research has developed an incisive and effective SVDMOR method [18], which explores the correlation between pairs of terminals and applies MIMO MOR algorithms [19]-[23] to the extracted system.

Unfortunately, none of methods mentioned above performs satisfactorily when the number of terminals is on the same numerical scale as the total number of states. The SISO methods will fail because they rely on the fact that the number of terminals is much smaller than that of states; the SVDMOR, which is based on the DC response, generates a reduced model that has disordered high-frequency responses and has unfruitful performances when the system is well-conditioned.

Aiming at these drawbacks, the research work below explores the inner connections inside a system and develops a new method based on SVDMOR. The proposed method iteratively extracts information while improving the condition for MOR on each iteration. Moreover, this method also guarantees accurate high-frequency responses and better medium-frequency characteristics approximation.

1.4 Notations

This thesis uses the standard form for a following **Continuous Linear Time-Invariant (LTI) dynamic system**, which is a typical representation of control systems:

$$\begin{cases} E_1\dot{x} = A_1x + B_1u, \\ y = C_1x + D_1u, \end{cases} \tag{1.1}$$

Here, the inputs are denoted as $u \in \mathbb{R}^p$, the states are denoted as $x \in \mathbb{R}^n$, and the outputs are denoted as $y \in \mathbb{R}^q$. We only focus on the system where E matrix is non-singular. Therefore, defining $A = E_1^{-1}A_1$, $B = E_1^{-1}B_1$, $C = C_1$, and $D = D_1$, the system can be rewritten in normal form as

$$\begin{cases} \dot{x} = & Ax + Bu, \\ y = & Cx + Du, \end{cases} \tag{1.2}$$

where the **system matrix** $A \in \mathbb{R}^{n\times n}$, the **input matrix** $B \in \mathbb{R}^{n\times p}$, the **output matrix** $C \in \mathbb{R}^{q\times n}$, and the **direct matrix** $D \in \mathbb{R}^{q\times p}$. This normal form representation will be used in the following.

The Laplace Transform of (1.2) yields the transfer function

$$G(s) = C(sI - A)^{-1}B + D, \tag{1.3}$$

where s is the Laplacian operator. We employ the notation

$$\tilde{G}(s) \approx G(s)$$

where $\tilde{G}(s)$ is the reduced model that approximates $G(s)$.

A special case of (1.2), used below, is

$$\begin{cases} \dot{x} = Ax + bI_{n\times n}u \\ y = cI_{n\times n}x \end{cases}, \tag{1.4}$$

where A is symmetric and $b, c \in \mathbb{R}$. This prototype will be studied in the next few chapters. The reason for introducing this special case is that the singular values of a system's transfer function, which will be crucial for MOR, is determined by A alone. The advantage of such a property will be explained in the following chapters.

1.5 Outline

This thesis is organized in the following way:

Chapter 2 gives the results generated by applying SVDMOR to a two-dimensional grid circuits and offers the reasons for poor performances.

Chapter 3 describes a single step of the shifting method, and justifies the method by examining both low and high frequency effects.

Chapter 4 introduces the iterative version of the implementation.

Chapter 5 explains some implementation issues and computational complexity concerns.

Chapter 6 shows the results of applications to circuits with different conditions.

Chapter 7 provides conclusions and suggestions.

Chapter 2

Terminals Extraction

2.1 Motivation

As stated in Chapter 1, MIMO MOR algorithms include [19]-[23]. Regardless of which algorithm is applied to a system, the computational complexity is determined by the dimensions of the system. In particular, it is related to the size of the transfer function of the system. For example, the computational complexity of reducing the system (1.2) is determined by the size of its transfer function (1.3): $p \times q$.

Therefore, it will be more efficient if MOR can be applied to a smaller system extracted from the original one. One option is to find a proper subspace and use this subspace to represent the system's characteristics. This approach is the most prevalent method for system simplifications.

2.2 SVDMOR Algorithm

Proposed in [24, 18], the SVDMOR algorithm is to extract information from the frequency response matrix of a system. This process is based on Singular Value Decomposition (SVD) low-rank approximations [25], which is carried out as follows.

Theorem 2.2.1. *Given matrix $A \in \mathbb{R}^{m\times n}$, if σ_i is the ith singular value of matrix A, u_i and v_i are associated left and right singular vectors, and $\sigma_1 \geq \sigma_2 \geq \sigma_{\min(m,n)}$, then*

the best rank $r(r \leq \min(m,n))$ *approximation of* A *in the 2-norm and the Frobenius norm is*

$$A_r = \sum_{i=1}^{r} \sigma_i u_i v_i' = U_r \Sigma_r V_r', \tag{2.1}$$

where $||A - A_r||_2 = \sigma_{r+1}$ *and* $||A - A_r||_F = \sqrt{\sigma_{r+1} + \cdots + \sigma_{\min(m,n)}}$.

Table 2.1 provides an overview of the SVDMOR algorithm.

Table 2.1: The SVDMOR algorithm

Given a dynamic system, as shown in (1.2):
ϵ: Tolerance for reducing singular values;
r: Number of remaining singular values;
1.Compute one matrix M[1] that reveals correlations.
2.Apply Low-Rank Approximation using ϵ: $M = U\Sigma V \approx U_r \Sigma_r V_r$.
3.Find weights using Pseudoinverse: $b = BU_r(U_r'U_r)^{-1}, c = (V_r'V_r)V_r'C$.
4.Form the transfer function: $G(s) \approx U_r c(sI + A)^{-1} b V_r'$.
5.Apply MIMO MOR techniques [19]-[23] to $G_r(s) = c(sI + A)^{-1}b$, which is an $r \times r$ matrix transfer function, and obtain $\tilde{G}_r(s)$.
6.Recover the system transfer function $G(s) \approx U_r \tilde{G}_r(s) V_r'$, the number of states being reduced from n to r.

This technique is based on the idea that the original system, $G(s)$, can be spanned by a smaller system, $G_r(s)$. Then the more computationally-inexpensive $\tilde{G}_r(s)$ can be obtained by applying MOR to $G_r(s)$, as explained below:

1. The Laplace transform of the transfer function is a matrix of rational expressions represented by the Laplacian operator s, which cannot be directly processed in many Linear Algebra manipulations. Therefore, a matrix comprising only

[1]The M matrix is often called Correlation Matrix.

numbers must be used instead. These matrices include

$$\text{DC Response: } G_{DC} = -CA^{-1}B \tag{2.2}$$

$$\text{The First Moment of the Response: } G_1 = CA^{-1}A^{-1}B \tag{2.3}$$

$$\text{Frequency Shifted Moments: } G_{s_0} = C(s_0 I - A)^{-1}B \tag{2.4}$$

The 'moments' are usually referred to as the coefficients of the Taylor series expansion, with respect to some point, for example, 0 in DC response matrix (2.2) or frequency shift s_0 in (2.3) and (2.4) above.

By constructing these matrices, we expect that they can, mostly, reveal the correlation between terminals.

2. B and C are approximately linear combinations of U_r and V_r, respectively. Pseudoinverse [25] is used to solve

$$bV_r' \approx B \tag{2.5}$$

$$U_r c \approx C, \tag{2.6}$$

as in Step 3, leaving the resulting system $G_r(s)$ with dimensions $r \times r$.

3. In this thesis, the SVDMOR algorithm is performed through the Lanczos process with deflation and look-ahead [26].

2.3 Drawbacks

Due to the fact that SVDMOR relies on low-rank approximation, this approach requires the singular values of correlation matrix to be distinguishable. In other words, some of the singular values must be relatively larger than others so that only a part will be kept. However, in many practical cases, the singular values are so close that SVDMOR will not eliminate any states. For example, in a system where B and C are identity matrices and A is diagonal with approximately the same entries like 2, the

singular values are approximately 0.5. In this case, unless a high threshold is allowed, no states will be deleted.

Another example is as follows:

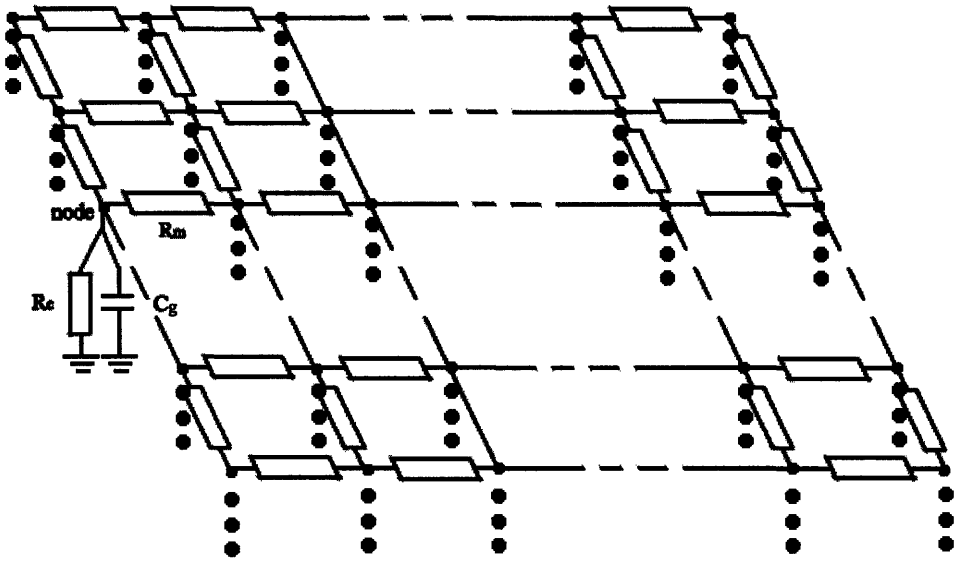

Figure 2-1: Structure of the Two-Dimensional RC Circuit

The objective is a $n \times n$ two-dimensional Resistor-Capacitor (RC) circuit. It contains n^2 nodes, each node connecting with its neighbouring node with a coupling resistor R_m and connecting with parellel capacitor C_g and ground resistor R_c to the ground. The three black dots connected to each node denote the parellel capacitor and ground resistor.

This is a typical example in simulation that can be formulated as an ODE, see [27]. Using the same technique as in equation (1.2), the simulation of this problem is formulated as a first order ordinary differential equation, which is a dynamic system. The input is a vector of currents into each node and the output is a vector of voltages on each node, both with length n^2. The states are the same as the output, therefore the system is at order-n^2 and $C = I_{n^2 \times n^2}$.

By specifying $n = 10$, $C_g = 20pF$, $R_m = 100\Omega$ and $R_c = 100\Omega$, we reach a system with $B = C = I_{100\times100}$, $D = 0$ and A whose eigenvalues are distributed in

$[-3.95e+6, -5.00e+4]$. If SVDMOR is applied, the singular values of $CA^{-1}B$ should be noted, which is shown below (the X-axis is the value and Y-axis is used to line the singular values):

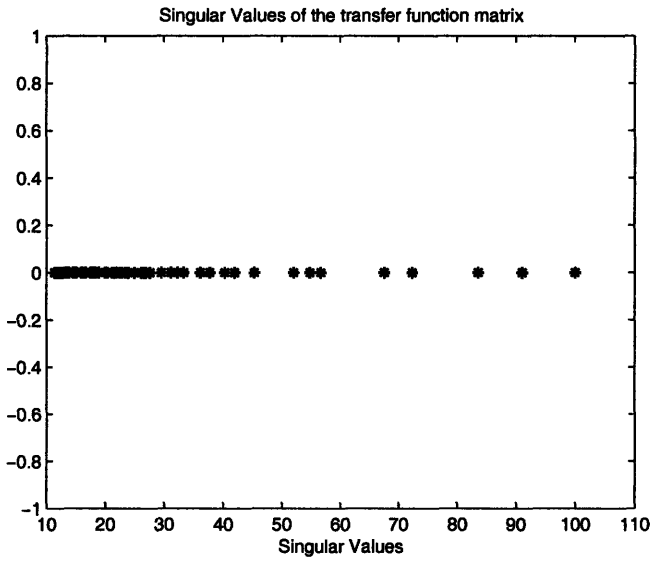

Figure 2-2: Singular Values of the transfer function matrix in 2-D circuit

The singular values are distributed between 11.36 and 100. This means that if the 5% threshold is chosen, no states can be eliminated. Therefore, the natural system configurations has hindered SVDMOR to work efficiently.

On the other hand, the SVDMOR has generated a model with disordered high frequency responses, which is shown in the following sections.

2.4 Simulation Results

Suppose we have the system

$$\begin{aligned} \dot{\bar{x}} &= \begin{bmatrix} -1.01 & -1 \\ -1 & -1.01 \end{bmatrix} \bar{x} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \bar{u} \\ \bar{y} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \bar{x} \end{aligned} \tag{2.7}$$

The transfer function, by means of Eigenvalue Decomposition, is

$$\begin{aligned} G(s) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} (s \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -1.01 & -1 \\ -1 & -1.01 \end{bmatrix})^{-1} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= (s \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} -0.01 & 0 \\ 0 & -2.01 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^{-1})^{-1} \\ &= \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{s+0.01} & 0 \\ 0 & \frac{1}{s+2.01} \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^{-1}. \end{aligned} \tag{2.8}$$

Now we apply the SVDMOR to this system (2.7) as follows:

1. Choose the DC response matrix (s=0) as the correlation matrix and extract its low-rank approximant:

$$\begin{aligned} G_{DC} &= \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{0.01} & 0 \\ 0 & \frac{1}{2.01} \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^{-1} \\ &\approx \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{0.01} \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}^{-1} \\ &= U_r \Sigma_r V_r'. \end{aligned} \tag{2.9}$$

2. Calculate weights:

$$\begin{aligned} b &= \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^T \\ c &= \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^T. \end{aligned}$$

3. Form the approximated transfer function:

$$G(s) \approx U_r c(sI + A)^{-1} b V_r'$$

$$= \begin{bmatrix} -\frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{2}}{2} & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{s+0.01} & 0 \\ 0 & \frac{1}{s+2.01} \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{2}}{2} & 0 \end{bmatrix}^T$$

$$= \begin{bmatrix} -\frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{2}}{2} & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{s+0.01} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{2}}{2} & 0 \end{bmatrix}^T.$$

4. Apply MOR techniques to this result and obtain the following transfer function:

$$\begin{aligned} \tilde{G}(s) &= \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} \left[s + 0.01 \right]^{-1} \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}^T \\ &= \tilde{B}(sI - \tilde{A})^{-1}\tilde{C}. \end{aligned} \tag{2.10}$$

It appears that we have successfully reduced one state, leaving the reduced system little changed from the original one. However, if we compare the frequency responses of the two systems, we observe disordered results in high frequency.

