

**EFFICIENT SOLUTION OF LYAPUNOV EQUATION
FOR DESCRIPTOR SYSTEM AND APPLICATION TO
MODEL ORDER REDUCTION**

The thesis submitted to the
Department of Mathematics, BUET, DHAKA-1000
in partial fulfilment of the requirements for the degree of

**MASTER OF PHILOSOPHY
IN
MATHEMATICS**

By
MD. SUMON HOSSAIN
Student No. 1014093001P
Registration No. 1014093001, Session: Oct. 2014



Under the supervision
of
Dr. Mohammed Forhad Uddin
Professor
Department of Mathematics

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
DHAKA-1000

**EFFICIENT SOLUTION OF LYAPUNOV EQUATION
FOR DESCRIPTOR SYSTEM AND APPLICATION TO
MODEL ORDER REDUCTION**

By

MD. SUMON HOSSAIN

Student No. 1014093001P

Registration No. 1014093001, Session: Oct. 2014

**MASTER OF PHILOSOPHY
IN
MATHEMATICS**



Department of Mathematics

**BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
DHAKA-1000**

The thesis entitled

**EFFICIENT SOLUTION OF LYAPUNOV EQUATION
FOR DESCRIPTOR SYSTEM AND APPLICATION TO
MODEL ORDER REDUCTION**

Submitted by

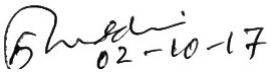
MD. SUMON HOSSAIN

Student No.1014093001P, Registration No.1014093001, Session: October 2014

has been accepted as satisfactory in partial fulfillment for the degree of

Master of Philosophy in Mathematics on 02 October 2017

BOARD OF EXAMINERS

1. 
02-10-17


Dr. Mohammed Forhad Uddin

Chairman

Professor

Department of Mathematics, BUET, Dhaka.

(Supervisor)

2. 

Head

Member (Ex-Officio)

Department of Mathematics, BUET, Dhaka.

3. 

Dr. Md. Abdul Hakim Khan

Member

Professor

Department of Mathematics, BUET, Dhaka.

4. 

Dr. Md. Manirul Alam Sarker

Member

Professor

Department of Mathematics, BUET, Dhaka.

5. 

Dr. Mohammad Monir Uddin

Member (External)

Assistant Professor

Department of Mathematics and Physics

North South University, Dhaka-1229.

Declaration of Authorship

I, Md. Sumon Hossain, declare that this thesis titled, 'EFFICIENT SOLUTION OF LYAPUNOV EQUATION FOR DESCRIPTOR SYSTEM AND APPLICATION TO MODEL ORDER REDUCTION' and the work presented in this thesis is the outcome of the investigation carried out by the author under the supervision of Dr. Mohammed Forhad Uddin, Professor, Department of Mathematics, Bangladesh University of Engineering and Technology (BUET), Dhaka-1000 and that it has not been submitted anywhere for the award of any degree or diploma.



Signed:

Date: 02.10.2017

DEDICATION

This work is dedicated

To

My beloved Parents and Teachers

Abstract

In this thesis under the title ” Efficient Solution of Lyapunov Equation for Descriptor System and Application to Model Order Reduction ”, an efficient solution of Lyapunov equation will be derived and the application of model order reduction to reduce the large system to lower dimensional system will be shown.

Mathematical models of physical systems are extending in engineering fields which can be used for simulation, optimization or control. Structured descriptor systems play important roles in many applications. Such systems are used to analyze properties of the system or simulate the system. However, many of these models are too complex, large and sometime impossible to handle for standard analysis or control system design. Hence, there is a need to reduce the complexity of models preserving the input-output behavior of the original complex model as far as possible. The existing techniques produce complexity for large dimensional state space systems.

In this thesis, the projection based techniques are considered to compute the low rank solutions of the state space systems. Recently, balanced truncation is being considered as a prominent technique for model reduction of linear time invariant systems. The most expensive part of the technique is the numerical solution of two Lyapunov matrix equations. Rational Krylov subspace method is one of the efficient methods for solving the Lyapunov equations of large-scale sparse dynamical systems. The method is well established to compute the low rank solution of the Lyapunov equations for standard state space systems. The main advantage of this solution technique is that the projected Lyapunov equations can be handled easily. We develop algorithms to solve the Lyapunov equations for large sparse structured index-1 descriptor system. The accuracy and suitability of the proposed method is demonstrated through different examples of different orders and the results are compared and discussed. Then resulting Lyapunov solutions have been applied for the balancing based model reduction.

Finally, numerical results are shown to illustrate the efficiency and accuracy of the proposed methods.

Acknowledgements

First and foremost, I would like to express my cordial gratitude to my thesis supervisor Prof. Dr. Mohammed Forhad Uddin for his continuous support, motivation, and guidance through out my thesis work. I am really thankful to him and extremely fortunate to have studied mathematics under his supervision. I am very thankful to him for introducing me in this highly fascinating and applicable research area and finish this thesis successfully. I am gratefully indebted to him.

I would like to express my hearty gratitude to all of my teacher for their valuable suggestions. I express my gratitude to my teachers from Bangladesh University of Engineering and Technology.

I am grateful to Dr. Md. Abdul Hakim Khan, Dr. Md. Manirul Alam Sarker and Dr. Md. Abdul Alim for being on my defense committee, reading my thesis and suggesting improvements.

Specially, I am extremely thankful and highly indebted to Dr. Mohammad Monir Uddin for his generous help, cooperation and valuable guidance during my research. I have learned a lot from him.

Finally, I wish to express my deep regards to my parents and all other family members and friends for their constant cooperation and motivations. Their sincerest wishes for me have played a very important role in the study.



(Md. Sumon Hossain)

Date: 02.10.2017

Contents

Declaration of Authorship	iii
Abstract	v
Acknowledgements	vi
Contents	vii
List of Figures	ix
List of Tables	xi
Symbols	xii
1 Introduction	1
1.1 Motivation and Overview	1
1.2 Chapter Outline	4
1.3 System and Control Theory	4
1.3.1 State Space Systems	5
1.3.2 Generalized Systems	8
1.3.3 Descriptor Systems	9
1.3.4 Input and Output Relation	10
1.3.5 Gramians	11
1.3.6 Stability	13
1.3.7 Hankel Singular Values	15
1.3.8 Realizations	16
1.4 Background in Linear Algebra	18
1.4.1 Eigenvalue Problem	18
1.4.2 Sparse and Dense Matrices	19
1.4.3 Matrix Decompositions	20
1.4.3.1 Eigenvalue Decomposition	20
1.4.3.2 Singular Value Decomposition	20
1.4.3.3 Schur Decomposition	21
1.4.3.4 QR Decomposition	21
1.4.3.5 Cholesky Decomposition	22
1.4.3.6 Arnoldi Decomposition	22

1.4.4	System Norms	24
1.4.4.1	Vector Norms	24
1.4.4.2	Matrix Norms	25
1.5	Existing Methods	26
1.5.1	Bartels-Stewart's Method	26
1.5.2	Hammarling's Method	27
1.5.3	Matrix Sign Function Method	27
1.5.4	LRCF-ADI Method	29
1.5.5	Krylov Subspace Method	30
1.6	Model Examples	32
1.6.1	Earth Atmospheric Model	32
1.6.2	Transmission Line Model	32
1.6.3	CD Player Model	33
1.6.4	Power System Model	33
2	Solution of Lyapunov Equation by Rational Krylov Subspace method	35
2.1	Rational Krylov Subspace Method	36
2.1.1	Generalized System	36
2.1.2	Formulation and Solution of Index-1 Descriptor System	39
2.2	Shift Parameter Selection	44
2.3	Stopping Criteria	45
2.4	Numerical Results	47
2.4.1	Artificial Data	47
2.4.2	Power System Data	50
2.4.3	Thermo-Elastic System	56
3	Application of Lyapunov Equation for Model Order Reduction	57
3.1	Concepts of Model Order Reduction	57
3.2	Model Order Reduction of Index-1 Descriptor System	60
3.3	Numerical Results	63
3.3.1	Model Reduction of Artificial Data	63
3.3.2	Model Reduction of Power System Data	64
3.3.3	Reduced Order Model for Mod-1 and Mod-2	70
3.3.4	Model Reduction of Thermo-Elastic System	70
3.4	Comparison with LRCF-ADI	71
4	Conclusions and Future Work	73
4.1	Summary	73
4.2	Future Work	74
	References	75

List of Figures

1.1	Block diagram for a work strategies	3
1.2	Block diagram for a state equation	5
2.1	Sparsity patterns of the artificial data	48
2.2	Convergence histories of RKSM for artificial data.	49
2.3	Singular values of the gramians computed by exact solver and RKSM for artificial data.	49
2.4	Computational time of controllability gramian of artificial model for different system by RKSM and exact solver	50
2.5	Sparsity patterns of the matrix A and its dense form for Mod-1 . . .	51
2.6	Convergence histories of RKSM for Mod-1.	52
2.7	Singular values of the gramians computed by exact solver and RKSM for Mod-1.	52
2.8	Convergence histories of RKSM for Mod-2.	53
2.9	Singular values of the gramians computed by exact solver and RKSM for Mod-2.	53
2.10	Sparsity patterns of the matrix A and its dense form for Mod-3 . . .	54
2.11	Convergence histories of RKSM for Mod-3.	54
2.12	Singular values of both gramians computed by RKSM for Mod-3. . .	55

2.13	Computational time for dense system and sparse system at different iteration by RKSM ($n = 21, 128$)	55
2.14	Sparsity patterns of the matrix E_1 and the matrix A of thermo-elastic system	56
3.1	The frequency responses of original system and 19 reduced-order systems in frequency domain and its errors.	64
3.2	Comparisons of Hankel singular values of original system and reduced system.	66
3.3	Eigenvalues of the original system and reduced model.	66
3.4	Errors between the original model, 70 and 55 dimensional reduced systems computed by RKSM using the system Mod-3.	67
3.5	Time domain responses for full system and reduced model, and their absolute deviations.	68
3.6	Each rows shows the relation of 1st input to 1st output, 4th input to 1st output and 2nd input to 4th output between full and reduced order models with respective relative errors.	69
3.7	Frequency responses of original system and 65 reduced-order systems in frequency domain and its errors.	71
3.8	Comparisons between full model, 67 order ADI and 67 order RKSM.	72

List of Tables

2.1	Computational time for different dimensional systems by exact solver and RKSM (sparse and dense linear system)	48
2.2	Dimensions of system matrices of BIPS models.	50
2.3	Number of differential & algebraic variables and largest eigenvalue of $(-A, E)$ for different models.	51
3.1	Comparisons of original and reduced models for all data.	65
3.2	BT tolerances and dimensions of reduced model Mod-3.	65
3.3	BT tolerances and dimensions of reduced model Mod-1.	70
3.4	BT tolerances and dimensions of reduced model Mod-2.	70
3.5	Comparisons of SLRCF-ADI method and RKSM	72

Symbols

\mathbb{R}	field of real numbers
\mathbb{C}	field of complex numbers
\mathbb{C}^-	the open left half-plane
\mathbb{C}^+	the open right half-plane
$\mathbb{R}^{m \times n}$	real matrices of order $m \times n$
$\mathbb{C}^{m \times n}$	complex matrices of order $m \times n$
a_{ij}	the i, j -th entry of matrix A
I_n	identity matrix of order n
$\operatorname{Re}(z)$	the real part of $z \in \mathbb{C}$
$\operatorname{Im}(z)$	the imaginary part of $z \in \mathbb{C}$
A^T	the transpose of matrix A
A^*	the complex conjugate transpose of matrix A
A^{-1}	the inverse of matrix A
$\ll (\gg)$	much less (greater)
\approx	approximately equal to
$G(s)$	transfer function matrix
$\ \cdot\ _{\mathcal{H}_\infty}$	\mathcal{H}_∞ -norm
$\sigma_{max}(A)$	largest singular value of A
Σ	diagonal matrix containing singular values
$\Lambda(A)$	spectral of A
$\lambda_i(A)$	i -th eigenvalue of A
$\operatorname{tr}(A)$	$:= \sum_{i=1}^n a_{ii}$, where a_{ii} be the diagonal entry of A

Chapter 1

Introduction

1.1 Motivation and Overview

Mathematical modeling plays an important role in many applications e.g., control theory, system analysis, optimization, signal processing, large space flexible structures, game theory and design of physical systems. Various complicated systems arise in many engineering applications (microelectronics, micro-electro-mechanical systems, aerospace, computer control of industrial processes, chemical processes, communication systems, etc.) are composed of large numbers of separate devices and they are described by very large mathematical models consisting of more and more mathematical systems with very large dimensions.

However, when a physical model is converted into a mathematical model, in many cases its dimension becomes extremely large [1, 2, 3, 4]. These large-scale models can make unsuitable to analyze the system. In large-scale settings, the system dimension makes the computation infeasible due to memory limitations, time limitations as well as ill-conditioning. Again, simulations of such systems can be unacceptably expensive and time consuming due to limited computer memory. Although, the computational speed and performance of the modern computers are increasing, simulation, optimization or real time controller design for such large scale systems is still difficult due to extra memory requirements and additional computational complexity. Therefore, it is suggested to replace the higher dimensional model by a substantially lower dimensional model. The process of

converting a large-scale model into small-scale model is called Model Order Reduction (MOR), e.g, [5, 6], for motivations, applications, restrictions and techniques of MOR.

Model reduction is concerned with replacing a large complex model by a much smaller one which can be fast and efficiently simulated and which has nearly the same response characteristics compare to the original large model. As the mathematical model of a device gets more detailed and the model is composed of a large system of Ordinary Differential Equations (ODEs), or a set of Partial Differential Equations (PDEs). It is quite common that the concerned mathematical model may consist a vast amount of redundant information that have very little importance in the input output characterization of the device. Model reduction is an efficient tool to eliminate those redundant parts from the original model so that the size of the reduced model becomes smaller compare to the original one and it is then amenable for simulation and analysis. The general idea of MOR is to approximate a large-scale model by a reduced model of lower state space dimension, while their input-output behavior is preserved to the largest possible extent.

In the literature, there exist many MOR techniques such as Balanced Truncation (BT) [7], pade approximation [8], moment matching approximation [9], modal truncation [10] and rational interpolation [11]. Among several methods, recently, the system theoretic method BT is prominent for the model of large-scale sparse dynamical system. Besides, the stability preservation, the method also guarantees the global error bound. These essentially make the method superior to some other existing method.

But, the only disadvantage of this method is to solve two continuous-time Lyapunov equations. Over the last decades, several iterative methods were proposed to solve large scale Lyapunov equation, e.g., LRCF-ADI (Low-Rank Cholesky Factor Alternating Direction Implicit) iterations [12, 13, 14], cyclic low-rank Smith methods [15, 16], projection methods [17, 18, 19, 20] and sign function methods [21, 22, 23]. Although, most of the methods are shown to be applicable for large scale sparse dynamical systems. Currently, LRCF-ADI method and Krylov subspace method are two frequently used methods for the solution of Lyapunov equations of large-scale sparse systems. Both the method can compute low-rank factors of the approximate solutions of the Lyapunov equations. The approaches have been developed that allow to exploit the sparsity of the matrices. Moreover,

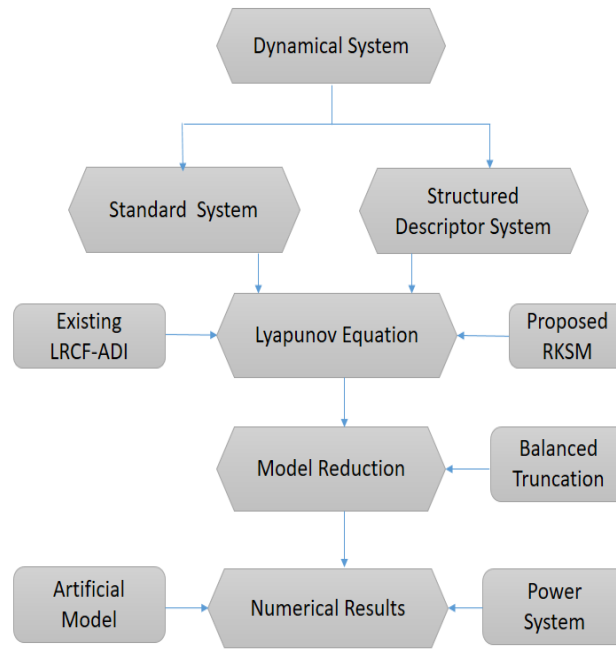


Figure. 1.1. Block diagram for a work strategies

both the methods have low-rank version for computing the low-rank gramian factors. This is important since the gramians can be approximated by therein low-rank factors if the number of inputs and outputs are very smaller than the number of Degree of Freedoms (DOFs). Therefore, instead of computing the full gramian factor, low-rank gramian factors are computationally cheap.