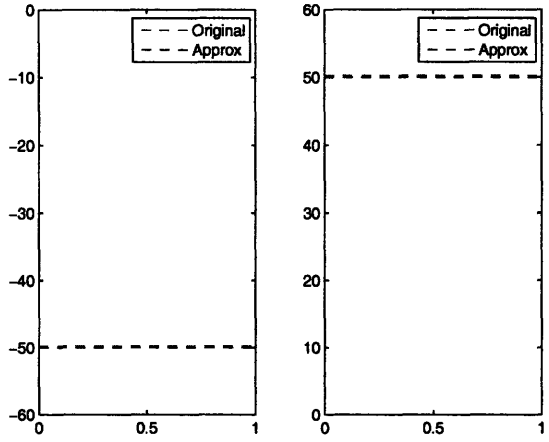

Figure 2-3: DC Responses ($\omega = 0$) Comparison Between the Original and Reduced System (SVDMOR, tol=5×10^-2)

Figures 2-3 and 2-4 show the comparisons of the responses in the two outputs, between the original and the approximated system, with inputs $u = (0, e^{i\omega})$ (X-axis: time, Y-axis: value of response). It is demonstrated that in high frequency, the

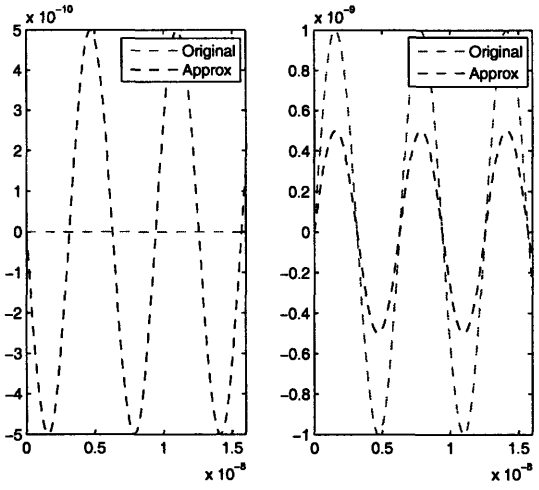

Figure 2-4: High-Frequency Responses ($\omega = 1 \times 10^6$) Comparison Between the Original and Reduced System (SVDMOR, tol=5×10^{-2})

approximated system exhibits a chaotic response, especially with a larger number of terminals.

2.5 Diagnostic Analysis

2.5.1 Error plot

To further explore the chaotic phenomenon presented in section 2.4, we plot the relative errors versus frequencies. The relative error referred to here is defined as follows:

$$e_r(\omega) = \frac{||M(\omega) - M_r(\omega)||_2}{||M(\omega)||_2}, \quad M(\omega) = G(j\omega). \tag{2.11}$$

Figure 2-5 shows the relative errors with respect to frequencies ranging from 10^{-10} to 10^{10} in semi-logarithm axes. It is clear that as frequency increases, the original and approximated systems have unrelated responses that lead to the relative error 1.

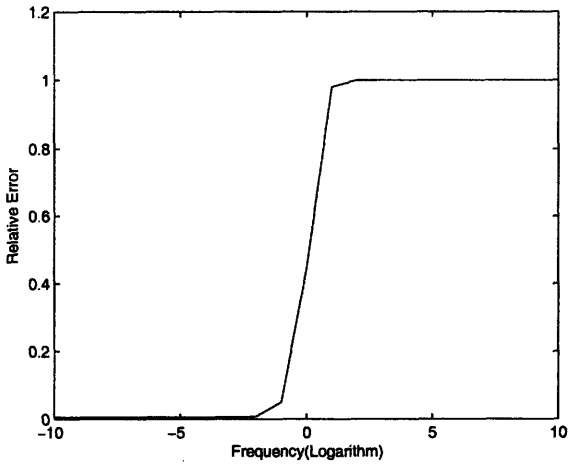

Figure 2-5: Relative Error (2.11) vs. Frequency ($10 \times 10^{-10} - 10 \times 10^{10}$)

2.5.2 SVD-based Analysis

We begin this analysis by comparing the two transfer functions (2.8) and (2.10):

$$
\begin{aligned}
G(s) &= \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{s+0.01} & 0 \\ 0 & \frac{1}{s+2.01} \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^{-1}, \\
\tilde{G}(s) &= \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} \left[s + 0.01 \right]^{-1} \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}^{T} \\
&= \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{s+0.01} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^{-1}.
\end{aligned}
$$

These are exactly the SVD decompositions of the systems. The error between the systems, which is also in SVD form, is

$$
e(s) = G(s) - \tilde{G}(s) = \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{s+2.01} \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}^{-1}. \tag{2.12}
$$

Then the relative error[2] is

$$e'(s) = \frac{||e(s)||_2}{||G(s)||_2} = \left|\frac{s+0.01}{s+2.01}\right|, \tag{2.13}$$

$$e_r(\omega) = \left|\frac{j\omega+0.01}{j\omega+2.01}\right|. \tag{2.14}$$

It should be guaranteed that this value should not be excessively large when frequency varies. However, easily,

$$\lim_{\omega\to\infty}\left|\frac{j\omega+0.01}{j\omega+2.01}\right| = 1.$$

The limit means that as frequency increases, the error is so large that the state kept at first is not informative enough to span the whole system. This drawback does not change if the correlation matrix in Step 1, in Table 2.1, is otherwise selected. This intrinsic high-frequency error is due to the fact that as frequency changes, the relations between different singular values will also change and finally breach the conditions used to reduce states. For example, in this case, the ratio between two singular values is $\frac{2.01}{0.01}$ at 0 frequency and near 1 at high frequency. In other words, the states considered small enough to be deleted at low frequency are not trivial at high frequency, thus cannot be eliminated.

2.5.3 Global Error

Other than the observations above, it is also noticeable that the frequency response decays with the frequency, or [3]

$$||G(s)||_2 \propto \frac{1}{s}. \tag{2.15}$$

This relation means that the DC response has the largest magnitude among all the frequency responses, gaining the most importance. With respect to the DC response,

[2]The 2-norm of a matrix is its largest singular value, as in [25].

[3]This is an approximation and is more accurate as $s = j\omega$ becomes large in magnitude.

we re-evaluated the difference versus frequency between the two systems:

$$e_r^{(2)} = \frac{||e(s)||_2}{||G(0)||_2}. \tag{2.16}$$

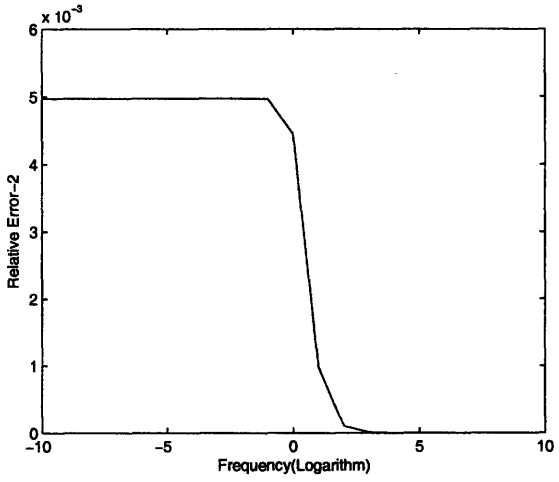

Figure 2-6: Global Error (2.16) vs. Frequency ($10 \times 10^{-10} - 10 \times 10^{10}$)

Figure 2-6 shows that the maximum error is approximately 5×10^{-3} through out the whole range of frequencies. In a separate experiment, it was demonstrated that if we impose white noise signals[4] as inputs, the error will remain below the defined threshold, which is satisfactory. Based on the high probability that an input signal may be random, the generated reduced model is accurate in this sense.

2.6 Conclusions

The SVDMOR method allows us to conclude the following:

1. The SVDMOR uses SVD to extract 'important' information from the 'trivial.' Then MOR is applied to the 'important' system. The generated reduced model

[4]The white noise signal is one of the most important test signals in System Identification, to capture system characteristics.

of this 'important' system is then used to span the whole system. This approach is insightful because it decouples the problem on a basis of a correlation matrix.

2. In many practical cases, SVDMOR method, based on low-rank approximation, cannot perform efficiently due to the close distribution of the singular values of the correlation matrix.

3. The SVDMOR generates an inaccurate, or disordered, model, especially at high frequencies, because the relations between singular values of the system matrix change as frequency varies. The feasibility of keeping only a portion of states is not guaranteed through out the whole range of frequencies.

4. However, the method guarantees global accuracy. By global, we mean the error with respect to the zero, or DC, response.

5. The SVD low rank approximation is the best approximation in the 2-norm, as shown in Theorem 2.2.1. This fact reminds us that this problem may not be easily settled if we still only transform the original system into another simple system.

Chapter 3

The Shifting Approach

We have seen that the proposed SVDMOR method has unignorable drawbacks. The conclusions drawn from Chapter 2 cast some illumination on system simplifications, which suggest that the original system may be transformed into more than one systems.

3.1 Eigenvalues and Modes

In control theory, for the system (1.2), stability is related with the system matrix A, or more specifically, the eigenvalues of A. Generally, it is known that the system is asymptotically ω-stable if and only if for all eigenvalues λ of A: $\mathrm{Re}(\lambda) < 0$ [28]. Moreover, many of a system's other characteristics are also associated with these eigenvalues. The reason is as follows:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \rightarrow \begin{cases} \dot{x} = Q\Lambda Q^{-1}x + Bu \\ y = Cx + Du \end{cases} \rightarrow \begin{cases} Q^{-1}\dot{x} = \Lambda Q^{-1}x + Q^{-1}Bu \\ y = CQQ^{-1}x + Du \end{cases},$$

where $A = Q\Lambda Q^{-1}$ is the eigenvalue decomposition and the diagonal entries of Λ are the eigenvalues of A. For $X = Q^{-1}x$, the system is decoupled to

$$\begin{cases} \dot{X} = \Lambda X + Q^{-1}Bu, \\ y = CQX + Du. \end{cases} \tag{3.1}$$

By solving the first equation of (3.1) evolving in time, it is seen that

$$X = C_X e^{\Lambda t} X_0 + D_X u$$

is represented as a linear combination of $e^{\Lambda t}$, where $C_X \in \mathbb{R}^{n \times n}, D_X \in \mathbb{R}^{n \times p}$ and

$$\begin{aligned} e^{\Lambda t} &= e^{\text{diag}(\lambda_1 t, \lambda_2 t, \cdots, \lambda_n t)} \\ &= \text{diag}(e^{\lambda_1 t}, e^{\lambda_2 t}, \cdots, e^{\lambda_n t}). \end{aligned}$$

These $e^{\lambda_i t}$'s are called the modes of the system, which determine the responses of the original system. In a stable system, all the eigenvalues of the system matrix must contain negative real parts; the modes represented by a smaller eigenvalue in magnitude is called slow decaying modes, and fast decaying modes otherwise.

In this thesis, we only deal with stable systems, which means that all eigenvalues of A only contain negative real parts.

3.2 System Split with a Single Shift

3.2.1 Motivation

The SVDMOR algorithm fails to work as an accurate and effective approach, but it does reveal an important perspective of generating reduced model. The research will be the modification based on this approach.

For the system (1.4), the transfer function is

$$G(s) = (Is - A)^{-1}.$$

On the one hand, as mentioned in section 2.5.2, in high frequency, $G(s) \approx (Is)^{-1}$. This is the reason for the tremendous error in high frequency using SVDMOR. This phenomenon reminds us that it would be more reasonable to define or extract some simple portion that can maintain high frequency accuracy.

On the other hand, even the DC response[1], which is $G(0) = A^{-1}$, is sometimes 'ill-conditioned' but very simple. By 'ill-conditioned', we mean the singular values of a matrix being not favorable for SVDMOR. For example a matrix whose largest singular value is 10 and smallest one is 5 cannot be reduced as otherwise any eliminated terminal will generate at least a 50% error. However, there are matrices such as the identity or matrices with clustered eigenvalues that can hopefully be described with less information than the whole-matrix-size storage. For instance, only 1 and the dimension n will be needed to record the identity matrix.

Note that $G(0) = A^{-1} = Q\Lambda^{-1}Q^{-1} = Q(-\Lambda^{-1}(-Q^{-1}))$. If A is symmetric and negative definite, this form is an SVD[2]. This is an explanation for the choice of the model in (1.4), as the singular values of $G(0)$ are analytically determined by A's eigenvalues, or more specifically the minus reciprocal of the eigenvalues.

3.2.2 The Algorithm

Basically, this approach is to extract a simple system from the original one and hence to hopefully readjust the system more favored by MOR, which is reflected as the Remainder system. The term singular values referred to here, meaning the singular values of the DC response matrix of the system, will be mentioned from time to time as it is a determining factor of the SVDMOR. We will derive the algorithm for the general model.

We split the original model into two parts, as shown in Figure 3-1:

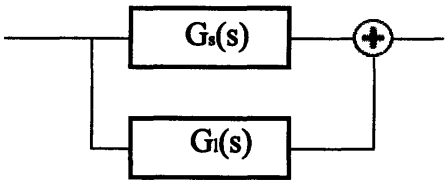

Figure 3-1: Split Model

[1]DC response is the correlation matrix mentioned in Table 2.1, and will be always referred to this way in the following chapters except that it is used for system verification.

[2]For a symmetric matrix, the eigenvectors can be chosen orthonormal.

We have[3]

$$\begin{aligned}
G(s) &= C(sI-A)^{-1}B = C(sI-Q\Lambda Q^{-1})B \\
&= C(Q(sI-\Lambda)Q^{-1})^{-1}B = CQ(sI-\Lambda)^{-1}Q^{-1}B \\
&= CQ\left[\frac{1}{s-\lambda_i}\right]_i Q^{-1}B \\
&= CQ\left[\frac{1}{s-\lambda'}\right]_i Q^{-1}B + CQ\left[\frac{1}{s-\lambda_i}-\frac{1}{s-\lambda'}\right]_i Q^{-1}B \\
&= C(\frac{1}{s-\lambda'})B + CQ\left[\frac{\lambda_i-\lambda'}{(s-\lambda_i)(s-\lambda')}\right]_i Q^{-1}B \\
&= \frac{1}{s-\lambda'}(CB) + \frac{1}{s-\lambda'}(CQ\left[\frac{\lambda_i-\lambda'}{s-\lambda_i}\right]_i Q^{-1}B) \\
&= G_s(s) + G_l(s)
\end{aligned}$$

After the linear algebra operations, the original system is transformed into a simple shifted system $G_s(s)$ and a higher-order remainder system $G_l(s)$. To simulate the system $G_s(s)$ is no more expensive because the system is decoupled and does not involve denser matrices. If we can reduce the remainder term $G_l(s)$ to certain extent that the overall simulation requires less computation, the result can be advantageous over the original one. For the remainder term, in SVDMOR

$$\begin{aligned}
H_{DC} &= C_l A_l^{-1} B_l \\
&= U\Sigma V \approx U_r \Sigma_r V_r,
\end{aligned}$$

and in MOR part

$$\begin{aligned}
H(s) &\approx U_r \underbrace{b_c(sI-A_l)^{-1}b_b}_{H_r(s)} V_r^T \\
&= U_r H_r(s) V_r^T \\
&\approx U_r \tilde{H}_r(s) V_r^T,
\end{aligned}$$

[3]We use the notation $[a_i]_i$ to denote a diagonal matrix, whose ith diagonal is a_i.

therefore

$$G(s) = G_s(s) + \frac{1}{s - \lambda'}(U_r \tilde{H}_r(s) V_r^T).$$

It is noticed that compared to the original system, there is a scaling factor $[\lambda_i - \lambda']_i$ that changes the distribution and mutual relations of the singular values of H_{DC}. In particular, for the system (1.4), the singular values are changed from $\left[\frac{1}{-\lambda_i}\right]_i$ to $\left[\left|\frac{\lambda_i - \lambda'}{-\lambda_i}\right|\right]_i \times \frac{1}{\lambda'}$. The shift rescales each of the original system's singular value with different factors and scales the whole system by $\frac{1}{\lambda'}$, in the hope that more states can be eliminated out of the remainder system than of the original one.

After defining the shifted system, the approximated model maintains high frequency accuracy, please refer to section 4.4 for details. Table 3.1 is the brief description of the algorithm:

Table 3.1: Single step shifting approach

Eigenvalue decompose A, get Q and Λ
choose λ' to be the shift[4]
% **Generate the shifted system**
$A_s = [\lambda']_i, B_s = B$
$C_s = C, D_s = D$
% **Generate the Remainder system**
$A_l = [\lambda_i]_i, B_l = Q^{-1}B$
$C_l = CQ\,[\lambda_i - \lambda']_i, D_l = D$
% **Obtain reduced model**
apply SVDMOR to the Remainder system

3.3 Examples

The following examples demonstrate the advantage of the shifting idea.

[4]The choice of shifts in discussed in 5

1. For a system (1.4), A is a symmetric dense matrix with eigenvalues uniformly distributed in $[-10, -1]$. If only SVDMOR is applied, no states will be deleted using tolerance 5%, as the singular values of its DC response matrix are in $[0.1, 1]$. However, by selecing $\lambda' = -8$ as the shift, the resulting remainder system has a better singular distribution, as shown in Figure 3-2

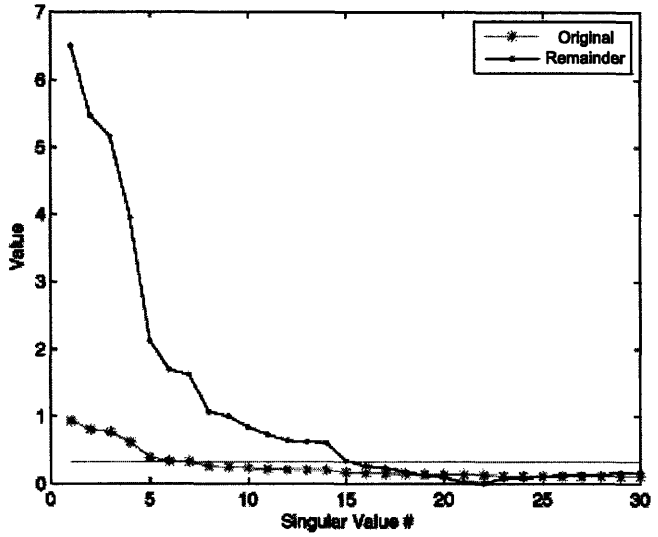

Figure 3-2: Change of Singular Values with a Single Shift

The dots and the stars are the singular values in the original and remainder systems. The straight line denotes 5% of the largest singular value in the remainder system. After the shift, 15 states can be eliminated out of a total 30, and the first 4 singular values are scaled much more than the others.

A shift can make some modes more significant than others in the remainder system.

2. For a system (1.4), A is a symmetric dense matrix with eigenvalues uniformly distributed in $[-2.05, -1.95]$. If only SVDMOR is applied, no states will be deleted using tolerance 5%, as the singular values of its DC response matrix are in $[0.488, 0.513]$. In this system the eigenvalues or the modes are clustered closely around -2. Therefore, if -2 is chosen as the shift, the simple system

extracted by the shift is accurate enough to represent the whole system and the remainder system is deleted completely.

The shifting idea can be more efficient if there are tightly clustered eigenvalues.

3.4 Conclusions

1. The shifting idea is to extract a simple system with identical modes and hence make the remainder system more favorable to be reduced than the original one.

2. A shift can either rescale some modes with larger amplifying factors than the others in the remainder system, or capture a cluster of eigenvalues and hence in both ways reduce states in the system.

3. The approximated system using the shifting idea maintains high frequency accuracy.

Chapter 4

Successive Extractions

As mentioned in the chapter before, the shifting idea is to extract a simple system with identical modes from the original and make the remainder system more favorable to be reduced than the original one. More often, more than one shift should be carried out to the original system.

4.1 System Splitting with Multiple Shifts

4.1.1 Motivation

For a system (1.4), A is a symmetric dense matrix with eigenvalues half -1 and half -2. If only SVDMOR is applied, no states will be deleted using tolerance 5%, as the singular values of its DC response matrix are in $[0.5, 1]$. Even by a single shift -1, we only could eliminate half the states. However, in this case, the remainder, which contains only -2 as eigenvalues, is still simple enough to be shifted again. This is the primitive motivation for the multiple shifts method, which is successive shifting of the original system.

4.1.2 Algorithm

After applying a single shift, a new system configuration (A_l, B_l, C_l, D_l) is set up as the remainder system, as in Table 3.1. In the approach described in this chapter,

another single shift will be applied to the remainder system without it being reduced, generating a new remainder system. This procedure is repeated for a certain number of times d, which is the number predefined by experiments and observation of the system's characteristics. Actually, the MOR part will not be performed until all shifts have been picked out.

The derivation with details is as follows (d shifts and λ^j denotes the jth shift):

$$
\begin{aligned}
G(s) &= C(Is-A)^{-1}B = CQ\left[\frac{1}{s-\lambda_i}\right]_i Q^{-1}B \\
&= \frac{1}{s-\lambda^{(1)}}(CB + CQ\left[\frac{\lambda_i-\lambda^{(1)}}{s-\lambda_i}\right]_i Q^{-1}B) \\
&= \frac{1}{s-\lambda^{(1)}}(CB + \frac{1}{s-\lambda^{(2)}}(CQ\left[\lambda_i-\lambda^{(1)}\right]_i Q^{-1}B \\
&\quad + CQ\left[\frac{(\lambda_i-\lambda^{(1)})(\lambda_i-\lambda^{(2)})}{s-\lambda_i}\right]_i Q^{-1}B)) \\
&= \frac{1}{s-\lambda^{(1)}}(CB + \frac{1}{s-\lambda^{(2)}}(CQ\left[\lambda_i-\lambda^{(1)}\right]_i Q^{-1}B \\
&\quad + \frac{1}{s-\lambda^{(3)}}(CQ\left[(\lambda_i-\lambda^{(1)})(\lambda_i-\lambda^{(2)})\right]_i Q^{-1}B \\
&\quad + CQ\left[\frac{(\lambda_i-\lambda^{(1)})(\lambda_i-\lambda^{(2)})(\lambda_i-\lambda^{(3)})}{s-\lambda_i}\right]_i Q^{-1}B))) \\
&= \cdots \\
&= \frac{CB}{s-\lambda^{(1)}} + \frac{CQ\left[\lambda_i-\lambda^{(1)}\right]_i Q^{-1}B}{(s-\lambda^{(1)})(s-\lambda^{(2)})} + \frac{CQ\left[(\lambda_i-\lambda^{(1)})(\lambda_i-\lambda^{(2)})\right]_i Q^{-1}B}{(s-\lambda^{(1)})(s-\lambda^{(2)})(s-\lambda^{(3)})} \\
&\quad + \cdots + \frac{CQ\left[\prod_{k=1}^{d-1}(\lambda_i-\lambda^{(k)})\right]_i Q^{-1}B}{\prod_{j=1}^{d}(s-\lambda^{(j)})} + \frac{1}{\prod_{j=1}^{d}(s-\lambda^{(j)})} CQ\left[\frac{\prod_{k=1}^{d}(\lambda_i-\lambda^{(k)})}{(s-\lambda_i)}\right]_i Q^{-1}B \\
&= \sum_{m=1}^{d} \frac{CQ\left[\prod_{k=1}^{m-1}(\lambda_i-\lambda^{(k)})\right]_i Q^{-1}B}{\prod_{j=1}^{m}(s-\lambda^{(j)})} + \frac{1}{\prod_{j=1}^{d}(s-\lambda^{(j)})} CQ\left[\frac{\prod_{k=1}^{d}(\lambda_i-\lambda^{(k)})}{(s-\lambda_i)}\right]_i Q^{-1}B \\
&= CQ\sum_{m=1}^{d} \frac{\left[\prod_{k=1}^{m-1}(\lambda_i-\lambda^{(k)})\right]_i}{\prod_{j=1}^{m}(s-\lambda^{(j)})} Q^{-1}B + \frac{1}{\prod_{j=1}^{d}(s-\lambda^{(j)})} CQ\left[\frac{\prod_{k=1}^{d}(\lambda_i-\lambda^{(k)})}{(s-\lambda_i)}\right]_i Q^{-1}B.
\end{aligned}
$$

With d shifts picked out of the original system, the remainder system is organized in a way that the singular values are rescaled by different factors successively. By carefully selecting these shifts, this approach will exhibit similar but more stronger efficiency than a single shift. The choice of shifts will be discussed in section 5.

In the derivation, the terms

$$G_{CT}(s) = \sum_{m=1}^{d} G_m(s) = CQ \sum_{m=1}^{d} \frac{\left[\prod_{k=1}^{m-1}(\lambda_i - \lambda^{(k)})\right]_i}{\prod_{j=1}^{m}(s - \lambda^{(j)})} Q^{-1}B \tag{4.1}$$

are called the **Contour Terms**, as they basically outline the characteristics of the system. $G_m(s)$ is a m-order ODE system. The term

$$G_{RS}(s) = \frac{1}{\prod_{j=1}^{d}(s - \lambda^{(j)})} CQ \left[\frac{\prod_{k=1}^{d}(\lambda_i - \lambda^{(k)})}{(s - \lambda_i)}\right]_i Q^{-1}B \tag{4.2}$$

is defined as the **Remainder**. The Remainder is a $(d+1)$-order ODE system. This term supplements some important properties that the Contour Terms are not able to fully represent. The full algorithm description is discussed in section 5.2.

4.2 MOR of the Remainder

The Remainder is a $(d+1)$-order ODE system, but there is a d-order scalar system in front. Therefore, for the MOR part, only a first order system is reduced.

Generate Remainder using equation (4.2):

$$\begin{aligned}
A_l &= [\lambda_i]_i, \\
B_l &= Q^{-1}B, \\
C_l &= CQ \left[\prod_{k=1}^{d}(\lambda_i - \lambda^{(k)})\right]_i, \\
D_l &= D.
\end{aligned}$$

Then

$$G_{RS}(s) = \frac{1}{\prod_{j=1}^{d}(s - \lambda^{(j)})} (C_l(sI - A_l)^{-1}B_l + D_l).$$

SVDMOR, as decribed in Table 2.1, is applied to $C_l(sI - A_l)^{-1}B_l + D_l$ obtain a

reduced model $C_r(sI - A_r)^{-1}B_r + D_r$. Therefore

$$\tilde{G}_{RS}(s) = \frac{1}{\prod_{j=1}^{d}(s - \lambda^{(j)})}(C_r(sI - A_r)^{-1}B_r + D_r), \tag{4.3}$$

where $A_r \in \mathbb{R}^{r\times r}$, $B \in \mathbb{R}^{r\times p}$, $C \in \mathbb{R}^{q\times r}$, and $D \in \mathbb{R}^{q\times p}$.

4.3 Examples

The following example shows how the singular values of the Remainder term in each step are rescaled to obtain a final Remainder more suitable to be reduced.

For a system (1.4), A is a symmetric dense matrix with eigenvalues uniformly distributed in $[-10, -1]$. -10, -6.55 and -3.1 are chosen as shifts, though it is not guaranteed that these are the optimal choice. The following is a set of plots after each shift, indicating the change of singular values of the remainder system.