In [24], the author discussed a balancing based method for the model reduction of index-1 descriptor system. They applied the LRCF-ADI to solve the Lyapunov equations. It is clear that LRCF-ADI can only be applied to the asymptotically stable systems. But, in many mathematical simulation, the system is often asymptotically unstable. In this thesis, we proposed an algorithm [25, 26] in projection technique so called Rational Krylov Subspace Method (RKSM) to solve the large scale Lyapunov equations. The main advantage of the proposed technique is that it can be applied in both stable and unstable systems.

In [27], the author discussed a method for standard cases. We extend the method for index-1 descriptor system. For better convergence, an efficient strategy is applied to compute the shift parameters such that the shift parameters are computed automatically [28, 29]. We apply the computed low-rank gramian factors to the

balancing based model reduction. The graph in figure 1.1 indicates the purpose of working procedure of the thesis.

1.2 Chapter Outline

The thesis is organized as follows.

In Chapter 2, one of the main work of this research is discussed. The low rank efficient solution of large scale algebraic Lyapunov equations are shown here. Also a review and short derivation of the RKSM for generalized system are given and have introduced in details the RKSM for index-1 descriptor systems with large sparse Lyapunov matrix equation. The related issues such as computation of shift parameters and stopping criteria of the algorithm are introduced here. The benefits of preserving the sparsity pattern are also shown by graphical illustration. The efficiency of the proposed algorithms are illustrated by numerical results at the end of the chapter.

Chapter 3 provides another main part of this thesis. The solution of the Lyapunov equations are applied for model order reduction. The introductory idea of model reduction is given in first part. The system theoretical background of model reduction approach via the BT method are considered and then extend the idea of BT for index-1 descriptor systems, which requires exact system gramians. The numerical experiments are discussed elaborately at the end of this chapter.

Chapter 4 shows conclusions and briefly discusses possibilities for improvements and future research of the work.

1.3 System and Control Theory

This section describes the concepts from system and control theory that are needed in this thesis. Here we only give a very brief introduction on the most important properties and results for theory of LTI finite dimensional control systems. The author in [5] provides a good introduction that is easily readable and gives a more information to system theory from the view point of model reduction and numerical linear algebra.

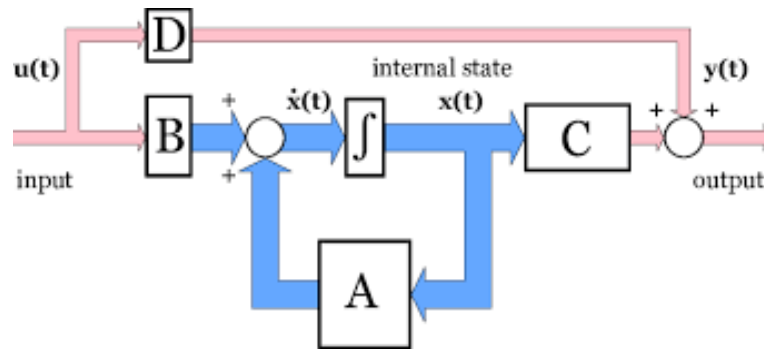


Figure. 1.2. Block diagram for a state equation

1.3.1 State Space Systems

The concept of the state of a dynamic system refers to a minimum set of variables, known as state variables, that describe the system and its response to any given set of inputs. The dynamic behavior of a state system is completely characterized by the response of the set of n variables $x_i(t)$, where the numbers n is the order of the system. The system in figure 1.2 has inputs $u(t)$ and output $y(t)$.

In general case, the form of the n state equations are as follows:

$$\begin{aligned}\dot{x}_1(t) &= f_1(x, u, t) \\ \dot{x}_2(t) &= f_2(x, u, t) \\ &\vdots = \vdots \\ \dot{x}_n(t) &= f_n(x, u, t)\end{aligned}$$

In vector notation, the set of n equations may be written as:

$$\dot{\mathbf{x}} = \mathbf{f}(x, u, t).$$

For restriction, the systems are LTI, i.e. the system is described by the linear differential equation with constant coefficients. This can be written compactly in the following matrix form:

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix}}_{\mathbf{B}} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \end{bmatrix}}_{\mathbf{u}} \quad (1.1)$$

which can be expressed as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

Again, consider a system of order n with an arbitrary output variable m with r input. In matrix form, the output m equations can be written as:

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{bmatrix}}_{\mathbf{C}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1p} \\ d_{21} & d_{22} & \cdots & d_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mp} \end{bmatrix}}_{\mathbf{D}} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \end{bmatrix}}_{\mathbf{u}} \quad (1.2)$$

which can be expressed as

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

Therefore, the complete system model for a LTI system consists of a set of n state equations and a set of output equations.

Consider a LTI dynamical system in state-space of the form:

$$\begin{aligned} \frac{dx(t)}{dt} &= Ax(t) + Bu(t); \quad x(t_0) = x_0, \quad t \geq t_0 \\ y(t) &= Cx(t) + Du(t). \end{aligned} \quad (1.3)$$

Here, the functions $x(t) : \mathbb{R} \mapsto \mathbb{R}^n$ is the state, $u(t) : \mathbb{R} \mapsto \mathbb{R}^p$ be the input or control, $y(t) : \mathbb{R} \mapsto \mathbb{R}^q$ be the output at time variable t and $x(t_0)$ is the initial condition of the system. Further, The system matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, $C \in \mathbb{R}^{q \times n}$ are called respectively, the system matrix, the input coefficient matrix, and the output coefficient matrix, and $D \in \mathbb{R}^{q \times p}$ is the direct transmission map. Moreover, we have $D = 0$ in most of our applications. The first equation is also referred to as state equation whereas the second is called the output equation.

If $p = q = 1$, i.e. the input $u(\cdot)$ and the output $y(\cdot)$ are both scalar functions, the system (1.3) is called a Single-Input Single-Output (SISO) system. On the otherhand the system is called a Multi-Input Multi-Output (MIMO) system if $p, q > 1$. For MIMO systems, the number of state variables n may be very large than the number of input p and the number of output m (i.e. $p, q \ll n$).

The components in $x(t)$ have physical meaning from discretized while the components in inputs $u(t)$ and outputs $y(t)$ have also physical meaning of the device

being modeled. Relation between the inputs and outputs are determined by the coefficient matrices B and C .

Usually, the LTI system appears from the modeling of an applications process. The mathematical model describes the behavior of physical system or device in terms of a set of mathematical equations, with schematic diagram of the device connection. The system matrices may depend on time as well. If the system matrices are depending on the state x or the control u as well the system is said to be *nonlinear*. We will concentrate on the LTI case here.

The following theorem gives the solution of first order LTI system.

Theorem 1.1. *The solution of the continuous time dynamical system (1.3) is*

$$x(t) = e^{A(t-t_0)}x_0 + \int_{t_0}^t e^{A(t-s)}Bu(s)ds, \quad (1.4a)$$

$$\text{and } y(t) = Ce^{A(t-t_0)}x_0 + \int_{t_0}^t Ce^{A(t-s)}Bu(s)ds + Du(t) \quad (1.4b)$$

Proof. The continuous time dynamical system (1.3) can be written as

$$\dot{x}(t) - Ax(t) = Bu(t) \quad (1.5)$$

Multiplying this equation by $e^{A(t-s)}$, and then integrating both sides, we get

$$\begin{aligned} \int_{t_0}^t \frac{d}{ds}[e^{A(t-s)}x(s)]ds &= \int_{t_0}^t e^{A(t-s)}Bu(s)ds \\ \Rightarrow [e^{A(t-s)}x(s)]_{t_0}^t &= \int_{t_0}^t e^{A(t-s)}Bu(s)ds \\ \Rightarrow x(t) - e^{A(t-t_0)}x_0 &= \int_{t_0}^t e^{A(t-s)}Bu(s)ds \\ \Rightarrow x(t) &= e^{A(t-t_0)}x_0 + \int_{t_0}^t e^{A(t-s)}Bu(s)ds \end{aligned}$$

By substituting the value of $x(t)$ in $y(t) = Cx(t) + Du(t)$ we get

$$y(t) = Ce^{A(t-t_0)}x_0 + \int_{t_0}^t Ce^{A(t-s)}Bu(s)ds + Du(t)$$

□

For the continuous-time system (1.3), the dynamical system responses $x(t)$ and $y(t)$ for $t \geq t_0$ can be determined from the formulas in (1.4). In order to study the behavior of a dynamical system, it is customary to determine the responses of the system due to different inputs.

Definition 1.2. The matrix exponential e^{At} is defined as

$$e^{At} = \sum_{m=0}^{\infty} \frac{(At)^m}{m!} \quad (1.6)$$

The matrix exponential is an important tool in system theory. Some properties of the matrix e^{At} are given below:

1. $e^{A(t+s)} = e^{At}e^{As}$
2. $e^{(A+B)t} = e^{At}e^{Bt}$ iff $AB = BA$
3. e^{At} is non singular and $(e^{At})^{-1} = e^{-At}$
4. $e^{U^{-1}AUt} = U^{-1}e^{At}U$

1.3.2 Generalized Systems

Eventually, the system (1.3) appears the special form, known as *generalized state space form* as:

$$\Sigma : \begin{cases} Ex(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t) \end{cases} \Leftrightarrow \Sigma := \left(\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right) \quad (1.7)$$

where E, A, B, C, D are of appropriate dimensions. Here, E is invertible and symmetric positive definite, known as mass matrix. If $E = I$, the system is same as the standard system (1.3). Again, since E is invertible, one can convert the differential part of the system (1.7) into the standard system of the form

$$\begin{aligned} \dot{x}(t) &= \bar{A}x(t) + \bar{B}u(t); \\ y(t) &= Cx(t) + Du(t), \end{aligned}$$

where $\bar{A} = E^{-1}A$ and $\bar{B} = E^{-1}B$.

1.3.3 Descriptor Systems

Recently, many model application arises the special form of the generalized system (1.7), where as E is singular, i.e., $\det(E) = 0$. Such system appears in the process of modeling electrical circuits in chip design, power systems, chemical engineering, piezo-mechanical systems and so on (see [2, 30]). The system is then known as descriptor systems. Sometimes, it is known as Differential Algebraic Equations (DAEs) or singular systems. A descriptor system can be solved if the corresponding matrix pencil is regular i.e.,

$$(\lambda E - A) \neq 0 \quad (1.8)$$

In terms of this assumption, there exists nonsingular matrices S and T such that the pencil has the following Weierstrass canonical form

$$E = S \begin{bmatrix} I & 0 \\ 0 & N \end{bmatrix} T \quad \text{and} \quad A = S \begin{bmatrix} J_1 & 0 \\ 0 & I_n \end{bmatrix} T \quad (1.9)$$

where N is nilpotent matrix so that $N^{p-1} \neq 0$ but $N^p = 0$. The nilpotency p indicates the index of the descriptor system. The case of DAEs and their derivation is discussed in [31].

In this thesis we focus on the special structured descriptor system of the form

$$\underbrace{\begin{bmatrix} E_1 & E_2 \\ 0 & 0 \end{bmatrix}}_E \underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_A \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_B u(t) \quad (1.10a)$$

$$y(t) = \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_C \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + D_a u(t), \quad (1.10b)$$

where $x_1(t) \in \mathbb{R}^{n_1}$, $x_2(t) \in \mathbb{R}^{n_2}$ ($n = n_1 + n_2$) and J_1, J_2, J_3, J_4 are block matrices of A .

The descriptor system (1.10) is called

1. index-1 if $\det(J_4) \neq 0$
2. index-2 if $J_4 = 0$ and $\det(J_3 J_2) \neq 0$ and
3. index 3 if $J_4 = 0$ and $\det(J_3 J_2) = 0$.

1.3.4 Input and Output Relation

In the time domain analysis, two most common inputs are the step response and the frequency response. The generalized dynamical system (1.7) can be expressed in frequency domain. Applying Laplace transformation ¹, the linear system yields,

$$sEX(s) - x_0 = AX(s) + BU(s) \quad (1.11a)$$

$$\text{and } Y(s) = CX(s) + DU(s), \quad (1.11b)$$

where $X(s)$, $U(s)$ and $Y(s)$ are the Laplace transformation of $x(t)$, $u(t)$ and $y(t)$ respectively.

Put $x_0 = 0$ in (1.11a) and inserting $X(s)$ into (1.11b), we obtain

$$X(s) = (sE - A)^{-1}BU(s), \quad (1.12a)$$

$$\text{and } Y(s) = (C(sE - A)^{-1}B + D)U(s) \quad (1.12b)$$

$$\text{or } Y(s) = G(s)U(s) \quad (1.12c)$$

where

$$G(s) = C(sE - A)^{-1}B + D \quad (1.13)$$

is called the transfer function of the system (1.7). In MIMO systems, $G(s)$ is the $p \times q$ matrix, can be defined as

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1q}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2q}(s) \\ \vdots & \vdots & \cdots & \vdots \\ G_{p1}(s) & G_{p2}(s) & \cdots & G_{pq}(s) \end{bmatrix} \quad (1.14)$$

where $G_{mn} = C(m, :)(sE - A)^{-1}B(:, n) + D_a(m, n)$ with $n = 1, 2, \dots, p$ and $m = 1, 2, \dots, q$. In fact, the transfer function is the input-output relation of the dynamical systems. In system and control theory, the error bound of the reduced model is established through the transfer function.

Definition 1.3. The transfer function $G(s)$ is called proper if $\lim_{s \rightarrow \infty} G(s) < \infty$, and strictly proper if $\lim_{s \rightarrow \infty} G(s) = 0$. Otherwise $G(s)$ is called improper. The points p at which $G(p) = \infty$ are called the poles of the system.

¹The Laplace transformation of a function $f(t)$, is defined by $F(s) = L[f(t)] = \int_0^{\infty} f(t)e^{-st} dt$ where $t \geq 0, s \in \mathbb{C}$.

1.3.5 Gramians

Controllability and observability are two major concepts in modern control system theory. The ideas of controllability and observability of a system play crucial roles in the MOR methods. The gramian based MOR methods are mainly emerged on the system controllability gramian and observability gramian [32].

Definition 1.4. The dynamical system (1.3) is said to be controllable if for any initial state $x(0) = x_0, t_f > 0$ and final state x_f , there exists a input $u(t)$ such that the solution (1.4) satisfies $x(t_f) = x_f$. Otherwise, the system is known as uncontrollable.

Alternatively, if a linear system (A, B, C, D) is controllable to the zero state if there exists an input function $u(t)$ and a time $t_1 < \infty$, such that the solution of the linear system vanishes at time t_1 , i.e., $\phi(u; x; t_1) = 0$. The system is completely controllable if $X^{contr} = \mathbb{R}^n$, where X^{contr} is the set of all controllable states.

The following theorem shows that the controllability of a system can be verified through some algebraic criteria.

Theorem 1.5. *The following are equivalent for the matrix pair (A, B) of the system (1.3):*

1. (A, B) is controllable.
2. The controllability matrix $\mathcal{C}(A, B) = [B, AB, A^2B, \dots, A^{n-1}B]$ has full rank.
3. The controllability gramian

$$P = \int_0^{t_f} e^{At} B B^T e^{A^T t} dt,$$

is positive definite for any $t > 0$.

4. The matrix $[A - sI, B]$ has full rank n for all $s \in \mathbb{C}$.
5. The pair (\tilde{A}, \tilde{B}) is controllable, where $\tilde{A} = TAT^{-1}$ and $\tilde{B} = TB$ for any non singular $T \in \mathbb{R}^{n \times n}$.

A proof of this theorem is available in [5, 32].

Observability is the dual concept of the controllability.

Definition 1.6. The dynamical system (1.3) is said to be observable if for any $t_f > 0$, initial state $x(0) = x_0$ can be uniquely determined from the time of the input $u(t)$ and the output $y(t)$ in the interval of $[0, t_f]$. Otherwise, the system is said to be uncontrollable.

Alternatively, a linear system (A, B, C, D) is unobservable if $y(t) = 0$ for all $t \geq 0$. The system is completely observable if $X^{unobs} = 0$, where X^{unobs} is the set of all unobservable states of the system.

Theorem 1.7. *The following are equivalent for the matrix pair (A, B) of the system (1.3):*

1. (C, A) is observable.