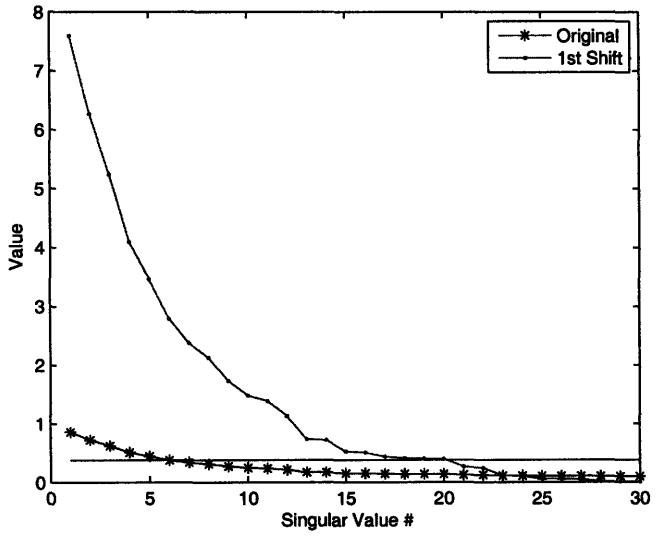

Figure 4-1: Change of Singular Values after the First Shift

Figures 4-1, 4-2 and 4-3 show the change of singular values of the remainder systems (excluding the scaling factor $\frac{1}{\prod_{j=1}^{d}(s-\lambda^{(j)})}$), and we can see that 20, 12 and 6

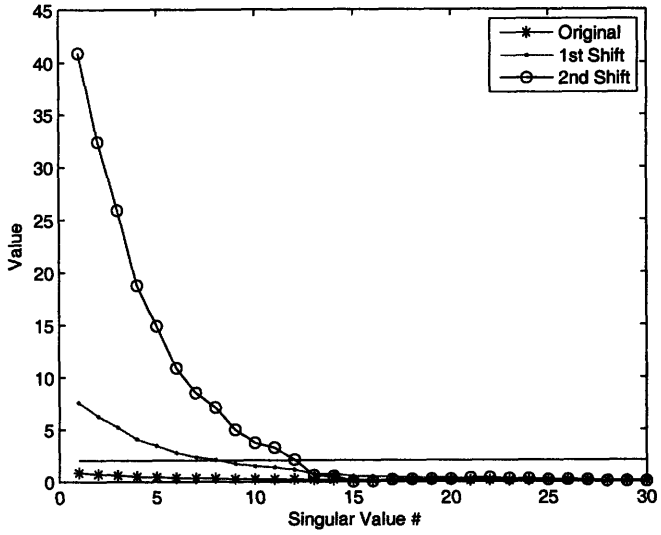

Figure 4-2: Change of Singular Values after the Second Shift

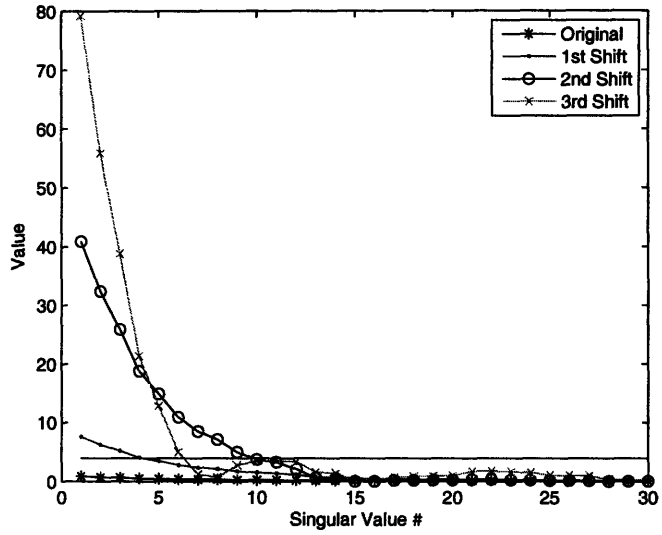

Figure 4-3: Change of Singular Values with after the Third Shift

singular values are kept with 5% as the threshold. It looks like that in each step, the whole curve is broken off around the shift point, the part containing the slow-decaying modes is lift up to a steeper slope and the other part is planished. With all shifts applied, the whole curve is twisted such that only a few modes outside the range of shifts are lifted to a very steep slope and all the other modes are reduced to be trivial enough to eliminate.

On the other hand, for the example mentioned in section 4.1.1, only two shifts -1 and -2 will be needed to represent the system. Similar to a single shift, the shifts can either rescale some modes more than others in the remainder system, or capture a cluster of eigenvalues then the remainder is so trivial that it can be completely eliminated.

4.4 Error Analysis

4.4.1 Relative 2-norm Error

The Relative 2-norm Error refers to the relative error between DC response matrices of the approximated and of the original system.

$$e_2 = \frac{||G_{CT}(0) + \tilde{G}_{RS}(0) - G(0)||_2}{||G(0)||_2} \tag{4.4}$$

Because MOR is based on SVD, this approach naturally guarantees low error, in the DC response matrix's 2-norm sense. In this thesis, experiments are performed with reference to this error. This is the error to estimate during model reduction.

4.4.2 Frequency Response 2-norm Error

Section 2.5 reveals disordered high frequency response of the approximated system, if only SVDMOR is applied. However, after shifts, high frequency response of the

approximant

$$G_{CT}(s) = \sum_{j=1}^{d} G_j(s), \tag{4.5}$$

as in (6.1), is

$$\lim_{s \to i\infty} G_{CT}(s) = \lim_{s \to i\infty} \sum_{j=1}^{d} G_j(s) \tag{4.6}$$

$$= \lim_{s \to i\infty} G_1(s), \tag{4.7}$$

and the Remainder is in higher order so that can be ignored in high frequency. Therefore, $\tilde{G}(s) \propto \frac{CB}{s}$, the same as the original system.

More, optimistically, in system (1.4) the original system's 2-norm, which is the largest singular value (λ_1 has the least magnitude), is

$$|\frac{1}{s - \lambda_1}|.$$

For any given frequency, if r states are kept out of the remainder, the 2-norm error between the original and the approximated system is the difference between the unreduced and reduced Remainder.

$$e_2 = \frac{||G_{CT}(0) + \tilde{G}_{RS}(0) - G(0)||_2}{||G(0)||_2} \tag{4.8}$$

$$= \frac{||\tilde{G}_{RS}(0) - G_{RS}(0)||_2}{||G(0)||_2}. \tag{4.9}$$

After MOR, the singular values of the difference in the numerator of (4.9) are

$$|\frac{\prod_{k=1}^{n}(\lambda_i - \lambda^{(k)})}{(s - \lambda_i)\prod_{j=1}^{n}(s - \lambda^{(j)})}| \quad i = 1, 2 \ldots, n - r$$

Supposedly any of them can be the largest singular value. For each,

$$\begin{aligned} e_2 &= \|\frac{\frac{\prod_{k=1}^{n}(\lambda_i-\lambda^{(k)})}{(s-\lambda_i)\prod_{j=1}^{n}(s-\lambda^{(j)})}}{\frac{1}{s-\lambda_1}}\|_2 = \|\frac{(s-\lambda_1)\prod_{k=1}^{n}(\lambda_i-\lambda^{(k)})}{(s-\lambda_i)\prod_{j=1}^{n}(s-\lambda^{(j)})}\|_2 \\ &\leq \|\frac{\prod_{k=1}^{n}(\lambda_i-\lambda^{(k)})}{\prod_{j=1}^{n}(s-\lambda^{(j)})}\|_2 \leq \|\frac{\prod_{k=1}^{n}(\lambda_i-\lambda^{(k)})}{\prod_{j=1}^{n}(0-\lambda^{(j)})}\|_2 = \|\frac{P(\lambda_i)}{P(0)}\|_2 \\ &\leq \frac{\max_{x\in[\lambda_1,\lambda_{n-r}]}|P(x)|}{|P(0)|}. \end{aligned}$$

If Chebyshev roots are chosen be the shifts, as in Figure 5-1, the value of $|P(0)|$ can be substantially larger than that of $\max_{x\in[\lambda_1,\lambda_{n-r}]}|P(x)|$, and the error is guaranteed to be a small number.

When C and B are not scaled identities, the problem is more complicated and not resolved in this research. Singular values are targeted in MOR. However, the choice of shifts only guarantees the change of eigenvalues, eigenvalues are not singular values and how these eigenvalues affect the singular values of $G_{RS}(s)$ is unknown, for reasons stated in 3.2.1. In fact, no relation is ascertained between eigenvalues and singular values, if B's and C's are allowed to change randomly. In practice, it is hoped that most probably C and B do not change the significance of the scaled eigenvalues, for example, a very large rescaled eigenvalue $|\frac{\prod_{k=1}^{n}(\lambda_i-\lambda^{(k)})}{(s-\lambda_i)\prod_{j=1}^{n}(s-\lambda^{(j)})}|$ does not take overwhelmingly small portion in the Remainder System. For example, the following example will not be considered as typical:

$$\begin{aligned} G_l(0) &= C_l A_l^{-1} B_l \\ &= \begin{bmatrix} 10^{-5} & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} 10^{+5} & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \end{aligned}$$

Although A^{-1} has a very favourable structure according to our reasoning, the structures of B and C spoil the advantageous position of the first eigenvalue.

4.4.3 Point-to-Point Error

Sometimes the 2-norm error is not strict enough to guarantee an accurate approximation and it is suggested to observe the point-to-point transfer function value in DC frequency. By point-to-point transfer function, we mean each entry of the transfer function. For example, the ith row jth column entry denotes the response of jth input to ith output. To observe the transfer function individually will allow us to know more about this approach. Normally, we only study a specific jth column denoting the interactions of all nodes with the jth input.

4.5 Conclusions

1. Successive extractions or multiple shifts is an extension of the single shifting approach, in particular that only one MOR is performed after all shifts are carried out.

2. Instead of one shift, multiple shifts will cooperatively reorganize the singular value distribution so that as few states as possible can be kept. This is done through making some modes sufficiently larger than others, or extracting the Contour Terms sufficiently more significant than the Remainder.

3. After shifts, the approximant remains accurate high-frequency response. In a specific system 1.4, the approximated system guarantees accurate responses at all frequencies.

Chapter 5

The Choice of Shifts

In Chapter 4, the single shift idea is extended to the multiple case and a simple example has illustrated the advantage of multiple shifts. Some implementation issues are discussed in this chapter.

A basic principle of choosing shifts is that these selected significant shifts should represent the whole system's characteristics as much as possible. The significance here is discussed in the scope of the DC response matrix of a dynamic system.

At this point, two kinds of systems the multiple-shift method favors should be pointed out. They are already illustrated in section 3.3. One is a system with well separated modes, and some mode can be sufficiently larger than others; the other one is a system with tightly clustered modes, or modes in a sufficiently narrow range. To deal with these types requires different philosophies: for the former one, as some modes are larger than others, we will further amplify these modes more while less to others, and normally MOR to this kind ends up with the slowest modes; however, for the latter, as the modes do not differ as significantly, they can be represented altogether as only a small number of shifts because of their similarity between each other, and it is often possible to eliminate all states in the Remainder. However, both philosophies reflect the same requirements of system configuration: in both kinds, there should be obvious significant shifts. More than often, a system will not be as simple as either of them, but a mixed type. Then we can think of it as a combination of subsystems, and deal with them according to the philosophies above. In this

thesis, typical systems referred to above will be referred to as **Type 1** and **Type 2** respectively.

As a result of the difference of philosophies, the type that is between these two is possibly intractable.

It is worthing pointing out that if a system contains a few modes relatively separated from others, the modes can be considered as candidates for shifts. For example, in the system where the eigenvalues are -2 and in $[-11, -10]$, -2 can be chosen as a shift.

5.1 Based on Chebyshev Polynomials

In this section, we focus on system (1.4) with concentrated modes. Take a look at how the Remainder is rearranged compared with the original system:

$$G_{RS}(s) = \frac{1}{\prod_{j=1}^{d}(s-\lambda^{(j)})} CQ \begin{bmatrix} \frac{\prod_{k=1}^{d}(\lambda_1-\lambda^{(k)})}{(s-\lambda_1)} & & & \\ & \frac{\prod_{k=1}^{d}(\lambda_2-\lambda^{(k)})}{(s-\lambda_2)} & & \\ & & \ddots & \\ & & & \frac{\prod_{k=1}^{d}(\lambda_n-\lambda^{(k)})}{(s-\lambda_n)} \end{bmatrix} Q^{-1}B, \tag{5.1}$$

$$G(s) = CQ \begin{bmatrix} \frac{1}{(s-\lambda_1)} & & & \\ & \frac{1}{(s-\lambda_2)} & & \\ & & \ddots & \\ & & & \frac{1}{(s-\lambda_n)} \end{bmatrix} Q^{-1}B. \tag{5.2}$$

Other than the scalar scaling the whole system, which can be ignored for a moment as it does not change the ratios between modes, each mode $\frac{1}{s-\lambda_i}$ is scaled differently by factor $\prod_{k=1}^{d}(\lambda_i - \lambda^{(k)})$.

Define

$$P(x) = \prod_{k=1}^{d}(x - \lambda^{(k)}). \tag{5.3}$$

It is a d-order polynomial with shifts $\lambda^{(k)}$ as zeros. Therefore, each mode is scaled by $P(\lambda_i)$. For the first kind of system, the idea that fast decaying modes far from the origin need to be scaled less than the modes closer to the origin is similar to the convergence analysis of the Generalized Conjugate Residual algorithm (GCR) or the Conjugate Gradient algorithm (CG) for solving linear equations [25]. The analysis is performed through the introduction of Chebyshev Polynomials.

Definition 5.1.1. The Chebyshev polynomials of the first kind are defined by the recurrence relation:

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_2(x) &= 2xT_n(x) - T_{n-1}(x). \end{aligned}$$

Or explicitly:

$$T_n(x) = \begin{cases} \cos(n \arccos(x)), & x \in [-1, 1] \\ \cosh(n \operatorname{arccosh}(x)), & x \geq 1 \\ (-1)^n \cosh(n \operatorname{arccosh}(-x)), & x \leq -1 \end{cases}$$

Theorem 5.1.1. *T_n is a Chebyshev polynomial of the first kind. For any given $1 \leq n$, among the polynomials of degree n with leading coefficient 1, $f(x) = \frac{1}{2^{n-1}}T_n(x)$ is the one of which the maximal absolute value on the interval $[-1, 1]$ is minimal. This maximal absolute value $\frac{1}{2^{n-1}}$ is and $|f(x)|$ reaches this maximum exactly n+1 times: in -1 and 1 and the other $n - 1$ extremal points of f.*

Corollary 5.1.2. *By polynomial manipulation, it is proved that for any given $1 \leq n$, among the polynomials of degree n with leading coefficient 1, $f(x) = \frac{(b-a)^n}{2^{2n-1}}T_n(\frac{2x-b-a}{b-a})$*

is the one of which the maximal absolute value on the interval $[-1, 1]$ *is minimal. This maximal absolute value* $\frac{(b-a)^n}{2^{2n-1}}$ *is and* $|f(x)|$ *reaches this maximum exactly n+1 times: in a and b and the other* $n-1$ *extremal points of* f.

Corollary 5.1.3. *A Chebyshev polynomial of either kind with degree n has n different simple roots, called Chebyshev roots, in the interval [-1,1]. The roots are sometimes called Chebyshev nodes because they are used as nodes in polynomial interpolation. Using the trigonometric definition and the fact that*

$$\cos(\frac{\pi}{2}(2k+1)) = 0 \tag{5.4}$$

one can easily prove that the roots of T_n *are*

$$x_k = \cos(\frac{\pi}{2}\frac{2k-1}{n}), k = 1, \ldots, n. \tag{5.5}$$

and the roots of Chebyshev polynomial in range $[a, b]$ *are*

$$x_k = \frac{b-a}{2}\cos(\frac{\pi}{2}\frac{2k-1}{n}) + \frac{a+b}{2}, k = 1, \ldots, n. \tag{5.6}$$

Figure 5-1 is an illustration of a Chebyshev Polynomial and its roots. The polynomial reaches low values in the range of roots, which is referred to as **Chebyshev Range** in the context (the range complementing this range in the whole range is referred to as **Remainder Range** or **Leftover Range**). With proper choice of this range, the modes outside this range will be amplified substantially more to overwhelm the modes inside.

Then we restudy the examples in section 3.3.

1. For a system (1.4), A is a symmetric dense matrix with eigenvalues uniformly distributed in $[-10, -1]$. The Chebyshev Range is $[-10, -1.5]$ and Remainder range is $[-1.5, -1]$. Figure 5-2 shows the scaling of different eigenvalues.

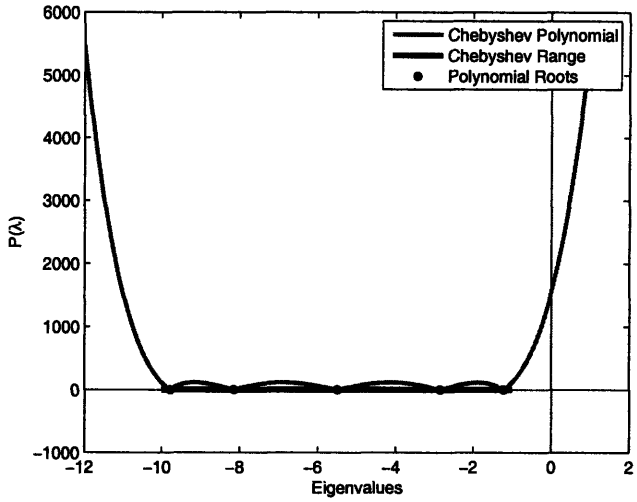

Figure 5-1: Chebyshev Polynomial and its roots

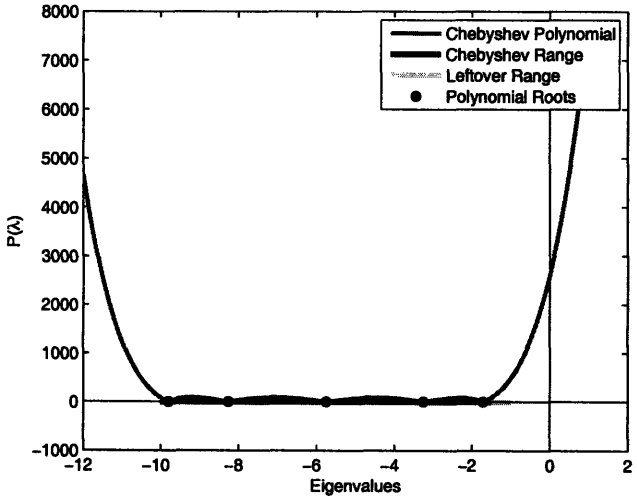

Figure 5-2: Chebyshev Polynomial and its roots: Type 1

The Remainder range is amplified substantially more than the Chebyshev Range. With proper choice of the ranges and number of shifts, the whole Chebyshev Range and some part of the remainder can be eliminated after shifts.

2. For a system (1.4), A is a symmetric dense matrix with eigenvalues uniformly distributed in $[-2.05, -1.95]$. The Chebyshev Range is $[-2.05, -1.95]$ and Remainder range is null. Then the Chebyshev Range will be eliminated whole.

In fact, in practice, no explicit analysis will be carried out about whether a system belongs to the first or the second kind. For the second kind, as in the latter example, even the Chebyshev Range is $[-2.05, -1.93]$ and the remainder range is $[-1.93, -1.95]$, the results will be little affected, as the roots generated from the $[-2.05, -1.95]$ will not differ much from those from $[-2.05, -1.93]$. Therefore, in practice, a percentage α, for example 1%, is chosen to determine how much of the whole eigenvalue range belongs to the Remainder range and the left range belongs to the Chebyshev Range.

The reason why the Chebyshev Range leans towards the fast-decaying modes (most negative eigenvalues) is that slow-decaying modes are in naturally advantageous positions, as in (5.2) the reciprocal of slow-decaying modes are larger. For example, for two eigenvalues -1 and -10, the corresponding singular values are 1 and $\frac{1}{10}$, it is more convenient and efficient to amplify and keep -1 instead of -10. This choice of Chebyshev Range is to make use of the favorable situation in the original system.

5.2 Full Algorithm Description

Table 5.1 is the brief description of the algorithm:

Table 5.1: Full Algorithm Description

ϵ: the threshold for keeping states
α: the percentage of the Remainder range
ρ': pre-stored shifts

Eigenvalue decompose $A = Q\Lambda Q^{-1}$, get Q and Λ

if no ρ' are inputted:

Choose shifts using Chebyshev Polynomials using α, in section 5.1

% **Generate the shifted system**

Generate equations (6.2) and (6.3)

% **Generate the Remainder system**

Generate Remainder using equation (4.2)

$A_l = [\lambda_i]_i$, $B_l = Q^{-1}B$

$C_l = CQ\left[\prod_{k=1}^{d}(\lambda_i - \lambda^{(k)})\right]_i$, $D_l = D$

% **Obtain reduced model**

Apply SVDMOR (in Table 2.1) to the system (A_l, B_l, C_l, D_l) with ϵ

and obtain (A_r, B_r, C_r, D_r) that constitute $\tilde{G}_{RS}(s)$

Output error e_2 (see (4.4))

% **If Simulation is required**

Evaluate $G_{CT}(s)u$ and $\tilde{G}_{RS}(s)u$

Add $G_{CT}(s)u$ and $\tilde{G}_{RS}(s)u$

5.3 Based on Optimization

5.3.1 Inaccuracy of Chebyshev Approach

The motivation of coming up with the idea of the Chebyshev approach is that only the amplifying factor $P(\lambda_i)$ is considered. In other words, our goal is to minimize the value of $P(\sigma) = \prod_{k=1}^{q}(\sigma - \sigma^{(k)})$ in a certain range. However, more accurately, we should minimize the value of $P(\sigma) = \frac{\prod_{k=1}^{q}(\sigma-\sigma^{(k)})}{-\sigma}$, as observed in the Remainder term.

On the other hand, under different circumstances, either the Remainder or the Contour Terms can take the dominance. The relative relation between these two

terms is not studied previously, because we only focused on the Remainder itself.

This means the Chebyshev approach may not generate an optimal set.

5.3.2 Basic Idea

The goal is to find the least number of states left for a given number of shifts. However, when the objective, the number of states, is an integer and the variable is the shifts, this optimization problem seems intractable, as the gradient of the objective function may be undefined.

Then the problem is formulated based on the idea that if only a certain number of states are kept, how low can the error be. If this error is lower than the predefined threshold, the set of shifts is acceptable. Keeping changing the number of states in ascending order searching for acceptable sets will yield the least number of states. For instance, for a 100×100 system, we are interested in choosing 3 shifts to keep the least number of states. The idea is like this: we force the Remainder to keep only 1 state, carry out the optimization and get the minimum error with only 1 state. If this error is less than threshold 5%, then 1 is the least number of states. However, if the error is larger than the threshold, we force the Remainder to keep 2 states and check the error again. We keep increasing the number of states we force to keep. Therefore, the process is an iteration of optimization, in each step, we fix the number of shifts and number of states and check whether the minimum error under this condition can reach the goal.

A single step of this optimization problem is described as follows:

Let $G(s)$ be the original system's transfer function and $G(s, \rho', a)$ be the approximated system's transfer function given the shifts ρ' and number of states kept a

regardless of the error, then the objective is to minimize the function

$$\begin{aligned} F(\rho', a) &= \frac{||G(0, \rho', a) - G(0)||_2}{||G(0)||_2}, \\ &= \frac{||G_{CT}(0, \rho', a) + \tilde{G}_{RS}(0, \rho', a) - G(0)||_2}{||G(0)||_2}, \\ &= \frac{||\tilde{G}_{RS}(0, \rho', a) - G_{RS}(0, \rho', a)||_2}{||G(0)||_2}, \end{aligned}$$

or to choose some ρ' to obtain the most accurate DC response matrix of the approximated system in the 2-norm. To evaluate $F(\rho', a)$ is through (5.1): the numerator is the $(a+1)$th largest singular value in (5.1) and the denominator is the largest singular value in (5.1).

The optimization problem is then formulated as

$$\begin{aligned} \min \quad & F(\rho', a) \\ s.t \quad & \rho' \leq \max(\rho) \\ & \rho' \geq \min(\rho) \\ & \rho' \in \mathbb{R}^d, \end{aligned} \tag{5.7}$$

where d is a predefined integer, a is the certain number of states and ρ' is the variable. Note that the gradient of the objective cannot be computed analytically. Then any suitable methods for solving nonlinear optimization problems with constraints can be applied here. Table 5.2 is the detailed description of the optimization approach.

5.3.3 The Algorithm

Table 5.2: Optimization Algorithm of the choice of shifts

Globalize A, B, C, D, a;
Manually choose a set ρ' (Often Chebyshev Polynomial roots);
Use ρ' to obtain NS, the number of states left;

```
%% p is a control over the optimization problem
a = NS/2;
Optimize[1] F(0, ρ', a) and find the least error, e_l;
%% Check to see whether the optimized set can keep a states.
If e_l > thres
%% No! Increase a
  While e_l > thres
    a = a + 1;
    Optimize F(0, ρ', a) and obtain ρ, e_l;
  end
else
%% Yes! Decrease a to expect a better result
  While e_l <= thres
    a = a − 1;
    Optimize F(0, ρ', a) and obtain ρ, e_l;
  end
end
```

This scheme first performs a MOR using a pre-set Chebyshev points and obtains the number of states left, NS. Every iteration, the optimization problem uses the manual set ρ' as the initial guess. Then it chooses half the number of states NS as the guess of how many states can be kept after optimization and then decreases or increases this number to reach our goal. This is more advantageous over increasing a from 0. For example, if 8 states are kept without optimization, this approach takes 4 as the initial guess of the minimal number of states left after optimization of choice. If the error is below the threshold with 4 states, then 3 is tried as the number of states forced to keep; otherwise, 5 is tried until we can find the least number of states with the error less than the threshold.

[1]Please refer to a single step (5.7) in 5.3.2.

5.3.4 Example

This is a 100×100 system with eigenvalues in $[-15.8, -0.02]$. We use 5% as the percentage to choose the Chebyshev roots. After optimization, as in Figure 5.3.4, the optimized set shifts nearer to the origin, which is a reflection of the effects of the damping factor $\frac{1}{\lambda_i}$. As a result, 4 states are kept after optimization while there are 8 before.

Here 5% is the optimal range chosen manually and the optimization slightly change the distribution of shifts. However, more often when α is not chosen properly (too large or too small), the optimization process will perform better by adjusting α and the distribution simultaneously. For example, in section 6.4, the distribution of the shifts is changed the other way around.

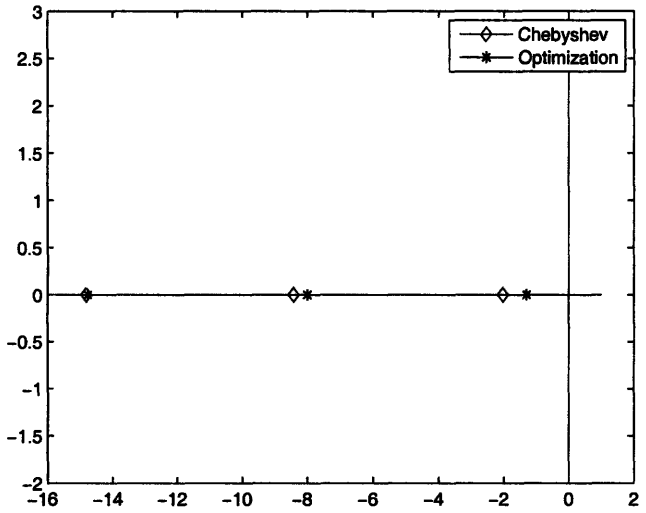

Figure 5-3: Optimization Results

5.3.5 Computational Complexity

The optimization process is not unreasonably expensive. After obtaining A's eigenvalues once, in the process of optimization, only the singular values of (5.1) and (5.2) are needed.

Due to the similarity of eigenvalue distributions of two systems with similar structures, the set obtained from a smaller system can also be advantageous for a large system. Unfortunately, due to time limitations, work still needs to be continued on this subject.

5.4 Conclusions

1. The basic principle of choosing shifts is to eliminate more states within a certain threshold. This is through the choice of Chebyshev roots.

2. Due to the inaccuracy of the Chebyshev idea, optimization is required to obtain a set in order to eliminate more states.