2. The observability matrix $\mathcal{O}(A, C) = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$ has full rank.

3. The controllability gramian

$$Q = \int_0^{t_f} e^{A^T t} C^T C e^{A t} dt,$$

is positive definite for any $t > 0$.

4. The matrix $\begin{bmatrix} A - sI \\ C \end{bmatrix}$ has full rank n , for all $s \in \mathbb{C}$.

5. There exists a similarity transformation T , such that $TAT^{-1} = \begin{bmatrix} \tilde{A}_{11} & 0 \\ \tilde{A}_{12} & \tilde{A}_{22} \end{bmatrix}$ and $CT^{-1} = [\tilde{C}_1 \ 0]$, where $(\tilde{C}_1, \tilde{A}_{11})$ is observable.

A proof of this theorem is available in [5, 32].

We now show that controllability and observability are characterized by the controllable matrix and observable matrix.

For details, let $x(0) = x_0 = 0$. Then the solution (1.4) yields,

$$\begin{aligned} x(t) &= \int_0^t e^{A(t-\iota)} Bu(\iota) d\iota \\ &= \int_0^t \left\{ I + A(t-\iota) + \frac{A^2}{2}(t-\iota)^2 + \dots \right\} Bu(\iota) d\iota \\ &= B \int_0^t u(\iota) d\iota + AB \int_0^t (t-\iota)u(\iota) d\iota + A^2B \int_0^t \frac{(t-\iota)^2}{2} u(\iota) d\iota + \dots \end{aligned}$$

But the expression is the linear combination of the matrices $B, AB, A^2B, \dots, A^{n-1}B$. Therefore, the system (1.3) is controllable if every state of the system is controllable i.e., $\mathcal{C}(A, B)$ has full rank.

Dually, the system (1.3) is observable if the observable matrix $\mathcal{O}(A, C)$ has full rank. A system is called complete if it is both controllable and observable.

1.3.6 Stability

Stability is an important properties of dynamical systems. Some basic concept of the stability of a system are discussed below.

Definition 1.8. The matrix A is called stable or Hurwitz-stable if all its eigenvalues are located in the open left half of the complex plane, i.e., $\lambda \in \mathbb{C}^-$. Like wise, the system is unstable if any eigenvalues of A lies in \mathbb{C}^+ (right complex plane).

The LTI system (1.3) is called stabilizable if there exists a matrix $L \in \mathbb{R}^{m \times n}$ such that $A - BL$ is stable.

In control theory, following theorem known as Lyapunov stability, is the most useful criteria in the analysis to measure the stability of a system.

Theorem 1.9 (Lyapunov Stability Theorem). *The system (1.3) is asymptotically stable if and only if for any symmetric positive definite matrix M , there exists a unique symmetric positive definite matrix X satisfying the Lyapunov equation:*

$$AX + XA^T = -M \tag{1.15}$$

The proof is given in [5, 32].

The controllability and observability gramians of the system can also be examined through the infinite gramians P and Q , defined as

$$P = \int_0^{\infty} e^{At} B B^T e^{A^T t} dt, \quad (1.16)$$

$$\text{and } Q = \int_0^{\infty} e^{A^T t} C^T C e^{At} dt \quad (1.17)$$

Now, we show an important theorem for this thesis. The following theorem shows the relation of the stability, controllability and observability of a system.

Theorem 1.10. *The controllability gramian P and observability gramian Q is the solution of the continuous-time algebraic Lyapunov equations of the form*

$$AP + PA^T + BB^T = 0, \quad (1.18)$$

$$A^T Q + QA + C^T C = 0 \quad (1.19)$$

Proof. Putting the value of controllability gramian $P = \int_0^{\infty} e^{At} B B^T e^{A^T t} dt$ in the Lyapunov equation (1.18), we get

$$\begin{aligned} AP + PA^T + BB^T &= A \int_0^{\infty} e^{At} B B^T e^{A^T t} dt + \int_0^{\infty} e^{At} B B^T e^{A^T t} dt A^T + BB^T \\ &= \int_0^{\infty} \underbrace{A e^{At} B B^T e^{A^T t} + e^{At} B B^T e^{A^T t} A^T}_{= \frac{d}{dt} e^{At} B B^T e^{A^T t}} dt + BB^T \\ &= \lim_{t \rightarrow \infty} e^{At} B B^T e^{A^T t} - e^{A \cdot 0} B B^T e^{A^T \cdot 0} + BB^T \\ &= -BB^T + BB^T \\ &= 0 \end{aligned}$$

In a similar way, the prove can be shown for the observability gramian. □

Again, it can easily be shown that the controllability gramian P and the observability gramian Q , are also the solution of the continuous-time generalized Lyapunov equations

$$APE^T + EPA^T + BB^T = 0, \quad (1.20)$$

$$A^T QE + E^T QA + C^T C = 0 \quad (1.21)$$

For stable systems, the both gramians can be interpreted in the following way:

(i) for controllability gramian the minimum energy

$$J(u) = \int_{-\infty}^0 u^*(t)u(t)dt, \quad x(0) = x_0, t \leq 0, \quad (1.22)$$

of the input is equivalent to $J(u) = x_0^* P^{-1} x_0$. Thus, states in the span of the eigenvectors corresponding to small eigenvalues of P are difficult to reach.

(ii) for observability gramian a system emerged from $x(0) = x_0$ with $u(t) = 0, t \geq 0$ has

$$\int_0^{\infty} y^*(t)y(t)dt = x_0^* Q x_0 \quad (1.23)$$

It follows that the state in the span of the eigenvectors corresponding to small eigenvalues of Q are difficult to observe.

1.3.7 Hankel Singular Values

In control theory, Hankel Singular Values (HSVs) are considered as a measure of energy for each state in a system. The HSVs play a crucial role in the balancing based model reduction. The HSVs are the basis of balanced model reduction, in which low energy states are discarded while high energy states are preserved.

The Hankel operator maps inputs $u(t), t < 0$ to output $y(t), t > 0$:

$$\mathcal{H} : u(t) \longrightarrow \int_{-\infty}^0 C e^{A(t-s)} B u(s) ds.$$

The fact that the Hankel operator has a finite number of singular values, which are a good useful in control theory and model reduction.

It can be shown that for stable complete system, the HSVs are the positive square roots of the eigenvalues of the product of the controllability and observability gramians, i.e.,

$$\sigma_i = \sqrt{\lambda_i(PQ)} = \sqrt{\lambda_i(QP)}, \quad i = 1, 2, \dots, n, \quad (1.24)$$

where λ_i denotes the eigenvalues.

Since the controllability gramian and the observability gramian are symmetric positive definite, they have always Cholesky decomposition:

$$P = \mathcal{R}_c \mathcal{R}_c^T \quad \text{and} \quad Q = \mathcal{L}_c \mathcal{L}_c^T \quad (1.25)$$

Therefore,

$$\begin{aligned}
 \sigma_i(PQ) &= \sqrt{\lambda_i(PQ)} \\
 &= \sqrt{\lambda_i(\mathcal{R}_c \mathcal{R}_c^T \mathcal{L}_c \mathcal{L}_c^T)} \\
 &= \sqrt{\lambda_i((\mathcal{R}_c^T \mathcal{L}_c)^T (\mathcal{R}_c^T \mathcal{L}_c))} \\
 &= \sigma_i(\mathcal{R}_c^T \mathcal{L}_c) \quad \text{for } i = 1, 2, \dots, n,
 \end{aligned} \tag{1.26}$$

where σ_i represents the singular values of $\mathcal{R}_c^T \mathcal{L}_c$. Therefore, one can use the gramian factors replaced by full gramian to compute the system HSVs.

The HSVs for a system and its transfer function are defined to be the same. HSVs are invariant under state space transformations, since similarity of PQ is preserved under state-space transformations and are so called input-output invariants.

1.3.8 Realizations

We know that the transfer function matrix is

$$G(s) = C(sE - A)^{-1}B + D$$

The state space representation Σ in (1.7) have a realization if the matrices E, A, B, C, D satisfy the transfer function $G(s)$.

It can be shown that the transfer function $G(s)$ is invariant of an LTI system (1.7) under the following coordinate transformations

$$\mathcal{T} : \begin{cases} x & \rightarrow Tx, \\ (E, A, B, C, D) & \rightarrow (\mathcal{T}E\mathcal{T}^{-1}, \mathcal{T}A\mathcal{T}^{-1}, \mathcal{T}B, C\mathcal{T}^{-1}, D), \end{cases} \tag{1.27}$$

where \mathcal{T} is nonsingular. Then the transfer function can be replaced by the non singular transformation matrix \mathcal{T} .

For a transfer function $\tilde{G}(s)$, we get

$$\begin{aligned}
 \tilde{G}(s) &= (C\mathcal{T}^{-1})(s\mathcal{T}E\mathcal{T}^{-1} - \mathcal{T}A\mathcal{T}^{-1})^{-1}(\mathcal{T}B) + D \\
 &= C(sE - A)^{-1}B + D \\
 &= G(s)
 \end{aligned}$$

Therefore, the new generalized state space system is

$$\mathcal{T}E\mathcal{T}^{-1}\mathcal{T}x(t) = \mathcal{T}A\mathcal{T}^{-1}\mathcal{T}x(t) + \mathcal{T}Bu(t) \quad (1.28)$$

$$y(t) = C\mathcal{T}^{-1}\mathcal{T}x(t) + Du(t). \quad (1.29)$$

Definition 1.11. A state space realization of a transfer function $G(s)$ is minimal if and only if the system is controllable and observable.

Definition 1.12. A realization (E, A, B, C, D) of a stable linear system Σ is called balanced if its controllability gramian P and observability gramian Q such that

$$P = Q = \text{diag}\{\sigma_1, \dots, \sigma_n\}, \quad \text{where } \sigma_j \geq \sigma_{j+1}, j = 1, \dots, (n-1).$$

Theorem 1.13. *The HSVs of a stable minimal linear systems are invariant.*

Proof. We know that the HSVs of a system are the positive square roots of the eigenvalues of the product of the gramians P and Q .

Let $(\hat{A}, \hat{B}, \hat{C}, D) = (TAT^{-1}, TB, CT^{-1}, D)$ be a transformation, then the associative controllability Lyapunov equation becomes

$$\begin{aligned} 0 &= \hat{A}\hat{P} + \hat{P}\hat{A}^T + \hat{B}\hat{B}^T \\ &= TAT^{-1}\hat{P} + \hat{P}T^{-T}A^TT^T + TBB^TT^T \\ &= A(T^{-1}\hat{P}T^{-T}) + (T^{-1}\hat{P}T^{-T})A^T + BB^T \\ &= AP + PA^T + BB^T \end{aligned}$$

Then, $P = T^{-1}\hat{P}T^{-T}$ gives $\hat{P} = TP T^T$.

Similarly, for observability Lyapunov equation, one can yields $\hat{Q} = T^{-T}QT^{-1}$.

Therefore, $\hat{P}\hat{Q} = (TP T^T)(T^{-T}QT^{-1}) = TPQT^{-1}$.

This shows that $\Lambda(\hat{P}\hat{Q}) = \Lambda(PQ) = \{\sigma_1^2, \sigma_2^2, \dots, \sigma_n^2\}$. □

For non-minimal systems $\Lambda(\hat{P}\hat{Q}) = \{\sigma_1^2, \sigma_2^2, \dots, \sigma_n^2, 0, \dots, 0\}$.

However, such realizations are not unique. In particular, we want to find a minimal realization in this thesis.

1.4 Background in Linear Algebra

1.4.1 Eigenvalue Problem

The eigenvalue problem is to find $\lambda \in \mathbb{C}$ and $x \in \mathbb{C}^n$ that satisfy

$$Ax = \lambda x, \quad x \neq 0,$$

where $A \in \mathbb{C}^{n \times n}$ is a complex matrix. The scalar $\lambda \in \mathbb{C}$ is called an eigenvalue of A that is, $\det(A - \lambda I)x = 0$. The vector x is called an eigenvector for λ . The pair (λ, x) is also referred to as an eigenpair of A .

Any Hermitian matrix $A = A^T \in \mathbb{R}^{n \times n}$ has real eigenvalues and an orthonormal basis of real eigenvectors.

Definition 1.14. The set of all eigenvalues of A is called the spectrum of A , denoted by $\Lambda(A)$. The maximum modulus of the eigenvalues is called spectral radius and is denoted by $\rho(A)$, i.e. $\rho(A) = \max_{\lambda \in \rho(A)} |\lambda|$.

The trace of a matrix is equal to the sum of all its diagonal elements. It can be easily shown that the trace of A is also equal to the sum of the eigenvalues of A . Also the set of eigenvalues of a triangular matrix are its diagonal entries.

If λ is an eigenvalue of a matrix A then $\bar{\lambda}$ is an eigenvalue of A^H . An eigenvector v of A^H associated with the eigenvalue $\bar{\lambda}$ is called a left eigenvector of A . If (λ, x) is an eigenpair of A , then $(\lambda - s, x)$ is an eigenpair of $(A - sI)$ where s is a scalar.

Lemma 1.15. *A matrix A is diagonalizable if and only if there exists a nonsingular $X \in \mathbb{C}^{n \times n}$ such that $X^{-1}AX = \text{diag}(\lambda_1, \dots, \lambda_n)$, where λ_i ($i = 1, \dots, n$) are the eigenvalues of A and the columns of X are eigenvectors of A .*

If all λ_i are distinct, then there are n independent eigenvectors. The generalized eigenvalue problem is of the form $Ax = \lambda Bx$, $x \neq 0$, where A and B are $n \times n$ complex matrix.

The scalar $\lambda \in \mathbb{C}$ is called an eigenvalue of the pair (A, B) if $\det(A - \lambda B)x = 0$. The set of all eigenvalues of (A, B) is called the spectrum of (A, B) , denoted by $\Lambda(A, B)$.

If A and B are Hermitian and B is positive definite, then all eigenvalues of (A, B) are real and there exists a nonsingular $X \in \mathbb{C}^{n \times n}$ such that $X^*AX = \text{diag}(\lambda_1, \dots, \lambda_n)$ and $X^*BX = I$, where $\lambda_i (i = 1, \dots, n)$ are the eigenvalues of (A, B) .

Definition 1.16. A square symmetric matrix A is said to be positive definite if $x^*Ax > 0$ for every non-zero column vector x of real numbers. Similarly, a square Hermitian matrix A is said to be positive definite (semidefinite) if $x^*Ax > 0$ for every non-zero column vector x of complex numbers.

For a given linear transformation $T : \mathcal{C}^n \mapsto \mathcal{C}^n$, a subspace $S \subset \mathcal{C}^n$ is said to be invariant if $Ax \in S$ for every $x \in S$. For example: $0, \mathcal{C}^n, \text{Ker}T$ and $\text{Im}T$ are all T -invariant subspaces.

1.4.2 Sparse and Dense Matrices

In many application, there exists some matrix with special structure that has a small number of nonzero entries.

Definition 1.17. Let A be a matrix such that most of the elements are zero, called sparse matrix. On the otherhand, A is called dense matrix if most of the elements are nonzero.

The number of zero-valued elements divided by the total number of elements is called the sparsity of the matrix. For example: Consider a matrix $M_{10 \times 9}$ contains only 18 nonzero elements and 72 zero elements. Then, its sparsity is 80 and density is 20.

Large sparse matrices often appear in scientific or engineering applications when solving partial differential equations. It is favorable and often necessary to use specialized algorithms and data structures. Other hand, dense matrix structures and algorithms are slow and inefficient for large scale system.

Definition 1.18. A projector matrix P is a square matrix such that $P = P^2$, also known as idempotent matrix.

Now $(I - P)^2 = I - 2P + P^2 = I - 2P + P = I - P$, then $I - P$ is called complementary projector to P .

Definition 1.19. If a projector projects onto a subspace S_1 along a subspace S_2 then it is called orthogonal projector.

1.4.3 Matrix Decompositions

Matrix decomposition or matrix factorization is a factorization of a matrix into a product of matrices. Now, we introduce some important decomposition which are used in many applications.

1.4.3.1 Eigenvalue Decomposition

Eigenvalue decomposition is the factorization of a matrix into a canonical form, whereas the matrix is represented in terms of its eigenvalues and eigenvectors.

Let $A \in \mathbb{C}^{n \times n}$ be a square matrix with n linearly independent vectors. The eigenvalue decomposition of A is defined as

$$A = V\Lambda V^{-1}$$

where $\Lambda \in \mathbb{C}^{n \times n}$ is an diagonal matrix whose entries are the eigenvalues of A and the columns of $V \in \mathbb{C}^{n \times n}$ is the eigenvectors of A .

1.4.3.2 Singular Value Decomposition

Singular Value Decomposition (SVD) is one of the most useful matrix decomposition used in linear control systems and model reduction techniques.

Let $A \in \mathbb{C}^{m \times n}$ be a matrix, the SVD of A is a factorization

$$A = U\Sigma V^*$$

where $U \in \mathbb{C}^{m \times m}$, $V \in \mathbb{C}^{n \times n}$ are unitary matrices and the diagonal entries σ_j of $\Sigma \in \mathbb{R}^{m \times n}$ are called singular values of A , which are non-negative and in decreasing order, i.e., $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k \geq 0$, $k = \min(m, n)$.

The thin SVD of A is obtained by taking only the first m singular values. This decomposition gives the following properties:

1. The singular values σ_j are the square roots of the eigenvalues of the symmetric positive semi-definite matrix $A^T A$.
2. If l be the number of singular values, $rank(A) = l$.
3. For $A = A^*$, $\sigma_j = \|\lambda(A)\|$.
4. For $A \in \mathbb{C}^{m \times m}$, $\prod_{j=1}^m \sigma_j = det(A)$.
5. $\|A\|_2 = \sigma_1$ and $\|A\|_F = \sqrt{(\sigma_1^2 + \sigma_2^2 + \dots + \sigma_r^2)}$.

MATLAB Commands: $[U, \Sigma, V] = svd(A)$.

1.4.3.3 Schur Decomposition

Given a square matrix $A \in \mathbb{C}^{m \times n}$, the Schur decomposition is defined as

$$A = UTU^*$$

where $U \in \mathbb{C}^{m \times m}$ is unitary matrix and T is upper triangular matrix. Since T is similar to A and triangular, the eigenvalues of A are the diagonal entries of T .

MATLAB Commands: $[T] = schur(A)$.

1.4.3.4 QR Decomposition

Let $A \in \mathbb{C}^{m \times n}$ be a matrix, this decomposition is defined by

$$A = QR$$

where Q is an orthogonal matrix and R is an upper triangular matrix.

In general, QR decomposition is used to solve the eigenvalue problem and least squares problem. There are several methods for computing the QR decomposition, such as modified Gram-Schmidt process, Householder transformations.

MATLAB Commands: $[Q, R] = qr(A)$.

1.4.3.5 Cholesky Decomposition

The Cholesky decomposition is a factorization of a Hermitian, positive-definite matrix into the product of a lower triangular matrix and its conjugate transpose.

Let $A \in \mathbb{C}^{m \times n}$ be a Hermitian and positive-definite matrix. The Cholesky decomposition of A is defined as

$$A = LL^*$$

where L is a lower triangular matrix with real and positive diagonal entries and L^* its conjugate transpose. Cholesky Decomposition can be used to solve the system of linear equation $Ax = b$, where A is real symmetric and positive definite matrix.

MATLAB Commands: $[L] = chol(A)$.

Lemma 1.20. *Every Hermitian positive-definite matrix has a unique Cholesky decomposition.*

Now we discuss the popular way, so called Krylov-based Arnoldi process, to compute the eigenvalues for large sparse matrices.

1.4.3.6 Arnoldi Decomposition

The Arnoldi decomposition is the classical iterative solvers and an important example of iterative methods. We will now study a different class of iterative solvers based on optimization. The Arnoldi iteration method to be derived will be applicable to both linear systems and eigenvalue problems. It is a typical large sparse matrix algorithm which does not access the elements of the matrix directly, but rather makes the matrix map vectors. One of the main ingredients in all of the following methods are Krylov subspaces.

Definition 1.21. Consider $A \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{n \times p}$. Then the m th Krylov matrix associated with A and V is defined by

$$K_m(A, V) = [V, AV, \dots, A^{m-1}V]. \quad (1.30)$$

Clearly, matrix-vector products play a crucial role in generating this sequence since each subsequent vector in the sequence is obtain from the previous one by multiplication by A .

Algorithm 1: Arnoldi algorithm using modified Gram-Schmidt**Input** : A, B , orthogonal matrix V_m .**Output:** Matrix $R_m \in \mathbb{R}^{n \times m}$ such that $X_m = R_m R_m^T$.

```

1 Set  $u_1 = \frac{B}{\|B\|_2}$ ,  $U_1 = u_1$ 
2 for  $j = 1, 2, \dots, m$  do
3   Compute  $w_j = Au_j$ ;
4   for  $i = 1 : j$  do
5      $h_{ij} = u_i^T w_j$ ;
6      $w_j = w_j - h_{ij}u_i$ ;
7   end for
8   Compute  $h_{j+1,j} = \|w_j\|_2$  and  $u_{j+1} = \frac{w_j}{h_{j+1,j}}$ ;
9    $H_j = \begin{bmatrix} H_{j-1} & h_j \\ 0 & h_{j+1,j} \end{bmatrix}$ ;
10   $U_{j+1} = [U_j, u_{j+1}]$ ;
11  Partition  $H_m = \begin{bmatrix} H_m \\ h_{m+1,m}e_m^T \end{bmatrix}$ ;
12 end for

```

Theorem 1.22. Let the column of $V_{m+1} = [V_m, v_{m+1}]$ form an orthogonal basis then there exists an upper Hessenberg matrix $\hat{H}_m \in \mathbb{R}^{m+1 \times m}$, defined as

$$\hat{H}_m = \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & h_{1m} \\ h_{21} & h_{22} & \cdots & \cdots & h_{2m} \\ 0 & h_{32} & \cdots & \cdots & h_{3m} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{m,m-1} & h_{mm} \\ 0 & 0 & \cdots & \cdots & 0 & h_{m+1,m} \end{bmatrix} \quad (1.31)$$

such that

$$AV_m = V_{m+1}\hat{H}_m. \quad (1.32)$$

Conversely, if a matrix V_{m+1} of orthogonal columns satisfies (1.32) then the columns of V_{m+1} form a basis for the Krylov subspace K_m .

An important property is given below for Hessenberg matrix:

$$AV_m = \begin{bmatrix} V_m & v_{m+1} \end{bmatrix} \begin{bmatrix} H_m \\ h_{m+1,m}, e_m^T \end{bmatrix} = V_m H_m + h_{m+1,m} v_{m+1} e_m^T, \quad (1.33)$$

where H_m can be found by removing the last row from \hat{H}_m and e_m is a matrix of the last p columns of the mp identity matrix.

After m steps $h_{m+1,m}$ will be vanishes. Thus after some iteration, the second term of the right side of (1.33) converge to zero. Therefore, we can rewrite (1.33) as

$$H_m = V_m^T A V_m. \quad (1.34)$$

Hence, it is clear that H_m represents the projection A onto the subspace $\mathcal{K}_m(A, V)$.

Definition 1.23. The eigenvalues λ_i of H_m is called a *Ritz values* and if \bar{v} is an eigenvector of H_m associate with λ , then $V_m \bar{v}$ is called a *Ritz vector* belong to λ .

1.4.4 System Norms

This subsection presents the norm of vectors and matrices, because these are useful in the discussion of stability of the algorithm, the stopping criteria and convergence analysis of the iterative methods. We refer to [3] and [2] for motivation.

1.4.4.1 Vector Norms

Let X be a vector space. The norm on X is a real valued function on X if for any $x \in X$ and $y \in Y$, it satisfies the following properties:

1. $\|x\| \geq 0$ and $\|x\| = 0$ iff $x = 0$,
2. $\|x + y\| \geq \|x\| + \|y\|$,
3. $\|\alpha x\| = |\alpha| \|x\|$, for any $\alpha \in \mathbb{R}$

The p norm of $x \in \mathbb{C}$ is defined as

$$\|x\|_p := \left(\sum_{i=1}^n \|x_i\|^p \right)^{1/p}, \quad \text{for } 1 \leq p < \infty$$

When $p = 1, 2, \dots, \infty$, the norm can be defined as:

$$\begin{aligned} \|x\|_1 &:= \sum_{i=1}^n |x_i|, \\ \|x\|_2 &:= \sqrt{\sum_{i=1}^n |x_i|^2} = \sqrt{x^T x}, \\ \|x\|_\infty &:= \max_{1 \leq i \leq n} |x_i|. \end{aligned}$$

1.4.4.2 Matrix Norms

Let $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ be a matrix, then the matrix norm of A denoted by $\|A\|$, is a mapping from $\mathbb{R}^{n \times n}$ to \mathbb{R} satisfies the following properties:

1. $\|A\| \geq 0$, if $A \neq 0$.
2. $\|\alpha A\| = |\alpha| \|A\|$, for any $\alpha \in \mathbb{R}$.
3. $\|A + B\| \leq \|A\| + \|B\|$.

We usually prefer matrix norms of a matrix $A = [a_{ij}] \in \mathbb{C}^{n \times n}$ induced by a vector p -norm is defined as

$$\|A\|_p := \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p}.$$

In particular, the induced matrix 1-norm and 2-norm can be computed as

$$\begin{aligned} \|A\|_1 &:= \max \|a_j\|_1, \quad a_j \text{ is the } j\text{th column of } A \\ \|A\|_2 &:= \sqrt{\lambda_{\max}(A^*A)}. \end{aligned}$$

Another important matrix norm is so called Frobenius norm.

Definition 1.24. The Frobenius norm $\|\cdot\|_F$ of a matrix $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ is defined as

$$\|A\|_F := \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} = \sqrt{\text{tr}(AA^*)} = \sqrt{\text{tr}(A^*A)}$$

It can be shown that

$$\begin{aligned} \|AB\|_F &\leq \|A\|_F \cdot \|B\|_F, \\ \sqrt{\rho(AA^*)} &\leq \|A\|_F \leq \sqrt{n} \sqrt{\rho(AA^*)} \end{aligned}$$

Definition 1.25. \mathcal{H}_∞ is a closed subspace of scalar valued functions and the norm is defined as

$$\begin{aligned} \|G\|_\infty &= \sup_{\text{Re}(s) > 0} \sigma |G(s)| \\ &= \sup_{w \in \mathcal{R}} \sigma |G(jw)| \end{aligned}$$

1.5 Existing Methods

Lyapunov equation has fundamental role in many applications areas such as system and control theory [12]. In this section, we introduce some standard methods which solve small dense Lyapunov equation, e.g., Bartels-Stewart methods, Hammarling methods and sign function methods. The first two are based on the Schur decomposition.

Later, iterative methods are very useful for large scale sparse problems because they are more suitable than direct methods and often do not destroy sparsity. In this section, we briefly review some iterative methods for generalized Lyapunov equations, e.g., Krylov subspace methods and LRCF-ADI iterations methods. Most of the methods use Galerkin projection technique to produce low-dimensional Lyapunov equations that are solved by using direct methods.

1.5.1 Bartels-Stewart's Method

The Bartels-Stewart method [33, 34] provided the first numerically standard technique to solve the dense small-to-medium scale ($n \leq 50$) Lyapunov equations. The main idea of the Bartels-Stewart algorithm is to transform A into a real Schur decomposition $H = U^T A U$ where U is orthogonal and H is quasi upper-triangular, while in the Hessenberg-Schur algorithm, A is reduced only to upper Hessenberg form. Then Lyapunov equation (1.15) can be written as

$$H\tilde{P} + \tilde{P}H^T + U^T B B^T U = 0. \quad (1.35)$$

This equation can be solved efficiently by backward substitutions so that $\tilde{P} = U^T P U$.

By Schur's lemma every matrix is unitarily similar to a triangular matrix such that $A = U R U^T$, $B = V S V^T$, where U, V are unitary matrices, R is upper triangular matrix and S is lower triangular matrix.

It transforms the system matrix A to real Schur form, and then back solves for the solution of the transformed Lyapunov equation. The solution X is then obtained by a congruence transformation. Reducing a general, possibly sparse matrix to

real Schur form requires work, as does the congruence transformation to produce X .

Unfortunately, this methods is not suitable for solving large Lyapunov equations, since the computation time and storage requirements are prohibitive. The Schur decompositions of sparse matrices are dense as for orthogonal transformation. Besides, the respective solution will be dense. To remedy this situation, we presents some well established methods in the next subsection.

1.5.2 Hammarling's Method

The Hammarling method [35] is merely another exact method, applies to Lyapunov equations (1.15) in which the right side ($M = BB^T$) is symmetric positive semidefinite.

The algorithm first transforms A to lower triangular form. It calculates recursively for the lower triangular matrix Cholesky factor of the solution X rather than X itself, i.e., $X = RR^T$. It requires $O(n^3)$ complexity.

Penzl in [36] generalized exactly the same technique to the generalized Lyapunov equation. All methods require Schur decomposition of A or Hessenberg decomposition of B , which requires $25n^3$ flops for Schur decomposition. Therefore, these methods are not feasible for large-scale problems with $n > 10000$.

1.5.3 Matrix Sign Function Method

One of the most popular approaches to solve large scale dense Lyapunov equations is the matrix sign function method.

The matrix sign function of A is defined as follows:

$$\text{sign}(A) = VDV^{-1}, \quad (1.36)$$

where $D = \text{diag}(d_1, d_2, \dots, d_n)$ and

$$d = \begin{cases} 1 & \text{Re}(\lambda_i) > 0 \\ -1 & \text{Re}(\lambda_i) < 0 \end{cases} \quad (1.37)$$

If X is a solution of the Lyapunov equation

$$\begin{bmatrix} I_n & -X \\ 0 & I_n \end{bmatrix} \begin{bmatrix} A & BB^T \\ 0 & A^T \end{bmatrix} \begin{bmatrix} I_n & X \\ 0 & I_n \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix}$$

Hence, if A is asymptotically stable, then

$$\text{sign}(H) = T \text{sign} \left(\begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix} \right) T^{-1} = \begin{bmatrix} -I_n & 2X \\ 0 & I_n \end{bmatrix}$$

The matrix sign function is compute by the following formula:

$$Z_0 = A = \begin{bmatrix} A & BB^T \\ 0 & A^T \end{bmatrix}, \quad Z_{k+1} = (Z_k + Z_k^{-1})/2, \quad k = 0, 1, 2, \dots \quad (1.38)$$

It can be shown that $Z_k \rightarrow \text{sign}(A)$ as $k \rightarrow \infty$.

Comparison of the matrix sign function method [37] to the generalized Bartels-Stewart and Hammarling methods with respect to the accuracy and computational cost can be found in [38].

The author in [39] applied this method with low rank approximation of the right side. There, it has been observed that the matrix sign function method is about as expensive as the Bartels-Stewart method and both methods require approximately the same amount of work space.

However, the matrix sign function method is more appropriate for parallelization than the generalized Bartels-Stewart method and is currently the only practicable approach to solve regular generalized Lyapunov equations with large scale dense coefficient matrices.

A disadvantage of the matrix sign function method is that a matrix inversion is required in every iteration step which may lead to significant roundoff errors for ill conditioned Z_k . Such difficulties may arise when eigenvalues of the pencil $\lambda E - A$ lie close to the imaginary axis or $\lambda E - A$ is nearly singular.

Note that if the matrix E is singular, then Z_k diverges for the pencil $\lambda E - A$ of index greater than two and converges to a singular matrix. Thus, the matrix sign function method cannot be directly utilized for projected generalized Lyapunov equations.

Algorithm 2: G-LRCF-ADI iteration**Input** : E, A, B, C, μ_i .**Output:** \mathcal{R} , such that $\mathcal{P} \approx \mathcal{R}\mathcal{R}^*$.

```

1 while  $\|W_{i-1}^T W_i\| \geq \text{tol}$  or  $i \leq i_{max}$  do
2   Compute  $V_i = (A + E)^{-1} W_{i-1}$ ;
3   if  $\text{Im}(\mu_i) = 0$  then
4      $Z_i = [Z_{i-1} \quad \sqrt{-2\mu_i} V_i]$ ;
5      $W_i = W_{i-1} - 2\mu_i E V_i$ ;
6   else
7      $\gamma = -2\text{Re}(\mu_i)$ ,  $\delta = \frac{\text{Re}(\mu_i)}{\text{Im}(\mu_i)}$ ;
8      $Z_{i+1} = [Z_{i-1} \quad \sqrt{2\gamma}(\text{Re}(V_i) + \delta \text{Im}(V_i)) \quad \sqrt{2\gamma(\delta^2 + 1)} \text{Im}(V_i)]$ ;
9      $W_{i+1} = W_{i-1} + 2\gamma E(\text{Re}(V_i) + \delta \text{Im}(V_i))$ ;
10     $i = i + 1$ ;
11  end if
12   $i = i + 1$ ;
13 end while

```

1.5.4 Low Rank Cholesky Factor-Alternating Direction Implicit Method

The Alternating Direction Implicit (ADI) methods are powerful techniques that arise from the solution methods for elliptic and parabolic partial differential equations, this method is competitive with the Bartels-Sewart and Hammarling methods. The low rank smith method [15] gives the same approximation as the ADI method and exploits the low rank of the right-hand side of the Lyapunov equation. The ADI iteration in [40] was first introduced for solving elliptic and parabolic difference equations.