Chapter 6

Numerical Results

6.1 Computational Complexity

6.1.1 Issues on Storage and Computation

Simulate the Contour Terms

The reduced system is normally used for simulation, or to obtain the output given a set of input vectors u. Moreover, the system matrix A is often sparse with $O(n)$ entries. Therefore, to multiply A with a vector only requires $O(n)$ operations. Therefore, instead of representing the system using Λ and sparse matrix Q, to use direct A is a better choice for simulation. Using the property $QP(\lambda)Q^{-1} = P(A)$ (P is a polynomial), each Contour Term is changed accordingly to:

$$\begin{aligned} G_0(s) &= \frac{CB}{s-\sigma^{(1)}}, \\ G_1(s) &= \frac{C(A-\sigma^{(1)}I)B}{(s-\sigma^{(2)})(s-\sigma^{(1)})}, \\ G_2(s) &= \frac{C(A-\sigma^{(2)}I)(A-\sigma^{(1)}I)B}{(s-\sigma^{(3)})(s-\sigma^{(2)})(s-\sigma^{(1)})}, \\ &\vdots \end{aligned}$$

$$
\begin{aligned}
G_d(s) &= \frac{C(\prod_{k=1}^{q-1}(A-\sigma^{(k)}I))B}{\prod_{k=1}^{q}(s-\sigma^{(k)})}, \\
G_{CT}(s) &= \sum_{m=1}^{d} G_m(s). \qquad (6.1)
\end{aligned}
$$

Defining $u_0 = Bu$, $u_1 = (A-\sigma^{(1)}I)Bu$ and so on,

$$
\begin{array}{llll}
\dot{x}_1^{(1)} - \sigma^{(1)}x_1^{(1)} = u_0 & \dot{x}_2^{(1)} - \sigma^{(1)}x_2^{(1)} = u_1 & \dot{x}_3^{(1)} - \sigma^{(1)}x_3^{(1)} = u_2 & \cdots \quad \dot{x}_d^{(1)} - \sigma^{(1)}x_d^{(1)} = u_d \\
 & \dot{x}_2^{(2)} - \sigma^{(2)}x_2^{(2)} = x_2^{(1)} & \dot{x}_3^{(2)} - \sigma^{(2)}x_3^{(2)} = x_3^{(1)} & \cdots \quad \dot{x}_d^{(2)} - \sigma^{(2)}x_d^{(2)} = x_d^{(1)} \\
 & & \dot{x}_3^{(3)} - \sigma^{(3)}x_3^{(3)} = x_3^{(2)} & \cdots \quad \dot{x}_d^{(3)} - \sigma^{(3)}x_d^{(3)} = x_d^{(2)} \\
 & & & \cdots \quad \dot{x}_d^{(4)} - \sigma^{(4)}x_d^{(4)} = x_d^{(3)} \\
 & & & \qquad \vdots \\
 & & & \dot{x}_d^{(d)} - \sigma^{(d)}x_d^{(d)} = x_d^{(d-1)}.
\end{array}
$$

Comparing this set of equations, we can get

$$
\begin{aligned}
x_2^{(1)} &= (A-\sigma^{(1)}I)x_1^{(1)}, \\
x_3^{(2)} &= (A-\sigma^{(2)}I)x_2^{(2)}, \\
&\vdots \\
x_q^{(d-1)} &= (A-\sigma^{(d-1)}I)x_{d-1}^{(d-1)}.
\end{aligned}
$$

Incorporating this into the triangular set of equations, we can only keep the last equation in each column:

$$
\begin{aligned}
\dot{x}_1^{(1)} - \sigma^{(1)}x_1^{(1)} &= u_0, \\
\dot{x}_2^{(2)} - \sigma^{(2)}x_2^{(2)} &= (A-\sigma^{(1)}I)x_1^{(1)}, \\
\dot{x}_3^{(3)} - \sigma^{(3)}x_3^{(3)} &= (A-\sigma^{(2)}I)x_2^{(2)}, \\
&\vdots \\
\dot{x}_d^{(d)} - \sigma^{(d)}x_d^{(d)} &= (A-\sigma^{(d-1)}I)x_{d-1}^{(d-1)}.
\end{aligned}
$$

or

$$
\begin{aligned}
\dot{x_1^{(1)}} - \sigma^{(1)} x_1^{(1)} &= u_0, \\
\dot{x_2^{(2)}} - \sigma^{(2)} x_2^{(2)} - (A - \sigma^{(1)} I) x_1^{(1)} &= 0, \\
\dot{x_3^{(3)}} - \sigma^{(3)} x_3^{(3)} - (A - \sigma^{(2)} I) x_2^{(2)} &= 0, \\
&\vdots \\
\dot{x_q^{(d)}} - \sigma^{(d)} x_d^{(d)} - (A - \sigma^{(d-1)} I) x_{d-1}^{(d-1)} &= 0.
\end{aligned}
$$

Defining $X = [x_1^{(1)}, x_2^{(2)}, \cdots, x_d^{(d)}]^T$,

$$X - A_N X = U,$$

where

$$
A_N = \begin{bmatrix}
I & & \cdots \\
-(A - \sigma^{(1)} I) & I & \cdots \\
& -(A - \sigma^{(2)} I) \quad I & \cdots \\
& \cdots & \cdots \\
& \cdots & -(A - \sigma^{(d-1)} I) \quad I
\end{bmatrix} \in \mathbb{R}^{nd \times nd}, \qquad (6.2)
$$

and $U = [u_0, 0, \cdots, 0]^T \in \mathbb{R}^{nd \times 1}$. To invert and apply this big A matrix using Gaussian Elimination, the computational complexity is $O(nd)$, because all the A's are off diagonal and will be seen only in multiplications.

And the original output is

$$y = C_N X + Du,$$

where

$$C_N = \begin{bmatrix} I_{n \times n} & I_{n \times n} & I_{n \times n} & \cdots & I_{n \times n} \end{bmatrix} \in \mathbb{R}^{n \times nq}. \qquad (6.3)$$

Operation Counts

Assume A is sparse with $O(n)$ entries and we only concern about the terms that cannot be pre-calculated. Table 6.1 shows detailed operation counts of a simulation of the reduced system.

Table 6.1: Operation Counts of the Shifting Method

Procedure	Operation Counts
$x_1 = (x_1 + Bu(t))/(1 - dt \times \rho^{(1)})$	$3n$
$i = 2 : d$	
$x_i = (x_i + dt \times (A - I\rho^{(i-1)}) \times x_{i-1})$	$3n(d-1)$
$y = C(x_1 + x_2 + \cdots + x_d)$	dn
Reduced System Modeling	$O(r^3 + rd)$
Total	$O(4nd + r^3 + rd)$

If A's are dense matrices with favorable eigenvalue distributions, this method will outperform the direct method very much. However, in practice, many A's are sparse and to Gaussian elimiate them does not require expensive operations. In section 6.2 a typical example will be discussed.

6.1.2 Extension to smaller number of terminals

The shifting method imposes no specification on the number of inputs and outputs. Assume a system with smaller number of inputs and outputs. The Contour Terms

are:

$$\begin{aligned} G_0(s) &= \frac{CB}{s-\sigma^{(1)}}, \\ G_1(s) &= \frac{C(A-\sigma^{(1)}I)B}{(s-\sigma^{(2)})(s-\sigma^{(1)})}, \\ G_2(s) &= \frac{C(A-\sigma^{(2)}I)(A-\sigma^{(1)}I)B}{(s-\sigma^{(3)})(s-\sigma^{(2)})(s-\sigma^{(1)})}, \\ &\vdots \\ G_d(s) &= \frac{C(\prod_{k=1}^{d-1}(A-\sigma^{(k)}I))B}{\prod_{k=1}^{d}(s-\sigma^{(k)})}. \end{aligned}$$

Suppose $B \in \mathbb{R}^{n\times p}$ and $C \in \mathbb{R}^{q\times n}$. The goal is to obtain the outputs, therefore it is unnecessary to get the states first and then multiply them with the C matrix.

So instead, we pre-calculate the terms: $M_1 = CB$, $M_2 = C(A-\sigma^{(1)}I)B$, $M_3 = C(A-\sigma^{(2)}I)(A-\sigma^{(1)}I)B, \cdots, M_d = C(\prod_{k=1}^{d-1}(A-\sigma^{(k)}I))B$, which are much probably dense matrices. If we multiply these matrices with a vector, the operation is on the order of $O(pq)$.

Defining $u_1 = M_1u$, $u_2 = M_2u$ and so on (To evaluate each u_i requires $O(pq)$ operations), as well as $y = Cx$

$$\begin{array}{llll} \dot{y}_1^{(1)} - \sigma^{(1)}y_1^{(1)} = u_1 & \dot{y}_2^{(1)} - \sigma^{(1)}y_2^{(1)} = u_2 & \dot{y}_3^{(1)} - \sigma^{(1)}y_3^{(1)} = u_3 & \cdots \quad \dot{y}_d^{(1)} - \sigma^{(1)}y_d^{(1)} = u_d \\ & \dot{y}_2^{(2)} - \sigma^{(2)}y_2^{(2)} = y_2^{(1)} & \dot{y}_3^{(2)} - \sigma^{(2)}y_3^{(2)} = y_3^{(1)} & \cdots \quad \dot{y}_d^{(2)} - \sigma^{(2)}y_d^{(2)} = y_d^{(1)} \\ & & \dot{y}_3^{(3)} - \sigma^{(3)}y_3^{(3)} = y_3^{(2)} & \cdots \quad \dot{y}_d^{(3)} - \sigma^{(3)}y_d^{(3)} = y_d^{(2)} \\ & & & \cdots \quad \dot{y}_d^{(4)} - \sigma^{(4)}y_d^{(4)} = y_d^{(3)} \\ & & & \qquad \vdots \\ & & & \dot{y}_d^{(d)} - \sigma^{(d)}y_d^{(d)} = y_d^{(d-1)}. \end{array}$$

To evaluate each of them requires $O(3q)$ operation. Moreover, adding all the y together requires $O((d-1)q)$, which adds up to a total of approximately

$$O(2(d^2+d)q + dpq)$$

if pq is much less than n, this approach can be effective. From this perspective, this

can be an alternative to do SISO MOR.

6.2 2-D Grid RC Circuit Simulation

The same as referred to in Section 2.3, the two-dimensional example is studied in the following.

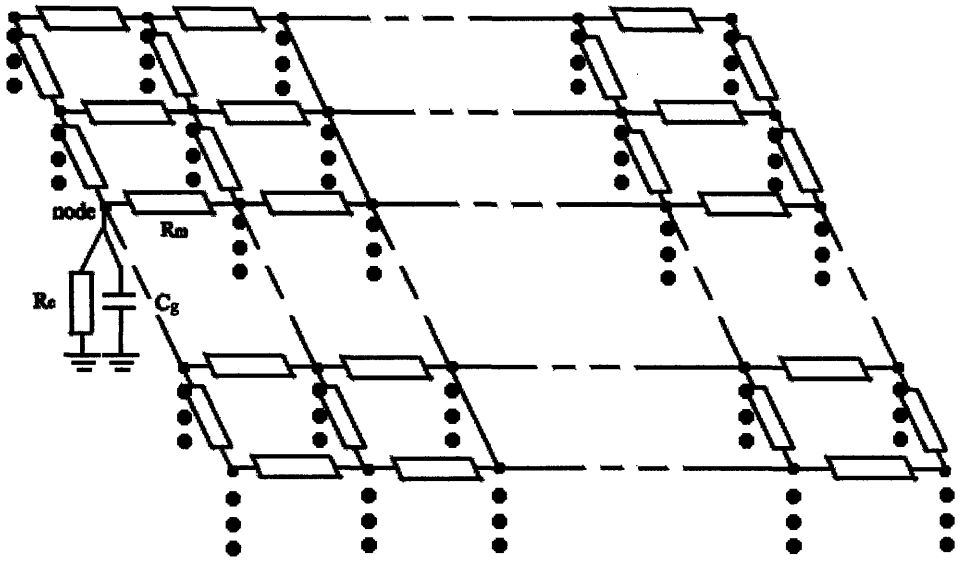

Figure 6-1: Structure of the Two-Dimensional RC Circuit

The objective is a $n \times n$ two-dimensional Resistor-Capacitor (RC) circuit. It contains n^2 nodes, each node connecting with its neighbouring node with a coupling resistor R_m and connecting with parallel capacitor C_g and ground resistor R_c to the ground. The three black dots connected to each node denote the parallel capacitor and ground resistor.

This is a typical example in simulation that can be formulated as an ODE, see [27]. Using the same technique as in equation (1.2), the simulation of this problem is formulated as a first order ordinary differential equation, which is a dynamic system. The input is a vector of currents into each node and the output is a vector of voltages on each node, both with length n^2. The states are the same as the output, therefore the system is at order-n^2 and $C = I_{n^2 \times n^2}$.

6.3 Comparison Study

We introduce three examples:

1. All $R_m = 100\Omega$, $C = 20pF(2 \times 10^{-8})$ and $R_c = 10k\Omega$. Eigenvalues of A are distributed between $[-3.91e+6, -5.00e+3]$.

2. All $R_m = 100\Omega$, $C = 20pF(2 \times 10^{-8})$ and $R_c = 1k\Omega \pm 10\%$. Eigenvalues of A are distributed between $[-3.95e+6, -4.79e+4]$.

3. All $R_m = 100\Omega$, $C = 20pF(2 \times 10^{-8})$ and $R_c = 100\Omega$. Eigenvalues of A are distributed between $[-4.40e+6, -5.00e+5]$.

In this experiment, 10×10, 20×20, 30×30, 40×40, 50×50, 60×60 grid systems are selected for simulation. Each simulation[1] takes 1000 steps, with errors defined in section 4.4 less than 5%. The shifting methods used here did not optimize the set of shifts. Table 6.2 shows the comparison of approximate operation counts.

Table 6.2: Comparison of methods to different systems

States #	100	400	900	1600	2500	3600
Direct	2000000	1.2e+007	3.6e+007	8e+007	1.5e+008	2.52e+008
(1)	611500	2.81e+006	9.33e+006	2.99e+007	6.6e+007	1.26e+008
Shifts #	1	1	1	1	3	4
States Left	3	8	17	30	33	41
(2)	1.67e+006	1.14e+007	3.66e+006	8.86e+007	n/a*	n/a
Shifts #	2	4	6	8		
States Left	8	15	22	31		
(3)	2.55e+006	1.03e+006	2.38e+006	4.41e+007	7.29e+007	1.08e+008
Shifts #	5	5	5	5	5	5
States Left	1	4	9	15	22	31
Shifts #	6	6	6	6	6	6
States Left	0	0	0	0	0	0

[1]Simulation is carried out in Matlab.