Consider the continuous time Lyapunov equation

$$AX + XA^T = -BB^T. \quad (1.39)$$

Then the ADI iteration can be written as

$$(A + \mu_k I)X_{k-1/2} = -BB^T - X_{k-1}(A^T - \mu_k I), \quad (1.40)$$

$$(A + \mu_k I)X_k = -BB^T - X_{k-1/2}^T(A^T - \mu_k I), \quad (1.41)$$

with $X_0 = 0$ and the shift parameters $\mu_1, \mu_2, \dots, \mu_k \in \mathbf{C}^-$.

The authors in [41] extend this idea, known as Cholesky Factor-Alternating Direction Implicit (CF-ADI) iterations and showed that this method produces the same approximation as ADI method, but is much more efficient. Recently, LRCF-ADI iteration performs another efficient method which can be found in [9]. This iterative method is also extended in [30] for solving the generalized projected Lyapunov equations for descriptor systems. In [42], the authors explained an efficient technique to compute real low-rank gramian factors by cleverly handling the complex shift parameters. The most recent developments were performed in [28], the author concentrates the updated version of Generalized Sparse (GS)-LRCF-ADI iteration.

On the other hand, a computationally cheap approach and a low-rank residual based stopping criterion of the LRCF-ADI iteration is introduced in [43]. For convenience, we present an Algorithm 2 for the updated version of the generalized (G-)LRCF-ADI iteration.

1.5.5 Krylov Subspace Method

In this subsection, we review some basic idea of projection based low rank iterative methods for the solution of large scale matrix Lyapunov equations. Projection strategies reduce the problem dimension so that the reduced problem can be numerically solved by a method. In [20], the author seemed to propose the idea of projecting the Lyapunov equation to a smaller space.

A prominent iterative techniques for Lyapunov equations are Krylov subspace method [17, 18] which become competitive with ADI iteration due to recent developments on extended and rational Krylov subspace. Such projection method is based on the Krylov subspaces techniques via the block Arnoldi or Lanczos process, introduced first by [20], in case of $p = 1$. Next, in [17], Jaimoukha and Kasenally extended this method for large scale Lyapunov equations.

In Krylov subspace methods, an approximate solution to the Lyapunov equation is determined in the form $X \approx V P V^T$. First, we need to determine the columns of $V_m \in \mathbb{R}^{n \times l}$, which span an orthonormal basis for the mp -dimensional Krylov subspace defined by

$$K_m(A, B) := \text{span}(B, AB, A^2B, \dots, A^{m-1}B).$$

Using Algorithm 1, one can compute an orthonormal basis $V_m = [u_1, u_2, \dots, u_m]$ from the Krylov subspace K_m .

The main problem is to ensure that each of the reduced order equation

$$H_m Y_m + Y_m H_m^T + V_m^T B B^T V_m = 0$$

has a unique solution Y_m .

If H is real matrix, then Lyapunov equation has a unique solution iff $\lambda_i + \lambda_j \neq 0$ for every pair of eigenvalues λ_i and λ_j for H_m .

We are aware of some distinct extensions. The Galerkin type method of Jbilou and Riquet [18] relied on their global Arnoldi process. This method claims the solution of certain reduced order equations, which may or may not have unique solutions. If A is sparse, then the most expensive part of the method is to compute the orthogonal column of V_m by modified Gram-Schmidt process.

The convergence of the Arnoldi-Lyapunov method has been studied in [44]. The residual corresponding to Y_m is given by

$$R_m = A V_m Y_m V_m^T + V_m Y_m V_m^T A^T + B B^T. \quad (1.42)$$

This method computes Y_m such that it is satisfied the Galerkin condition

$$V_m^T R_m V_m = 0.$$

Theorem 1.26. *Let the Arnoldi process have been taken to m steps then an approximation $X_m = V_m Y_m V_m^T$ satisfies the Galerkin condition $V_m^T R_m V_m = 0$ if and only if Y_m is a solution of*

$$H_m Y_m + Y_m H_m^T + V_m^T B B^T V_m = 0.$$

In [19], Simoncini has developed a variant of the Arnoldi method which requires the solution of linear systems with coefficient matrix A . Again authors in [45], use a two pass version of the Lanczos algorithm to reduce memory consumption. The convergence of the Arnoldi method has been studied in [44].

1.6 Model Examples

In this section, we introduce some real problems of standard and descriptor LTI system. But we consider only the descriptor system for our numerical tests.

1.6.1 Earth Atmospheric Model

This is a model of an atmospheric storm track. In order to simulate the lack of coherence of the cyclone waves around the Earth's atmosphere, linear damping at the storm track's entry and exit region is introduced. The perturbation variable is the perturbation geopotential height. The mean flow is taken to be in a periodic channel in the zonal x -direction, $0 < x < 12\pi$, the channel is taken to be bounded with walls in the meridional y -direction located at $y = \pm\frac{\pi}{2}$ and at the ground, $z = 0$, and the tropopause, $z = 1$. The mean velocity is varying only with height and it is $U(z) = 0.2 + z$. Zonal and meridional lengths are nondimensionalized by $L = 1000\text{km}$, vertical scales by $H = 10\text{km}$, velocity by $U_0 = 30\text{ms}^{-1}$ and time is nondimensionalized advectively, i.e. $T = \frac{L}{U_0}$, so that a time unit is about $9h$.

The model can be represented by stable LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), & x(0) &= x_0, & t > 0 \\ y(t) &= Cx(t), & t &\geq 0 \end{aligned} \tag{1.43}$$

where $A \in \mathbb{R}^{598 \times 598}$, $C = [1, 1, 1, 1, \dots, 1]_{1 \times 598}$ and $B = C^T$.

1.6.2 Transmission Line Model

A transmission line is a circuit model modeling the impedance of interconnect structures accounting for both the charge accumulation on the surface of conductors and the current traveling along conductors.

The LTI system is defined as

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), & x(0) &= x_0, & t > 0 \\ y(t) &= Cx(t), & t &\geq 0 \end{aligned} \tag{1.44}$$

where $E, A \in \mathbb{R}^{256 \times 256}$, $B \in \mathbb{R}^{256 \times 2}$ and $C \in \mathbb{R}^{2 \times 256}$.

1.6.3 CD Player Model

The CD player control task is to achieve track following, which basically amounts to pointing the laser spot to the track of pits on the CD that is rotating. The mechanism treated here, consists of a swing arm on which a lens is mounted by means of two horizontal leaf springs. The rotation of the arm in the horizontal plane enables reading of the spiral shaped disc-tracks, and the suspended lens is used to focus the spot on the disc. Due to the fact that the disc is not perfectly flat, and due to irregularities in the spiral of pits on the disc, the challenge is to find a low-cost controller that can make the servo-system faster and less sensitive to external shocks. This is a sparse model like as the system 1.43.

1.6.4 Power System Model

Power system simulation involves power system modeling and network simulation in order to analyze electrical power systems using design or real-time data. It can be used in a wide range of planning and operational situations such as Electric power generation (Nuclear, Conventional, Renewable), Commercial facilities, Utility transmission, Utility distribution, Railway power systems and Industrial power systems.

Key elements of power systems that are modeled include:

1. Load flow (power flow study),
2. Short circuit or fault analysis,
3. Protective device coordination, discrimination or selectivity,
4. Transient or dynamic stability,
5. Harmonic or power quality analysis and
6. Optimal power flow.

The power system model can be represented by a number of differential-algebraic equations. The Brazilian Interconnected Power System (BIPS) models, introduced

in [24] provide a number of differential-algebraic systems which can be represented by

$$\underbrace{\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}}_{\tilde{E}} \underbrace{\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix}}_{\dot{\tilde{x}}(t)} = \underbrace{\begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} x(t) \\ z(t) \end{bmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{\tilde{B}} u(t), \quad (1.45)$$

$$y(t) = \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{\tilde{C}} \begin{bmatrix} x(t) \\ z(t) \end{bmatrix} + D_a u(t).$$

where A_4 is nonsingular and other sub matrices are highly sparse with appropriate dimensions. Since E is singular, the formulation is not as straightforward. Such systems are known as index-1 descriptor system.

Eliminating the algebraic variables from (1.45), one can reduce the system into standard state space form as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + D_a u(t), \end{aligned} \quad (1.46)$$

where $A := A_1 - A_2 A_4^{-1} A_3 \in \mathbb{R}^{n_d \times n_d}$, $B := B_1 - A_2 A_4^{-1} B_2 \in \mathbb{R}^{n_d \times p}$, $C := C_1 - C_2 A_4^{-1} A_3 \in \mathbb{R}^{n_d \times p}$ and $D_a := D_a - C_2 A_4^{-1} B_2 \in \mathbb{R}^{n_d \times p}$.

We refer *bips*² models for motivation of power systems.

²Available at <http://sites.google.com/site/rommes/software>

Chapter 2

Solution of Lyapunov Equation by Rational Krylov Subspace method

This chapter introduces the most integral part of this thesis for computing the low-rank solution of large-scale Lyapunov equations. During the last decades, several iterative methods have been developed to solve the large scale Lyapunov equation, e.g., LRCF-ADI iterations, cyclic low-rank Smith methods, projection methods and sign function methods. A brief discussion of some methods are given in section 1.5. Although, most of the methods can be applied for large scale sparse dynamical systems LRCF-ADI iteration [12] and Krylov subspace based projection method [19] are attractive. In this thesis, we concentrate our attention on RKSM [25] for generalized system which is well established in [27] for standard system. This technique is attractive since it is assumed to be cheap in computation and need not be required the stability of the system.

In first section, a short summary of the origin of RKSM is given for generalized system. Then RKSM is extended for index-1 descriptor system in Section 2.1.2. Another two major contributions are discussed in the next two sections. Section 2.2 discusses shift parameter selection to build the Krylov subspace and suitable termination criteria of the algorithm is given in Section 2.3. Finally, some numerical results are performed for the proposed methods.

2.1 Rational Krylov Subspace Method

This section presents the major contribution of this thesis to develop the RKSM. Among several projection methods, RKSM has some advantage in simplicity. Subsection 1.5.5 gives a brief overview of Krylov subspace method. In first subsection, the solution of the Lyapunov equation is discussed for generalized state space system. And next we formulate the index-1 descriptor system to generalized system so that the Lyapunov equation can be solved efficiently.

2.1.1 Generalized System

Consider a LTI continuous-time system of the form

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), & x(t_0) &= x_0, & t &\geq t_0 \\ y(t) &= Cx(t) + Du(t), \end{aligned} \quad (2.1)$$

where $E, A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, $C \in \mathbb{R}^{m \times n}$, $D \in \mathbb{R}^{m \times p}$. The vector valued functions $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^p$, $y(t) \in \mathbb{R}^m$ are prescribed to as the state, input and output of the system respectively, together with an initial condition $x(t_0) = x_0$. Here, n is called the order of the system and is assumed to be very large. If $E = I$, then (2.1) is known as standard state space system. Otherwise, (2.1) is called generalized or descriptor system, or differential-algebraic system.

In Section 1.3 we discussed some theorem which are shown that the system (2.1) is asymptotically stable if all eigen values of (A, E) lie in \mathbb{C}^- , while the controllability Gramian $P \in \mathbb{R}^{n_g \times n_g}$ and the observability Gramian $Q \in \mathbb{R}^{n_g \times n_g}$ are unique and symmetric positive definite. But, it can be shown that these two gramians are the solution of two continuous-time algebraic Lyapunov equations (CALE)

$$APE^T + EPA^T + BB^T = 0, \quad (2.2)$$

$$\text{and } A^TQE + E^TQA + C^TC = 0, \quad (2.3)$$

where the matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, $C \in \mathbb{R}^{m \times n}$, ($p \ll n$).

Consider in all cases, $\lambda_i(A) + \bar{\lambda}_j(A) \neq 0$, for all $i, j = 1, 2, \dots, n$, which ensure that the solution P of the equation (2.2) exists and is unique. Note that, controllability Lyapunov equation (2.2) and observability Lyapunov equation (2.3) are dual of

each other. Therefore, the solution of the controllability Lyapunov equations is the main stream entirely in this chapter.

For large n , it is impossible to store the original solution P of the controllability Lyapunov equation (2.2). Thus a low rank approximate solution is investigated such that

$$P \approx V\tilde{P}_aV^*, \quad V \in \mathbb{R}^{n \times r}, \quad \tilde{P}_a \in \mathbb{R}^{r \times r}.$$

RKSM computes the low-rank factor R to the approximate solution P_a of the Lyapunov equation (2.2) so that

$$P \approx P_a = RR^*.$$

The method is carried out by projecting the system onto the lower-dimensional Rational Krylov Subspace which can be generated iteratively.

Consider the m dimensional Rational Krylov Subspace defined as

$$K_m := \prod_{i=1}^m (A - \mu_i E)^{-1} B,$$

for a set of given shift parameters $\mu_i \in \mathbb{C}; i = 1, 2, \dots, m$.

We construct a projector V in such way that

$$\text{Rang}(V) = \text{span} \left(\prod_{i=1}^m (A - \mu_i E)^{-1} B \right).$$

Then the Galerkin condition is satisfied as

$$V^*(APE^T + EPA^T + BB^T)V = 0$$

$$(V^*AV)(V^*PV)(V^*EV)^T + (V^*EV)(V^*PV)(V^*AV)^T + V^*B(V^*B)^T = 0.$$

Consider $\tilde{P} = V^*PV$, $\tilde{E} = V^*EV$, $\tilde{A} = V^*AV$ and $\tilde{B} = V^*B$. Then the projected Lyapunov equation (??) can be written as

$$\tilde{A}\tilde{P}\tilde{E}^T + \tilde{E}\tilde{P}\tilde{A}^T = -\tilde{B}\tilde{B}^T. \quad (2.4)$$

Algorithm 3: Proposed method for generalized Lyapunov equation	
Input : E, A, B, i_{max} (number of iterations), μ_1 (initial shift).	
Output: R such that $P \approx RR^*$.	
1	Compute $v = (A - \mu_1 E)^{-1} B, V_1 = \frac{v}{\ v\ }$;
2	while (<i>not converged</i>) or $m \leq i_{max}$ do
3	Find $v = (A - \mu_{m+1} E)^{-1} V_m$ and compute adaptive shifts if store is empty as [29];
4	Orthogonalize v against V_m to obtain $v_{m+1}, V_{m+1} = [V_m, v_{m+1}]$;
5	Solve the small Lyapunov equation
	$A_{m+1} P E_{m+1}^T + E_{m+1} P A_{m+1}^T = -B_{m+1} B_{m+1}^*,$
	for P where $A_{m+1} = V_{m+1}^* A V_{m+1}, E_{m+1} = V_{m+1}^* E V_{m+1}$ and $B_{m+1} = V_{m+1}^* B$;
6	Compute the norm of the residual as defined in (2.23)
7	Compute eigen value decomposition $Y = T \Lambda T^* = [T_1 \ T_2] \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{bmatrix} \begin{bmatrix} T_1^* \\ T_2^* \end{bmatrix}$;
8	Truncate Λ_2 if the eigenvalues are sufficiently small and construct $R = V_{m+1} T_1 \Lambda_1^{\frac{1}{2}}$

The equation (2.4) is a small Lyapunov equation and can be solved by an existing methods, e.g., Bartels-Stewart method.

The solution \tilde{P} is symmetric and positive definite, so it can be factorized as

$$\tilde{P} = SS^*.$$

Using back substitution, the original solution can be computed as

$$P = V \tilde{P} V^*.$$

Applying eigenvalue decomposition to the matrix \tilde{P} and truncating the negligible eigenvalues. This ensure that the computed low rank factor R has smallest possible number of columns. Then

$$\begin{aligned} P &= V \tilde{P} V^* \\ &= V S S^* V^* = (VS)(VS)^* = RR^*. \end{aligned}$$

Finally, we store

$$R = VS,$$

which is the required low rank solution of the Lyapunov equation. The whole procedure is summarized in Algorithm 3.

2.1.2 Formulation and Solution of Index-1 Descriptor System

Now, we formulate the index-1 descriptor system into generalized system and apply RKSM to find the low rank solution of the Lyapunov equation, arises from index-1 descriptor system. We also show how to overcome the complexity of index-1 descriptor system such that it can be solved efficiently.

In matrix vector form, a LTI continuous time (semi explicit) index-1 descriptor system can be represented as

$$\underbrace{\begin{bmatrix} E_1 & E_2 \\ 0 & 0 \end{bmatrix}}_{\tilde{E}} \underbrace{\begin{bmatrix} \dot{x}_a(t) \\ \dot{x}_b(t) \end{bmatrix}}_{\dot{\tilde{x}}(t)} = \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{\tilde{B}} u(t), \quad (2.5)$$

and $y(t) = \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{\tilde{C}} \begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix} + D_a u(t).$

This equations are equivalent to the following differential-algebraic equations

$$E_1 \dot{x}_a(t) + E_2 \dot{x}_b(t) = J_1 x_a(t) + J_2 x_b(t) + B_1 u(t), \quad (2.6a)$$

$$0 = J_3 x_a(t) + J_4 x_b(t) + B_2 u(t), \quad (2.6b)$$

$$\text{and } y(t) = C_1 x_a(t) + C_2 x_b(t) + D_a u(t). \quad (2.6c)$$

Here, $x_a(t) \in \mathbb{R}^{n_a}$ and $x_b(t) \in \mathbb{R}^{n_b}$ are state vectors. The vector $x_b(t)$ is known as algebraic variables. The sub-matrices $E_1, E_2, J_1, J_2, J_3, J_4, B_1, B_2, C_1, C_2$ and D_a are sparse in appropriate dimensions.

Since J_4 is nonsingular block matrix, the systems (2.5) is claimed as index-1 descriptor system. Again in [46] authors called it semi explicit descriptor system of index-1 since the block matrix E_2 of E is non zero.

From the algebraic equation (2.6b), we obtain

$$x_b(t) = -J_4^{-1} J_3 x_a(t) - J_4^{-1} B_2 u(t). \quad (2.7)$$

Substituting the value of $x_b(t)$ in (2.6a) and (2.6c), respectively, we obtain

$$\begin{aligned} (E_1 - E_2 J_4^{-1} J_3) \dot{x}_a(t) &= (J_1 - J_2 J_4^{-1} J_3) x_a(t) + (B_1 - J_2 J_4^{-1} B_2) u(t) \\ &\quad + E_2 J_4^{-1} B_2 \dot{u}(t) \\ &= (J_1 - J_2 J_4^{-1} J_3) x_a(t) + [B_1 - J_2 J_4^{-1} B_2, E_2 J_4^{-1} B_2] \\ &\quad [u^T(t), \dot{u}^T(t)]^T, \end{aligned}$$

and

$$\begin{aligned} y(t) &= (C_1 - C_2 J_4^{-1} J_3) x_a(t) - C_2 J_4^{-1} B_2 u(t) + D_a u(t) \\ &= (C_1 - C_2 J_4^{-1} J_3) x_a(t) + [D_a - C_2 J_4^{-1} B_2, 0] [u^T(t), \dot{u}^T(t)]^T \end{aligned}$$

Define the following relations:

$$\begin{aligned} E &:= E_1 - E_2 J_4^{-1} J_3, & A &:= J_1 - J_2 J_4^{-1} J_3, & B &:= [B_1 - J_2 J_4^{-1} B_2, E_2 J_4^{-1} B_2], \\ C &:= C_1 - C_2 J_4^{-1} J_3, & D &:= [D_a - C_2 J_4^{-1} B_2, 0], & \bar{u}(t) &= [u^T(t), \dot{u}^T(t)]^T. \end{aligned}$$

Then the converted generalized systems can be written as

$$\begin{aligned} E \dot{x}_a(t) &= A x_a(t) + B \bar{u}(t)(t), \\ y(t) &= C x_a(t) + D \bar{u}(t)(t). \end{aligned} \tag{2.8}$$

Therefore, the descriptor system (2.5) can be converted to the generalized system (2.1) by the above relations. Note that the system matrices in (2.8) are formed in dense formulation.

The following proposition shows that the converted dynamical systems (2.8) and original system (2.5) are equivalent in a sense that their transfer functions and the finite spectrums are same [46].

The mapping from inputs to outputs are described by the transfer function matrix, so it is essential for the input-output relations of any system. In frequency domain, the transfer function matrix $G(\mu) \in \mathbb{C}^{m \times p}$ of a system is defined by

$$G(\mu) = C(\mu E - A)^{-1} B + D, \quad \mu \in \mathbb{C}. \tag{2.9}$$

Proposition 2.1. *Let $\tilde{G}(\mu)$ be a transfer function of the (semi explicit) index-1 descriptor system (2.5) and let $G(\mu) = C(\mu E - A)^{-1} B + D$ be obtained for the system (2.8). Then $\tilde{G}(\mu)$ and $G(\mu)$ are same.*

Proof. The transfer function of the semi explicit descriptor system (2.5) can be written as

$$\tilde{G}(\mu) = \tilde{C}(s\tilde{E} - \tilde{A})^{-1}\tilde{B} + D_a = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} + D_a \quad (2.10)$$

where

$$\begin{aligned} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} &= (s\tilde{E} - \tilde{A})^{-1}\tilde{B}, \\ \text{or, } \begin{bmatrix} \mu E_1 - J_1 & \mu E_2 - J_2 \\ -J_3 & -J_4 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} &= \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \end{aligned}$$

This implies that

$$(\mu E_1 - J_1)F_1 + (\mu E_2 - J_2)F_2 = B_1, \quad (2.11)$$

and

$$\begin{aligned} -J_3F_1 - J_4F_2 &= B_2 \\ \text{or, } -J_4F_2 &= (B_2 + J_3F_1) \\ \text{or, } F_2 &= -J_4^{-1}(B_2 + J_3F_1) \end{aligned}$$

Putting the value of F_2 in (2.11), we get

$$\begin{aligned} (\mu E_1 - J_1)F_1 + (\mu E_2 - J_2)(-J_4^{-1}(B_2 + J_3F_1)) &= B_1 \\ \text{or, } [(\mu E_1 - J_1) - (\mu E_2 - J_2)J_4^{-1}J_3]F_1 &= B_1 + (\mu E_2 - J_2)J_4^{-1}B_2 \end{aligned}$$

$$\begin{aligned} \text{or, } F_1 &= [\mu(E_1 - E_2J_4^{-1}J_3) - (J_1 - J_2J_4^{-1}J_3)]^{-1}[B_1 - J_2)J_4^{-1}B_2 + \mu E_2J_4^{-1}B_2] \\ &= (\mu E - A)^{-1}B. \end{aligned}$$

Then equation (2.10) gives

$$\begin{aligned} \tilde{G}(\mu) &= C_1F_1 + C_2F_2 + D_a \\ &= C_1F_1 - C_2J_4^{-1}(B_2 + J_3F_1) + D_a \\ &= (C_1 - C_2J_4^{-1}J_3)F_1 + D_a - C_2J_4^{-1}B_2 \\ &= C(\mu E - A)^{-1}B + D = G(\mu). \end{aligned}$$

□

The main computational cost in RKSM is the solution of linear systems with matrices of the form $(A - \mu E)^{-1}V$ (i.e., steps 1 and 3 in Algorithm 3).

The matrix A in system (2.8) is not sparse or a poor sparsity pattern so it becomes a most expensive operations, i.e., the converted system claimed large and dense system matrices A and E .

At each iteration we have to solve a shifted linear system like

$$(A - \mu_i E)v_i = V_{i-1} \quad (2.12)$$

$$\text{or, } ((J_1 - J_2 J_4^{-1} J_3) - \mu_i E_1)v_i = V_{i-1}. \quad (2.13)$$

To overcome the complexity, we can use the sparsity pattern of A instead of its dense form so that the operation can be solved efficiently by suitable direct or iterative solvers [4, 47].

Therefore, we solve the linear system as

$$\begin{bmatrix} J_1 - \mu_i E_1 & J_2 - \mu_i E_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} v_i \\ \star \end{bmatrix} = \begin{bmatrix} V_{i-1} \\ 0 \end{bmatrix} \quad (2.14)$$

Similarly, the observability Lyapunov equation (2.3) can be solved by the above procedure. In that case, input (E, A, B) is changed to (E^T, A^T, C^T) .

For observability gramian, we can be handled the system $(A^T - \mu E^T)w = C^T$ by the following linear system

$$\begin{bmatrix} J_1^T - \mu E_1^T & J_3^T \\ J_2^T - \mu E_2^T & J_4^T \end{bmatrix} \begin{bmatrix} w_i \\ \star \end{bmatrix} = \begin{bmatrix} V_{i-1} \\ 0 \end{bmatrix} \quad (2.15)$$

Special case I:

When $E_2 = 0$ in system (2.5), the index-1 descriptor system can be represented by

$$\underbrace{\begin{bmatrix} E_1 & 0 \\ 0 & 0 \end{bmatrix}}_{\tilde{E}} \underbrace{\begin{bmatrix} \dot{x}_a(t) \\ \dot{x}_b(t) \end{bmatrix}}_{\dot{\tilde{x}}(t)} = \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{\tilde{B}} u(t), \quad (2.16)$$

$$y(t) = \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{\tilde{C}} \begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix} + D_a u(t).$$

This is equivalent to the following differential-algebraic equations of system

$$E_1 \dot{x}_a(t) = J_1 x_a(t) + J_2 x_b(t) + B_1 u(t), \quad (2.17)$$

$$0 = J_3 x_a(t) + J_4 x_b(t) + B_2 u(t), \quad (2.18)$$

$$y(t) = C_1 x_a(t) + C_2 x_b(t) + D_a u(t). \quad (2.19)$$

In the similar way, the descriptor system (2.16) can be converted to the system (2.8) with the following relations

$$\begin{aligned} E &:= E_1, & A &:= J_1 - J_2 J_4^{-1} J_3, & B &:= B_1 - J_2 J_4^{-1} B_2, \\ C &:= C_1 - C_2 J_4^{-1} J_3, & D &:= D_a - C_2 J_4^{-1} B_2. \end{aligned}$$

To overcome the complexity, we solve the linear system

$$\begin{bmatrix} J_1 - \mu_i E_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} v_i \\ \star \end{bmatrix} = \begin{bmatrix} V_{i-1} \\ 0 \end{bmatrix} \quad (2.20)$$

For observability Lyapunov equation, we solve the linear system

$$\begin{bmatrix} J_1^T - \mu E_1^T & J_3^T \\ J_2^T & J_4^T \end{bmatrix} \begin{bmatrix} w_i \\ \star \end{bmatrix} = \begin{bmatrix} V_{i-1} \\ 0 \end{bmatrix} \quad (2.21)$$

Special case II:

When $J_2 = 0$ in (2.16), the system (2.16) becomes

$$\begin{aligned} \underbrace{\begin{bmatrix} E_1 & 0 \\ 0 & 0 \end{bmatrix}}_{\tilde{E}} \underbrace{\begin{bmatrix} \dot{x}_a(t) \\ \dot{x}_b(t) \end{bmatrix}}_{\dot{\tilde{x}}(t)} &= \underbrace{\begin{bmatrix} J_1 & 0 \\ J_3 & J_4 \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix}}_{\tilde{x}(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{\tilde{B}} u(t), \\ y(t) &= \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{\tilde{C}} \begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix} + D_a u(t). \end{aligned} \quad (2.22)$$

Then the new converted system can be generated by the relations:

$$E := E_1, \quad A := J_1, \quad B := B_1, \quad C := C_1 - C_2 J_4^{-1} J_3, \quad D := D_a - C_2 J_4^{-1} B_2.$$

For controllability gramian, the linear system can be solved by $(J_1 - \mu E_1)^{-1} v = B_1$.

For observability gramian, we can use the linear system $(J_1^T - \mu E_1^T)^{-1} v = C^T$.

2.2 Shift Parameter Selection

It is mentioned that at each iteration of RKSM, a linear system has to be solved with shifted matrix. Proper shift computation is another important task to fast convergence of the RKSM. There are many way to compute the shift parameter in the research. In this section, we present a simple and efficient technique, so called adaptive shift selection approach.

In [29], the authors are shown that a set of optimal shift parameters μ_i can be computed by solving the so called min-max problem

$$\min_{\mu_1, \dots, \mu_j \in \mathbb{C}^-} \left(\max_{1 \leq l \leq n} \prod_{i=1}^J \frac{|z_i - \lambda|}{|z_i + \lambda|} \right), \quad \lambda \in \Lambda(A, E).$$

Recently, this technique is extended for the descriptor system. In this approach, an adaptive shift parameters are generated automatically by itself. Here, we apply the modified version of adaptive shift selection, available in [28]. The approximate solutions of Lyapunov equation in (2.2) converge to exact solutions if the projected system is asymptotically stable. Sometimes the projected eigenvalue cannot be in the negative real part. Then we convert in \mathbb{C}^- .

For a given orthogonal matrix $v \in \mathbb{R}^{n \times k}$, $k \ll n$, one can compute the k eigenvalues of the projected matrix pencil

$$(\lambda v^T E v - v^T A v), \quad \lambda \in \mathbb{C}.$$

The computed eigenvalues are used as shift parameters. There, the shifts are initialized by the eigenvalues of the pencil projected by the subspace. Whenever, the current set has been used, then choose the next set of shifts following by the current subspace.

If any eigenvalues will be in C^+ , we project them in negative real part of the complex plane. Note that in this case, the problem does not exist the infinite eigenvalues.

2.3 Stopping Criteria

One way to stop Algorithm 3 is to compute the changes in the approximation, i.e., when $\|P_a - \tilde{P}\| < \text{tolerance}$. A relative change can also be used by scaling $\|P\|$. However, the computation of the norm of approximate solution might be expensive in large dimensions. A similar stopping criterion is based on the relative change in the low-rank factors [19, 27] measured in the Frobenius norm.

The method can be stopped at the m -th iteration efficiently, if

$$\frac{\|\mathcal{R}(P_a)\|_F}{\|BB^T\|_F + \|A\|_F\|P\|_F} \leq \text{tol},$$

where $\|\cdot\|_F$ denotes Frobenius norm and the residual at m th iteration step is

$$\mathcal{R}(P_a) = A\tilde{P}_m E^T + E\tilde{P}_m A^T + BB^T \quad (2.23)$$

We focus on terminating Algorithm 3 based on the norm of the Lyapunov residual $\mathcal{R}(P_a)$, which is the most expensive to evaluate. The F -norms of $\|BB^T\|$ and $\|A\|$ are computed explicitly at the beginning of the process. Other hand, $\|P\|_F$ has to be computed in each iteration step.

For large scale Lyapunov equations, $\mathcal{R}(P_a)$ is difficult to compute, since $\mathcal{R}(P_a)$ is a large and dense matrix. Hence, the residual norm can be computed cheaply using the following observation [27, Proposition 4.1].

Theorem 2.2. *Let V_m be the orthogonal basis of the Rational Krylov Subspace K_m and $P = V_m Y V_m^*$ be the approximate solution of the Lyapunov equation. Then the residual R_m can be computed as*

$$\|\mathcal{R}_m\|_F = \|SJS^T\|_F, \quad J = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

where S is the upper triangular matrix in the QR factorization of

$$U = [v_{m+1}\mu_{m+1}, EV_m Y H_m^{-T} e_m h_{m+1,m} - (I - V_m V_m^T) A v_{m+1}],$$

where H_m is a block upper Hessenberg matrix and e_m be the matrix formed by the last p columns of the $mp \times mp$ identity matrix.