* means that with less than 9 shifts, keeping less than $\sqrt{n}$ states is impossible and the shifting method cannot beat the direct method.

For this 2-D circuit problem, to apply A^{-1} to a vector requires approximately $O(n^{1.5})$ operations. However, in Table 6.1, the shifting method has complexity $O(4nd + r^3 + rd)$. Therefore, to pick up the shifts, both $d \leq \frac{\sqrt{n}}{4}$ and $r \leq \sqrt{n}$ should be guaranteed.

It is interesting to observe that example (1) belongs to a typical type 1, as the eigenvalues are far apart and there will always be states left regardless of how many shifts are used; example (3) belongs to a typical type 2, as the eigenvalues are close enough that after 6 shifts, no states are needed to be kept; example (2) belongs to the kind we believe is relatively difficult to handle, where the shifting method is not even as efficient as the direct one in large systems.

With retrospect to the configurations, in example (1), ground resistors are much larger than coupling resistors, which means the injected currents are most affected by heavy loads, and in example (3), ground resistors are the same as coupling ones, and most of currents will flow into the ground rather than passing through other nodes. Therefore, in these cases, there are prominent characteristics and this is why shifting method can capture well.

Table 6.3 shows the number of states left after shifts in example (1).

Table 6.3: Number of States left vs Shifts

Shifts #	1	2	3	4	5	6	7	8
100	3	3	3	3	3	1	1	1
400	8	8	8	6	6	4	4	4
900	17	15	13	13	11	9	8	8
1600	30	28	22	20	17	15	15	13
2500	43	39	33	30	26	22	22	20
3600	60	54	48	41	37	31	30	28

The convergence is showed in Figures A-1, A-2 and A-3. It needs to be pointed out that for example (1) for larger size of systems, more shifts are needed in order to guarantee the accuracy. Here is the dilemma: on the one hand, as in Figure 5-1, for a given number of shifts using Chebyshev approach will keep a portion, say ϵ, of the whole number as the number of states left, which is ϵn but $r \leq \sqrt{n}$. This means that for a given number of shifts, there will a larger n where these shifts are not sufficient. on the other hand, as the larger size of system requires more shifts, the tolerant number of shifts is also increasing, as in $d \leq \frac{\sqrt{n}}{4}$. As no optimization of choice of shifts is carried out and storage is limited, it is still unclear whether this will finally affect the convergence of the shifting method for the first type.

6.4 Optimization of the choice of shifts

The optimization is applied to example (2) (10×10,20×20,30×30 grid) where direct Chebyshev method is in unsatisfactory performance. The optimization of the choice of shifts shows:

1. Compared with random initial guesses, Chebyshev points with $\alpha = 3\%$ lead to the minimal number of states left. This is that a random initial guess may lead to a local minimum because of non-convexity.

2. For the 30×30 case, Table 6.4 shows the comparison of states left before and after optimization.

   Table 6.4: Number of shifts before and after optimization in example (2)

Shifts #	3	4	5	6	7	8
Before	>30	30	28	22	22	22
After	31	22	22	17	13	13

The change of shifts' distribution is shown in Figure A-7, A-8 and A-9. The primary change in the distribution is in the shifts near to the origin, or the slow modes. As mentioned in section 5.1, the slow decaying mode have a naturally larger impact and attention paid to these modes results in much better outcome, that is, less states left.

6.5 Point-to-Point Error

Sometimes the 2-norm error is not strict enough to guarantee accurate approximation and we observe the point-to-point transfer function value in DC frequency. Table 6.5 shows the different properties of the transfer functions of a middle point to all the other points in the grid in DC case, or a column of DC response matrix. Please refer to Figures A-4, A-5 and A-6.

Table 6.5: Comparison of point-to-all transfer functions

Example	Transfer Function Value	DC Error	MOR
1	Concentrated around the point itself, low value at others.	Low Error around the point itself, may have enormous errors at other positions.	Efficient
2	The positions around point itself are a little larger than at other places	May have large errors around the points nearby.	Difficult
3	Relatively similar at different positions.	Low Error at every positions. Mostly below 5%.	Efficient

From Table 6.5, it is seen that either example 1 or 3 can be satisfying because at every point where the transfer function value is significant, the error is maintained at a low level. For example 2, it is a poor type, as applying MOR to it is more difficult than the other two and it has large errors where there are unignorable transfer function values, although the overall 2-norm error is less than the threshold.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

1. The shifting method is to most possibly represent a system's characteristics through successively picking shifts and the computational complexity is on order n.

2. Systems with sufficiently clustered or sufficiently separated eigenvalues are most favored by the shifting approach, though using different philosophies.

3. The choice of shifts requires experiments and is an optimization process that generates statistically optimal results if Chebyshev polynomial roots are chosen as the initial guess.

4. For 2-D RC circuit problems, the shifting method outperforms direct method when there is strong coupling or there are heavy loads. For the former case, convergence is guaranteed but not sure for the latter one.

7.2 Future Work

1. For the system between type 1 and 2, research should be continued about picking better shifts and guaranteeing more satisfying point-to-point errors. For current approach, in 2-D problem, without optimization, the shifting method could

not even outrun the direct method. Normally, the norm, which is a global metric of matrices, is a looser standard than the point-to-point error guarantee. Therefore, regarding point-to-point errors, mere 2-norm based SVDMOR will require further actions to address the problem. [29] provides inspiring directions in how to guarantee more strict point-to-point errors more efficiently.

2. The dilemma between the fact that more shifts are required for larger systems and the one that more shifts are allowed in larger systems calls for more attentive research, as discussed in section 6.3.

3. The relation of rescaled eigenvalues of the original system and singular values of the Remainder is worth studying. The reason for choosing the system (1.4) is that in this system the eigenvalues of A can be directly related to the singular values of the Remainder, which SVDMOR is based on. However, for most system, this is not the case. Symbolical SVD may be helpful for this problem [30, 31].