Proof. Applying Arnoldi algorithm, we can write

$$(A - \mu_{j+1}E)^{-1}v_j = V_{j+1}H_{1:j+1,j}, \quad j = 1, \dots, m.$$

We can obtain

$$AV_m = [V_m T_m + v_{m+1} h_{m+1,m} e_m^T D_m H_m^{-1}, -(I - V_m V_m^T) A v_{m+1} h_{m+1} e_m^T H_m^{-1}].$$

Therefore,

$$\begin{aligned} \mathcal{R} &= A\tilde{P}E^T + E\tilde{P}A^T + BB^T \\ &= AV_m P_m V_m^T E^T + EV_m P_m V_m^T A^T + BB^T \\ &= (V_m T_m + v_{m+1} h_{m+1,m} e_m^T D_m H_m^{-1} - (I - V_m V_m^T) A v_{m+1} h_{m+1} e_m^T H_m^{-1}) P_m V_m^T E^T \\ &\quad + EV_m P_m (V_m T_m + v_{m+1} h_{m+1,m} e_m^T D_m H_m^{-1} - (I - V_m V_m^T) A v_{m+1} h_{m+1} e_m^T H_m^{-1})^T \\ &\quad + BB^T \\ &= v_{m+1} h_{m+1,m} e_m^T D_m H_m^{-1} P_m V_m^T E^T - f h_{m+1,m} e_m^T H_m^{-1} P_m V_m^T E^T + \\ &\quad + EV_m P_m H_m^{-1} e_m h_{m+1,m}^T D_m v_{m+1}^T - EV_m P_m H_m^{-1} e_m h_{m+1,m}^T f^T + V_m T_m P_m V_m^T E^T \\ &\quad + EV_m P_m V_m T_m + BB^T \\ &= v_{m+1} \mu_{m+1} h_{m+1,m}^T e_m^T H_m^{-1} P_m^T V_m^T E^T - f h_{m+1,m}^T e_m^T H_m^{-1} P_m^T V_m^T E^T + \\ &\quad + EV_m P_m H_m^{-1} e_m h_{m+1,m}^T \mu_{m+1} v_{m+1}^T - EV_m P_m H_m^{-1} e_m h_{m+1,m}^T f^T \\ &= EV_m P_m H_m^{-1} e_m h_{m+1,m} \mu_{m+1} v_{m+1}^T + (v_{m+1} \mu_{m+1} - f) h_{m+1,m}^T e_m^T H_m^{-1} P_m^T V_m^T E^T \\ &\quad - EV_m P_m H_m^{-1} e_m h_{m+1,m}^T f^T \\ &= \begin{bmatrix} EV_m P_m H_m^{-1} e_m h_{m+1,m} & \mu_{m+1} v_{m+1} - f & EV_m P_m H_m^{-1} e_m h_{m+1,m} \end{bmatrix} \\ &\quad \begin{bmatrix} \mu_{m+1} v_{m+1}^T \\ h_{m+1,m}^T e_m^T H_m^{-1} P_m^T V_m^T E^T \\ -f^T \end{bmatrix} \\ &= \begin{bmatrix} \mu_{m+1} v_{m+1} - f & EV_m P_m H_m^{-1} e_m h_{m+1,m} & -f \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mu_{m+1} v_{m+1}^T \\ h_{m+1,m}^T e_m^T H_m^{-1} P_m^T V_m^T E^T \\ -f^T \end{bmatrix} \\ &= SJS^T. \end{aligned}$$

□

2.4 Numerical Results

To assess the performance of the proposed method, some results of numerical experiments are presented in this section. First consider an artificial data so that one can construct different sizes of models to test the proposed method. Although, we found some real power system data for second type descriptor system.

All of the following results were carried out using MATLAB (R2013b) on an Intel Core i5 @ 1.70 GHz clock, RAM 8GB with machine precision $\epsilon = 1.7 \times 10^{-16}$.

2.4.1 Artificial Data

A set of artificial data is considered for the index-1 descriptor system by the following ways:

```

2n=number of state differential variables;
m=number of algebraic variables;
inp=number of input;
outp=number of output;
N=spdiags(-5*ones(n,1),0,n,n)+spdiags(-1*ones(n,1),2,n,n)+spdiags(-1*ones(n,1),-2,n,n)
    +spdiags(2*ones(n,1),4,n,n)+spdiags(2*ones(n,1),-4,n,n);
L=spdiags(10*ones(m,1),0,m,m)+spdiags(ones(m,1),2,m,m)+spdiags(ones(m,1),-2,m,m)
    +spdiags(-2*ones(n,1),4,m,m)+spdiags(-2*ones(n,1),-4,m,m);
Q=sprand(n,m,0.001);
E1=speye(2*n);
E2=spdiags(-.5*ones(2*n,1),0,2*n,m); % for E2 is not zero
E2=spalloc(size(J2,1),size(J2,2),0); % for E2=0
E=[E1 E2;spalloc(size(J3,1),size(J3,2),0) spalloc(size(J4,1),size(J4,2),0)];
J1=[-2*speye(n) speye(n);-N -5*speye(n)];
J2=[spalloc(n,m,0);-Q];
J3=J2';
J4=-L;

```

By changing the value of n and m , one can constructed the different sizes of artificial models for index-1 descriptor system. Also the number of input and output can be fixed by replacing the values of ' inp ' and ' $outp$ ', and ' den ' denotes the number of nonzero entries.

Assume the formulation of E_2 , we can construct two different models as for case-1, $E_2 \neq 0$ and for case-2, $E_2 = 0$. For our numerical test, we consider $den = 0.001$, $inp = outp = 1$ (SISO). Also different types data can be constructed by replacing the block matrices of each system matrix.

Table. 2.1. Computational time for different dimensional systems by exact solver and RKSM (sparse and dense linear system)

Dimension	Exact solver(sec)	RKSM sparse(sec)	RKSM dense(sec)
1600	23	1.43	2.7
1800	34	1.61	3.08
2200	67	1.93	4.68
2600	117	2.3	6.47
3000	177	2.68	7.74
4000	398	3.47	8.72
6000	1501	7.13	19.8

Compute the controllability gramian factor for different systems, table 2.1 shows the computational time between exact solver (matlab: lyap) and proposed methods (consider 50 iterations). This shows that RKSM is more efficient than the exact solver.

Figure 2.1 represents the sparsity patterns of the system matrix E and matrix A , for 5000 dimensional artificial data. Figure 2.2a shows the convergence history for the controllability Lyapunov equation and figure 2.2b is for the observability Lyapunov equation of artificial data. One can notice that the convergence is very fast with the iteration steps 35, in both cases.

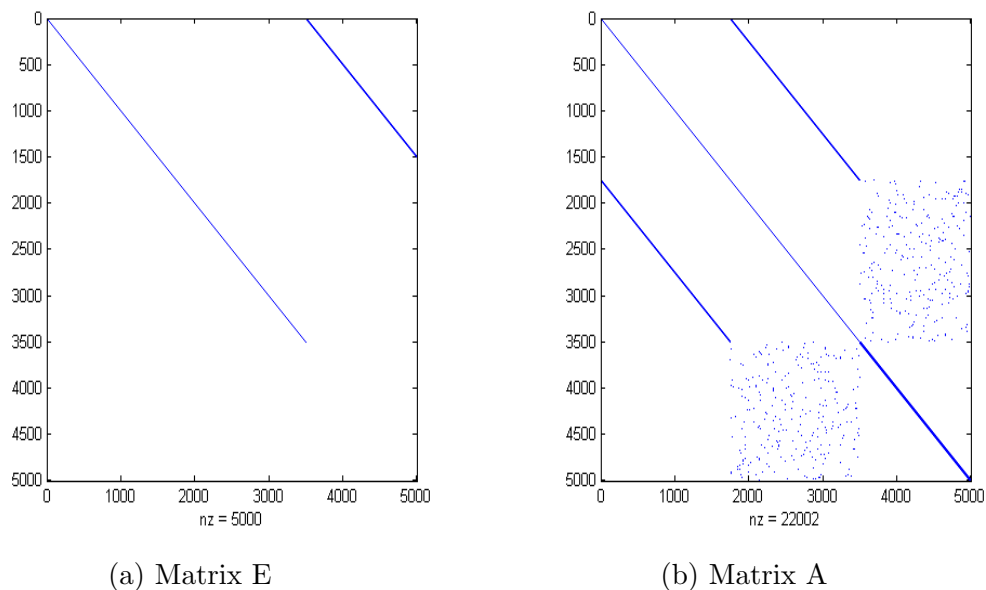


Figure. 2.1. Sparsity patterns of the artificial data

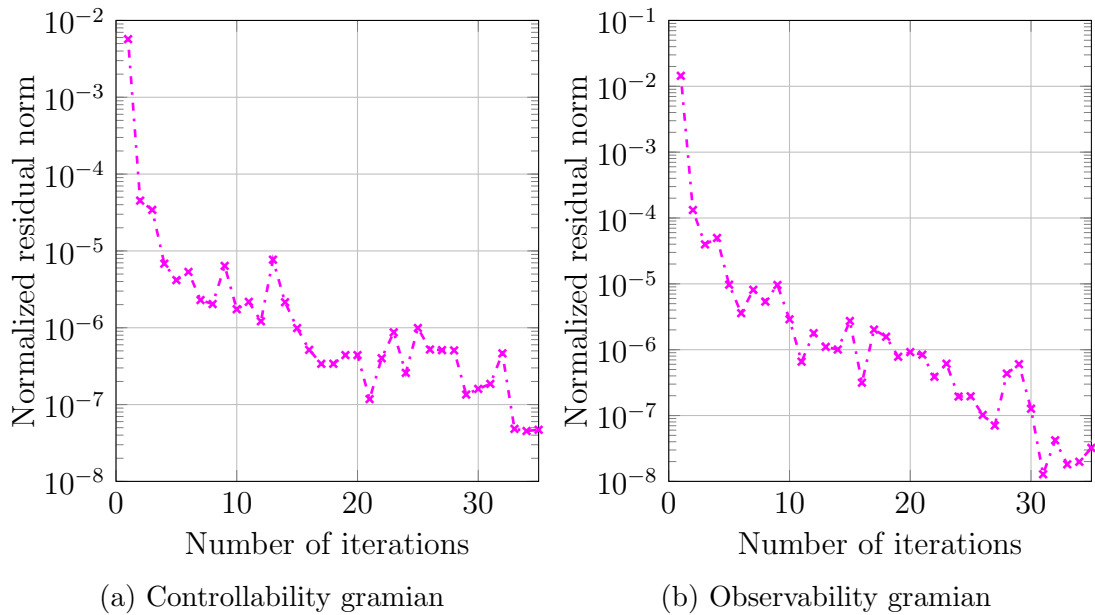


Figure. 2.2. Convergence histories of RKSM for artificial data.

Figure 2.3 shows the accuracy of the RKSM. In figure 2.3b, it can easily be shown that the largest singular values of the controllability gramian match accurate for both exact solver and RKSM, respectively. The same histories for observability gramian are showed in figure 2.3b for exact solver and RKSM, respectively.

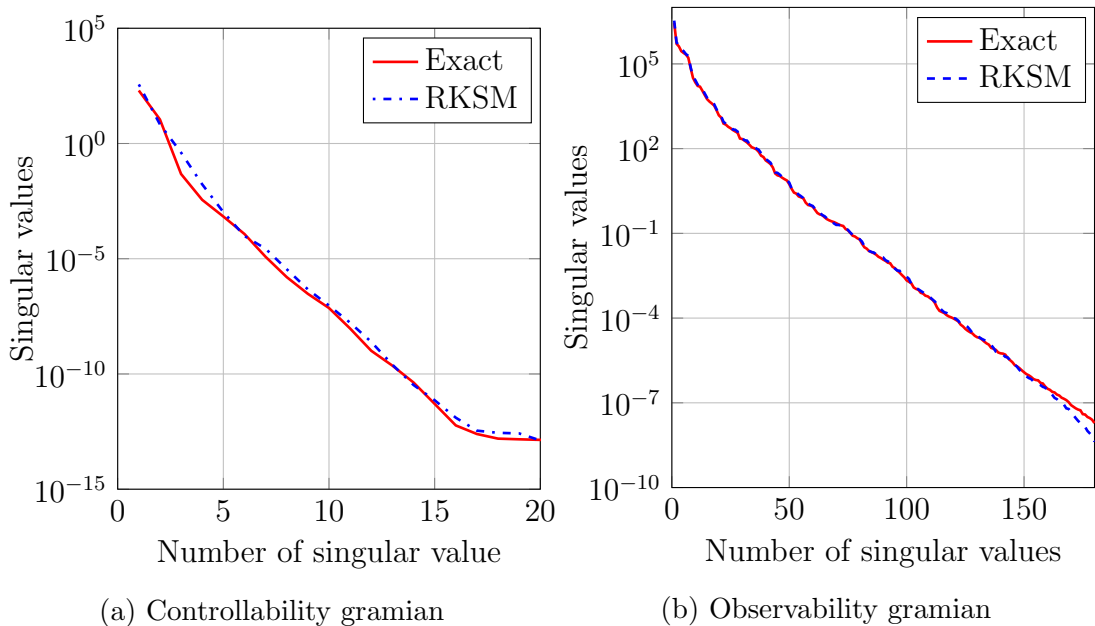


Figure. 2.3. Singular values of the gramians computed by exact solver and RKSM for artificial data.

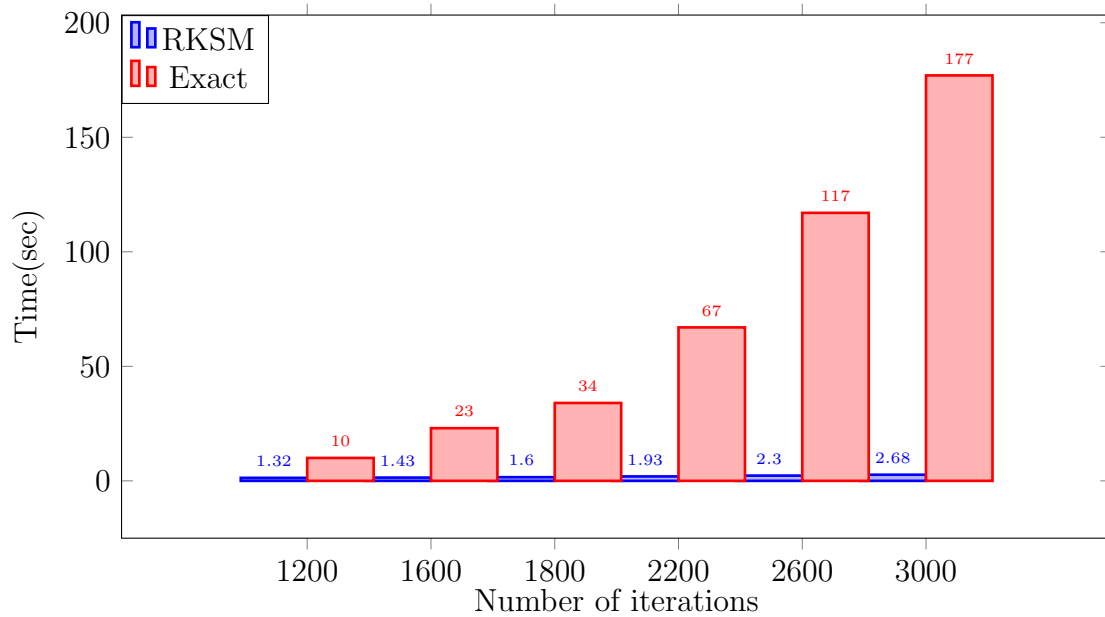


Figure. 2.4. Computational time of controllability gramian of artificial model for different system by RKSM and exact solver

Bar chart of figure 2.4 shows the computational time of controllability gramian for different systems by RKSM and exact solver.

2.4.2 Power System Data

In Subsection 1.6.4, some real data are discussed, the power systems are generated from BIPS [24]. Three models are considered for our numerical test which are all index-1 descriptor system (case-2).

Table 2.2 shows a summary of some power system data. The number of differential and algebraic variables, and large eigenvalue of the pair $(-A, E)$ for different sizes of the models are shown in table 2.3.

Table. 2.2. Dimensions of system matrices of BIPS models.

Model	A	B	C	D
Mod-1	7 135	(7135, 4)	(4, 7135)	(4, 4)
Mod-2	9 735	(9735, 4)	(4, 9735)	(4, 4)
Mod-3	21 128	(21128, 4)	(4, 21128)	(4, 4)

Table. 2.3. Number of differential & algebraic variables and largest eigenvalue of $(-A, E)$ for different models.