Appendix A

Figures

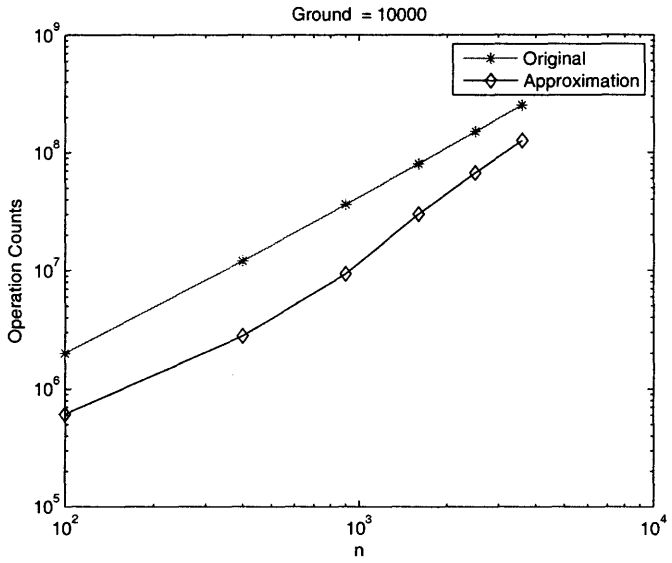

Figure A-1: Comparison of methods with different number of shifts in Example 1

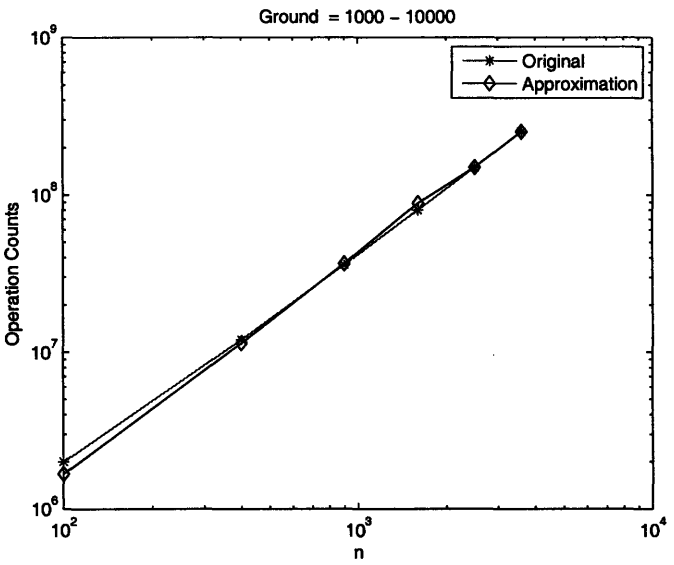

Figure A-2: Comparison of methods with different number of shifts in Example 2

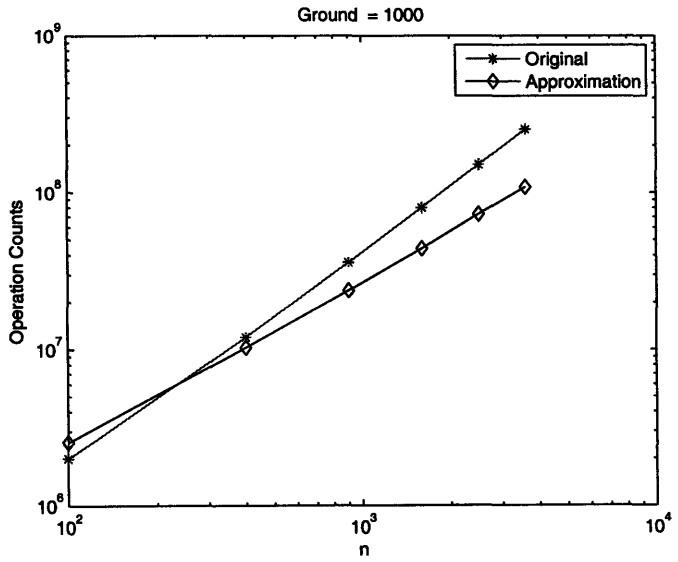

Figure A-3: Comparison of methods with different number of shifts in Example 3

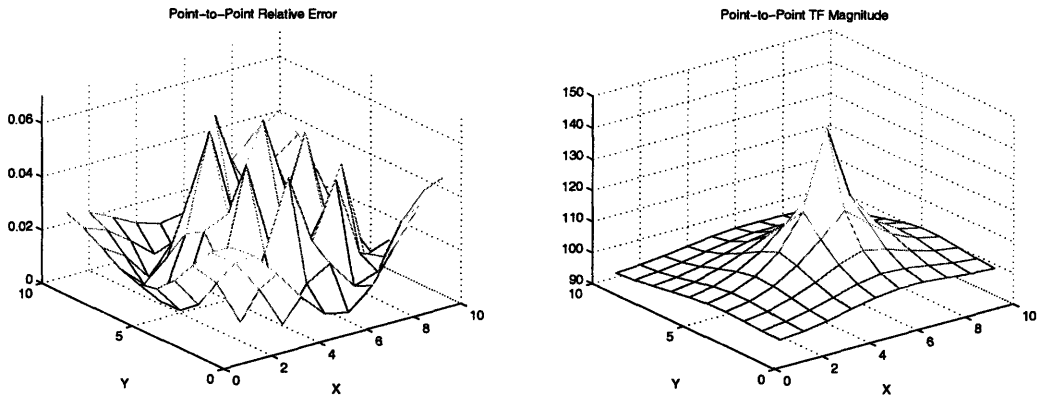

Figure A-4: Point-to-Point Transfer Function Error and Magnitude in Example 1

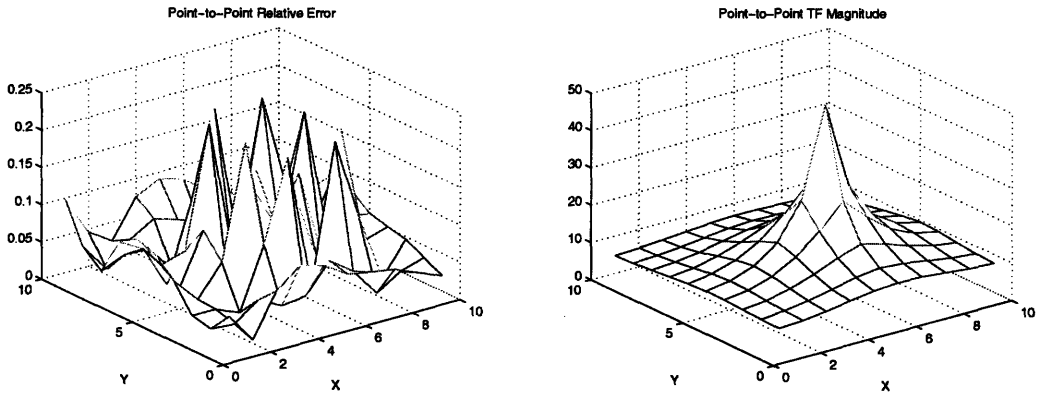

Figure A-5: Point-to-Point Transfer Function Error and Magnitude in Example 2

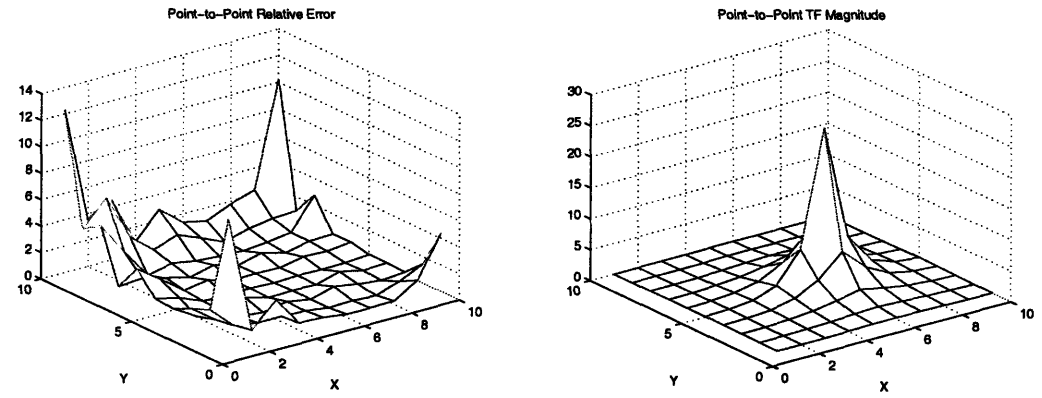

Figure A-6: Point-to-Point Transfer Function Error and Magnitude in Example 3

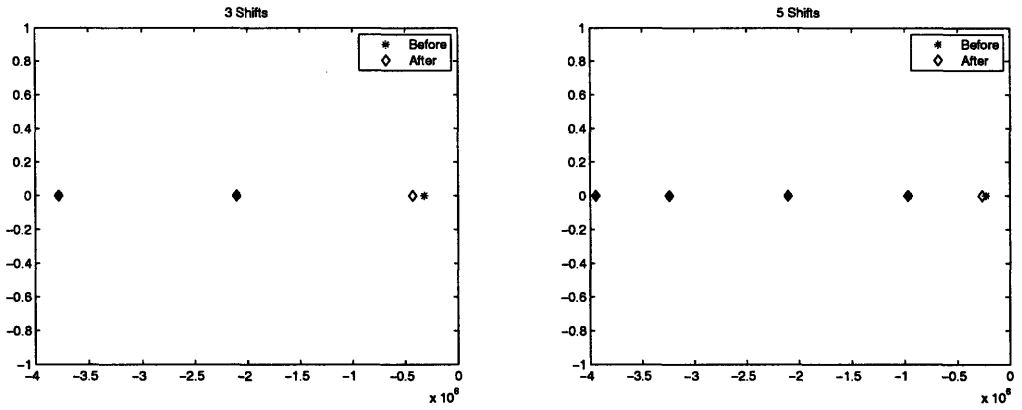

Figure A-7: Distribution of shifts before and after optimization

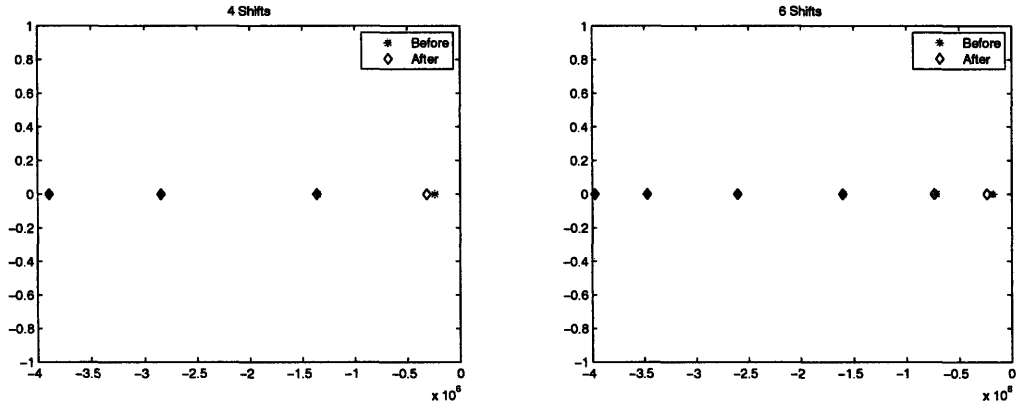

Figure A-8: Distribution of shifts before and after optimization

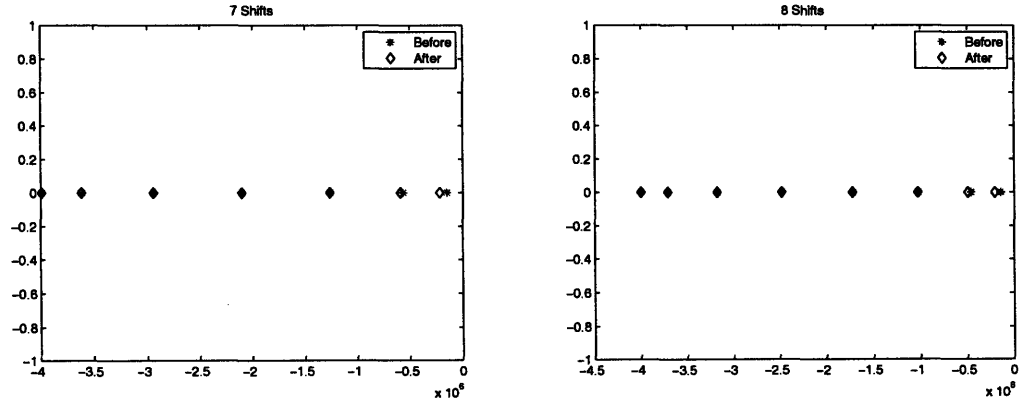

Figure A-9: Distribution of shifts before and after optimization

Appendix B

Full Algorithm

Table B.1: Full Algorithm

A, B, C, D: original system matrices
n: the number of states
d: the number of shifts
r: the number of states left in the reduced Remainder
ϵ: the threshold for keeping states
α: the percentage of the Remainder range
ρ': pre-stored shifts
Eigenvalue decompose $A = Q\Lambda Q^{-1}$, get Q and Λ
if no ρ' are inputted:
Choose shifts using Chebyshev Polynomials using α
$a = \lambda_{min}, b = \lambda_{min} + (\lambda_{max} - \lambda_{min})(1 - \alpha)$
$\rho'(k) = \frac{b-a}{2}\cos(\frac{\pi}{2}\frac{2k-1}{n}) + \frac{a+b}{2}, k = 1, \ldots, d.$
% **Generate the shifted system**
$A_{CT} = A_N, B_{CT} = B$
$C_{CT} = C_N, D_{CT} = D$
% **Generate the Remainder system**
Generate Remainder using equation (B.3)
$A_l = [\lambda_i]_i, B_l = Q^{-1}B$

$C_l = CQ\left[\prod_{k=1}^{d}(\lambda_i - \lambda^{(k)})\right]_i, D_l = D$

% **Obtain reduced model**

Apply SVDMOR (in Table B.2) to the system (A_l, B_l, C_l, D_l) with ϵ and obtain (A_r, B_r, C_r, D_r) that constitute $\tilde{G}_{LO}(s)$

Output error $e_2 = \frac{||G_{CT}(0)+\tilde{G}_{LO}(0)-G(0)||_2}{||G(0)||_2}$

% **If Simulation is required**

Evaluate $G_{CT}(s)u$ and $\tilde{G}_{LO}(s)u$

Add $G_{CT}(s)u$ and $\tilde{G}_{LO}(s)u$

$$A_N = \begin{bmatrix} I & & \cdots & \\ -(A-\sigma^{(1)}I) & I & \cdots & \\ & -(A-\sigma^{(2)}I) \quad I & \cdots & \\ & \cdots & \cdots & \\ & \cdots & -(A-\sigma^{(d-1)}I) & I \end{bmatrix} \in \mathbb{R}^{nd\times nd}, \tag{B.1}$$

$$C_N = \begin{bmatrix} I_{n\times n} & I_{n\times n} & I_{n\times n} & \cdots & I_{n\times n} \end{bmatrix} \in \mathbb{R}^{n\times nq}. \tag{B.2}$$

$$G_{LO}(s) = \frac{1}{\prod_{j=1}^{d}(s-\lambda^{(j)})} CQ \left[\frac{\prod_{k=1}^{d}(\lambda_i - \lambda^{(k)})}{(s-\lambda_i)}\right]_i Q^{-1}B \tag{B.3}$$

Table B.2: The SVDMOR algorithm

A, B, C, D: input system matrces;
ϵ: Tolerance for reducing singular values;
r: Number of remaining singular values;
1.Compute one matrix M[1] that reveals correlations.
2.Apply Low-Rank Approximation using ϵ: $M = U\Sigma V \approx U_r\Sigma_r V_r$.
3.Find weights using Pseudoinverse: $b = BU_r(U_r'U_r)^{-1}, c = (V_r'V_r)V_r'C$.
4.Form the transfer function: $G(s) \approx U_r c(sI + A)^{-1} b V_r'$.
5.Apply MIMO MOR techniques [19]-[23] to $G_r(s) = c(sI + A)^{-1}b$, which is an $r \times r$ matrix transfer function, and obtain $\tilde{G}_r(s)$.
6.Recover the system transfer function $G(s) \approx U_r\tilde{G}_r(s)V_r'$, the number of states being reduced from n to r.

Commonly used Correlation matrices include

$$\text{DC Response: } G_{DC} = -CA^{-1}B \tag{B.4}$$

$$\text{The First Moment of the Response: } G_1 = CA^{-1}A^{-1}B \tag{B.5}$$

$$\text{Frequency Shifted Moments: } G_{s_0} = C(s_0 I - A)^{-1}B \tag{B.6}$$

[1]This matrix is often called Correlation Matrix.

Bibliography

[1] Lewis A. Rossman. *Storm Water Management Model, User's Manual.* Water Supply and Water Resources Division, National Risk Management Research Laboratory, Environment Protection Agency, 5.0 edition, October 2005.

[2] Athanasios C.Antoulas. *Approximation of Large-Scale Dynamical Systems.* SIAM, 2005.

[3] K. Glover. All optimal Hankel-norm approximations of linear multivariable systems and their $l\infty$ error bounds. *International Journal of Control*, 39(6):1115–1193, 1984.

[4] J. Phillips, L. Daniel, and L.M. Silveira. Guaranteed passive balancing transformations for model order reduction. In *Proceedings of the 39th Design Automation Conference*, pages 52–57, 2002.

[5] K.C. Hall, J.P. Thomas, and E.H. Dowell. Reduced-order modelling of unsteady small-disturbance flows using a frequency-domain proper orthogonal decomposition technique. In *Proceedings of the 37th Aerospace Sciences Meeting and Exhibit*, pages 99–0655. AIAA, 1999.

[6] K. Willcox and J. Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal*, 40(11):2323–2330, 2002.

[7] Jing-Rebecca Li. *Model Reduction of Large Linear Systems via Low Rank System Gramians.* PhD thesis, Massachusetts Institute of Technology, 2000.

[8] P. Feldmann and R.W. Freund. Efficient linear circuit analysis by Pade approximation via the Lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14:639–949, 1995.

[9] Eric James Grimme. *Krylov Projection Methods for Model Reduction.* PhD thesis, University of Illinois at Urbana-Champaign, 1997.

[10] Z. Bai. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Applied Numerical Mathematics*, 43(1-2):9–44, 2002.

[11] J.R. Phillips. Projection frameworks for model reduction of weakly nonlinear systems. In *Proceedings of the 37th Design Automation Conference,*, pages 184–189, 2000.

[12] J.R. Phillips. Projection-based approaches for model reduction of weakly nonlinear, time-varying systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(2):171–187, 2003.

[13] S. Lall, J.E. Marsden, and S. Glavaski. A subspace approach to balanced truncation for model reduction of nonlinear systems. *International Journal of Robust and Nonlinear Control*, 12:519–535, 2002.

[14] J.M.A. Scherpen. Balancing for nonlinear systems. *Systems and Control Letters*, 21:143–153, 1993.

[15] E.S. Hung and S.D. Senturia. Generating efficient dynamical models for microelectromechanical systems from a few finite-element simulation runs. *IEEE Journal of Microelectromechanical Systems*, 8(3):280–289, 1999.

[16] K.E. Willcox, J.D. Paduano, J. Peraire, and K.C. Hall. Low order aerodynamic models for aeroelastic control of turbomachines. In *Proceedings of the American Institute of Aeronautics and Astronautics Structures, Structural Dynamics, and Materials Conference*, volume 3, pages 2204–2214, 1999.

[17] Michal Rewienski and Jacob White. Model order reduction for nonlinear dynamical systems based on trajectory piecewise-linear approximations. *Linear Algebra and its applications*, pages 1–28, 2005.

[18] P. Feldmann. Model order reduction techniques for linear systems with large number of terminals. In *Proceedings of the Design, Automation and Test in Europe Conference 2004*. IEEE Computer Society Press, 2003.

[19] P. Feldmann and R. W. Freund. Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm. In *Proceedings of the 32nd Design Automation Conference*, pages 47–49, San Francisco, California, 1995. Association for Computing Machinery, Inc.

[20] K. L. Kerns and A. T. Yang. Preservation of passivity during RLC network reduction via split congruence transformations. In *Design Automation Conference*, pages 34–39, 1997.

[21] K. Gallivan, E. Grimme, and P. Van Dooren. A rational Lanczos algorithm for model reduction. *Numerical Algorithms*, 12(1-2):33–63, 1996.

[22] P. Feldmann and R. W. Freund. The SyMPVL algorithm and its applications to interconnect simulation. In *Proceedings of the 1997 International Conference on Simulation of Semiconductor Processes and Devices*, pages 113–116. IEEE, 1997.

[23] A. Odabasioglu, M. Celik, and L. Pileggi. PRIMA: Passive reduced-order interconnect macromodeling algorithm. *IEEE trans. on CAD*, 17(8):645–654, 1998.

[24] Peter Feldmann and F. Liu. Sparse and efficient reduced order modeling of linear subcircuits with large number of terminals. In *ICCAD*, pages 88–92, 2004.

[25] Trefethen and Bau. *Numerical Linear Algebra*. SIAM, 1997.

[26] J.I.Aliaga, D.L.Boley, R.W.Freund, and V.Hernandez. A Lanczos-type Method for Multiple Starting Vectors. *Mathematics of Computation*, 69:1577 – 1601, 2000.

[27] Luca Daniel. Lecture 14: Methods for ordinary differential equations. Lecture notes of Introduction to Numerical Simulation, Massachusetts Institute of Technology.

[28] Lyapunov A.M. *Stability of motion.* Academic Press, 1966.

[29] Joseph Daniel Kanapka. *Fast Methods for Extraction and Sparsification of Substrate Coupling.* PhD thesis, Massachusetts Institute of Technology, 2002.

[30] Angelika Bunse-Gerstner, Ralph Byers, Volker Mehrmann, and Nancy K.Nichols. Numerical Computation of an Analytical Singular Value Decomposition of a Matrix Valued Function. *Numerical Mathematics*, 60:1–40, 1991.

[31] Manja V. Kircanski. Symbolical singular value decomposition for a 7-dof manipulator and its application to robot control. In *ICRA (3)*, pages 895–900, 1993.