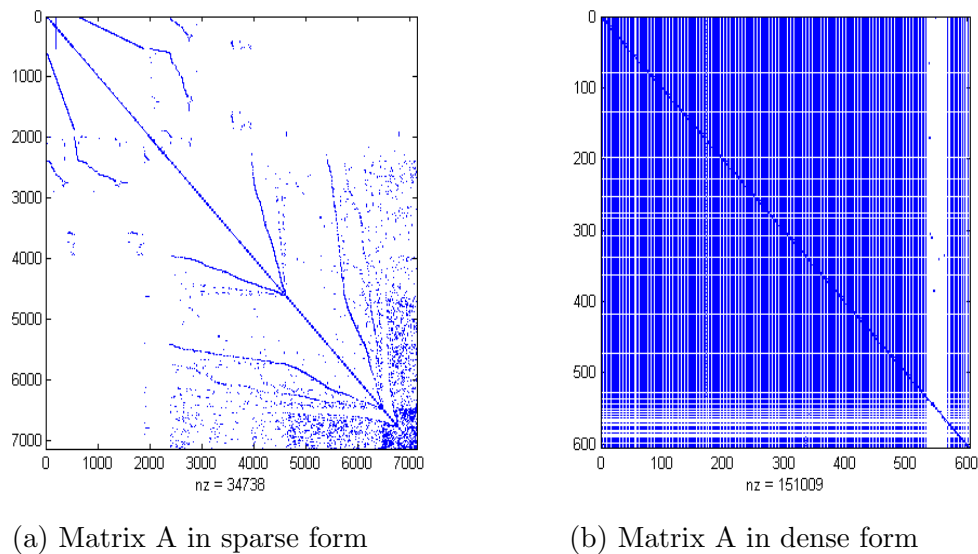
Model	Dim. of differential	Dim. of algebraic	Inputs/outputs
Mod-1(small)	606	6 529	4/4
Mod-2(medium)	1 142	8 593	4/4
Mod-3(large)	3 078	18 050	4/4

For Mod-1, the sparsity patterns of the matrix A and its dense form are shown in figure 2.5. After dense formulation the model reduces low sparsity pattern.

We compute the controllability gramian and observability gramian by exact solver and RKSM, respectively. The relative error for controllability gramian is $\frac{\|X_c - P\|_2}{\|X_c\|_2^2} = 1.95 \times 10^{-8}$ and for observability gramian is $\frac{\|X_o - Q\|_2}{\|X_o\|_2^2} = 9.6 \times 10^{-9}$, where X_c and P are the controllability gramian for the exact solver and RKSM, respectively and X_o and Q are same history for observability gramian.

Figure 2.6a shows the convergence history for the controllability Lyapunov equation and figure 2.6b is for the observability Lyapunov equation, for Mod-1. We observe that the convergence is very fast with the 60 iteration steps for both cases.

Figure 2.7 shows the accuracy of the RKSM method for Mod-1. In figure 2.7a, it can easily be shown that the largest singular values of the controllability gramians

Figure. 2.5. Sparsity patterns of the matrix A and its dense form for Mod-1

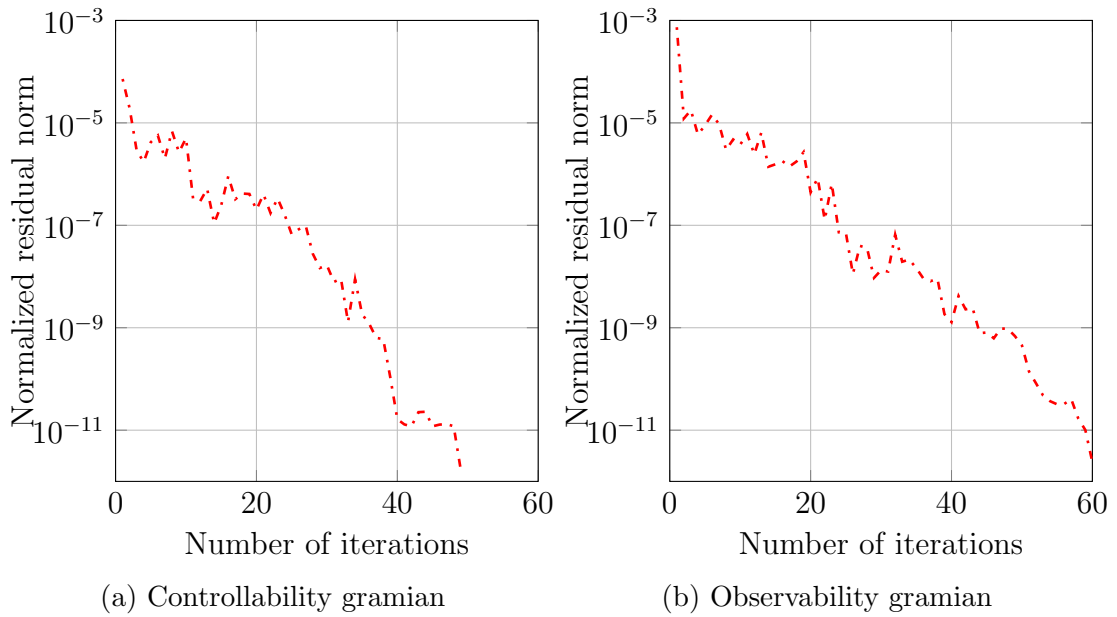


Figure. 2.6. Convergence histories of RKSM for Mod-1.

by exact solvers and RKSM, respectively, match accurately. The same histories for observability gramians are depicted in figure 2.7b.

For Mod-2, the convergence history for controllability gramian and observability gramians are shown in figure 2.8. One can notice that very fast convergence is obtained with the iteration steps 80 and 65, respectively in both cases.

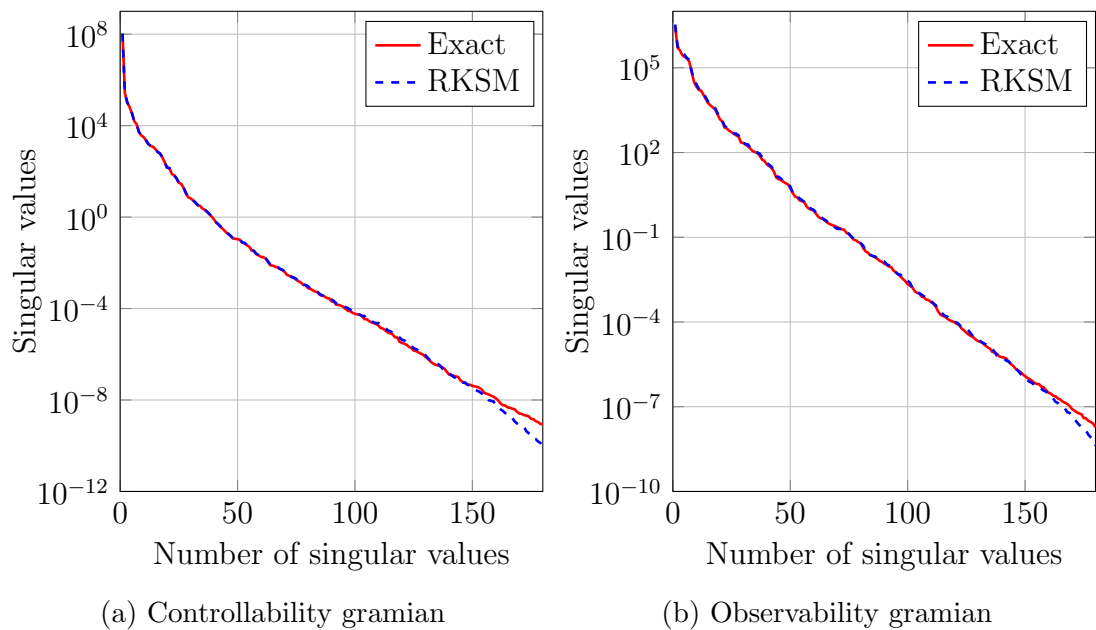


Figure. 2.7. Singular values of the gramians computed by exact solver and RKSM for Mod-1.

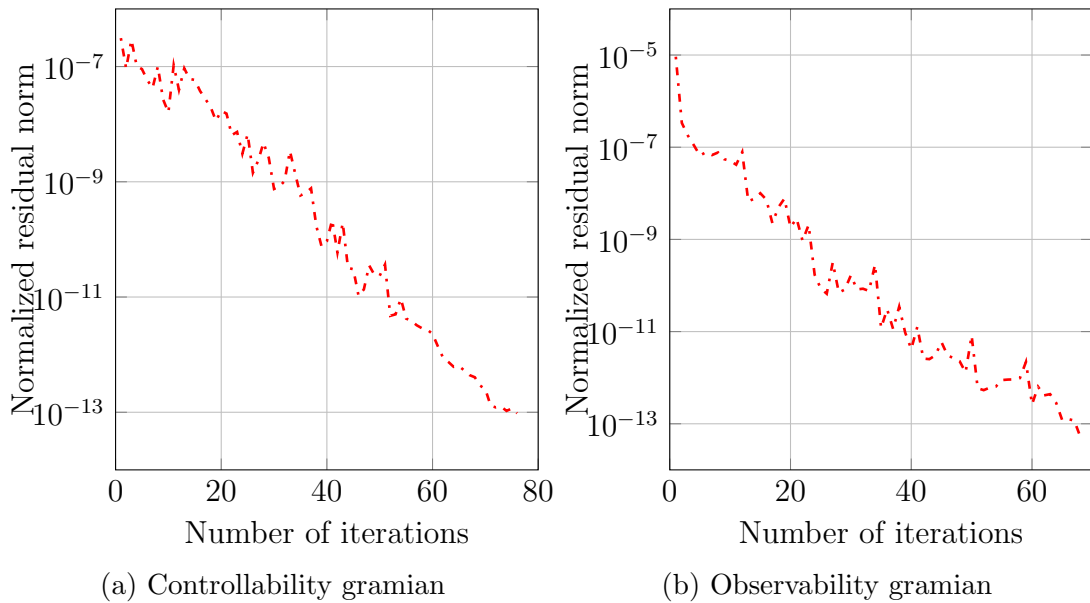


Figure. 2.8. Convergence histories of RKSM for Mod-2.

Figure 2.9 shows the accuracy of the RKSM method for Mod-2. The largest singular values of the controllability gramian are depicted in figure 2.9a by exact solver and RKSM, respectively. The same histories for observability gramian are also shown in figure 2.9b. Both figures show that the 200 singular values match accurately in both cases.

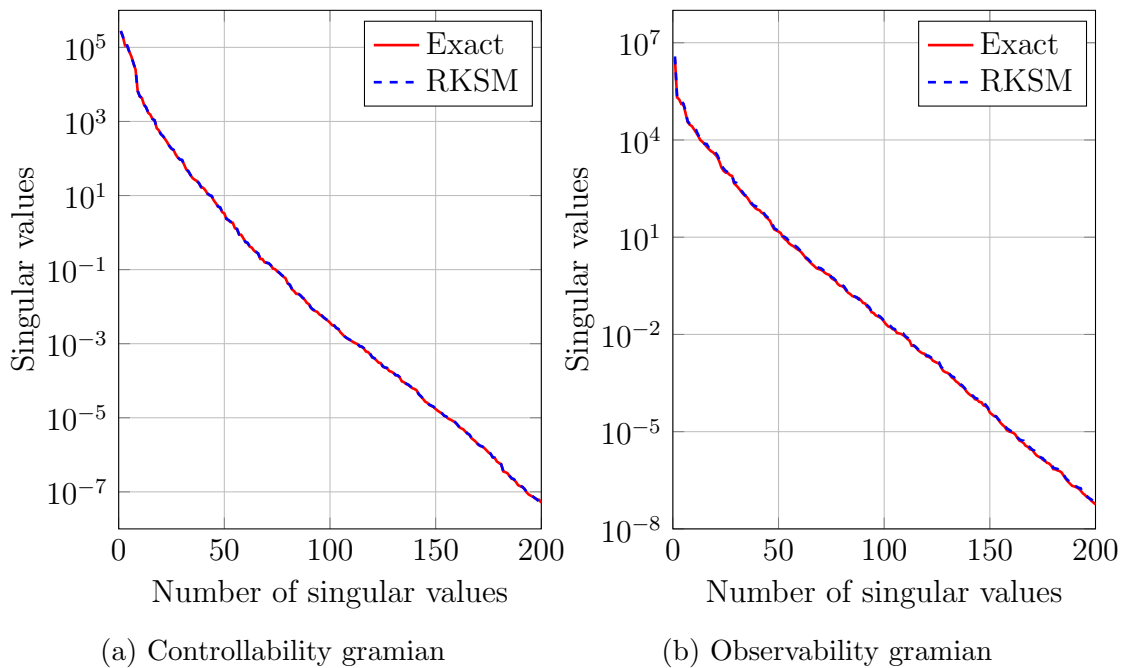
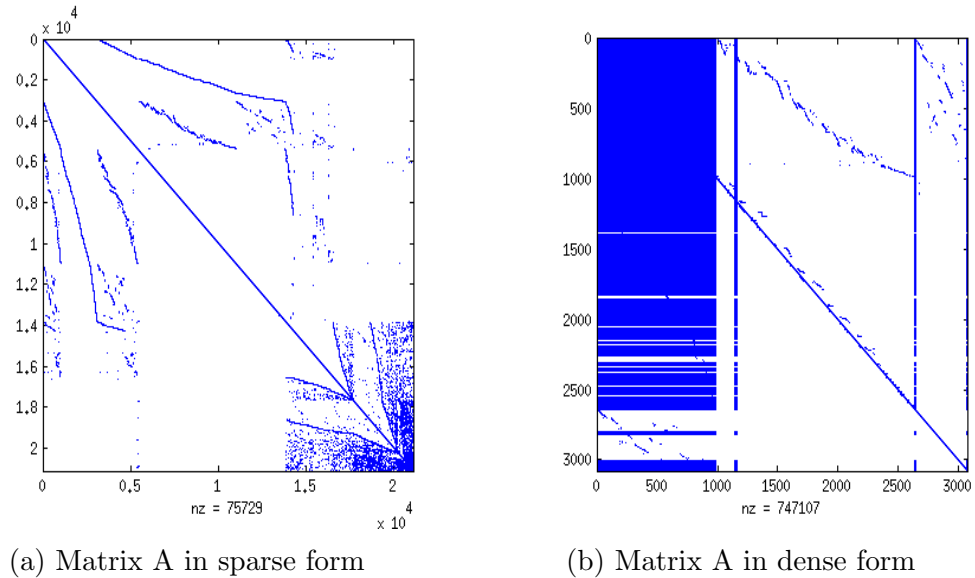


Figure. 2.9. Singular values of the gramians computed by exact solver and RKSM for Mod-2.

Figure. 2.10. Sparsity patterns of the matrix A and its dense form for Mod-3

For Mod-3, the sparsity patterns of the matrix A and its dense form are shown in figure 2.10. The dense formulation yields low sparsity for this large model.

We compute the controllability gramian and observability gramian by exact solver and RKSM respectively. Figure 2.11 shows the convergence histories for both the gramians and the convergence is very fast. After 60 iterations it goes to 10^{-11} which shows that RKSM gives a good approximate solution.

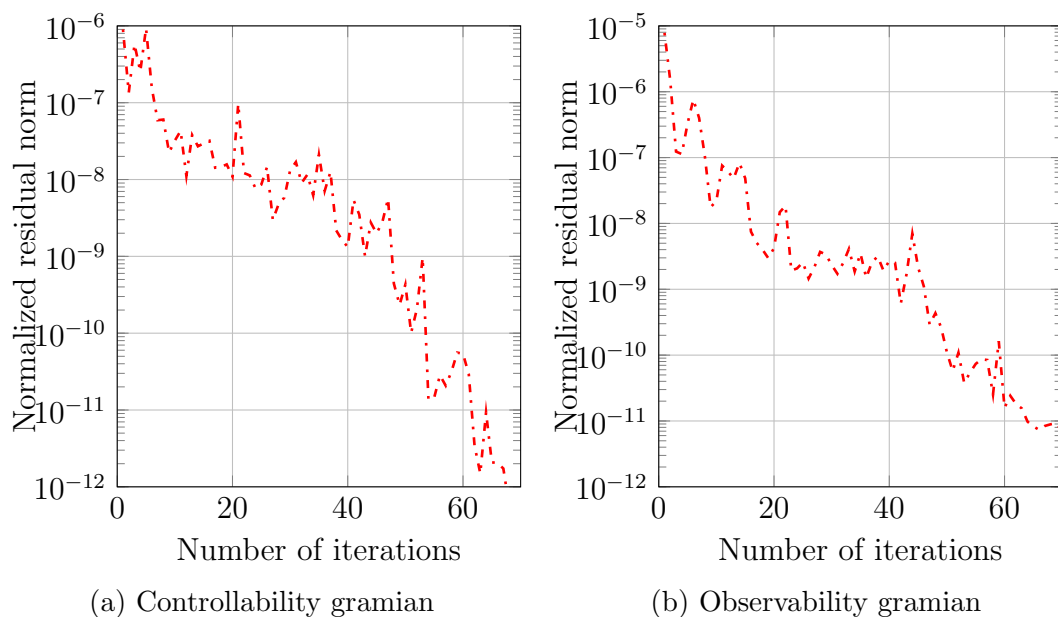


Figure. 2.11. Convergence histories of RKSM for Mod-3.

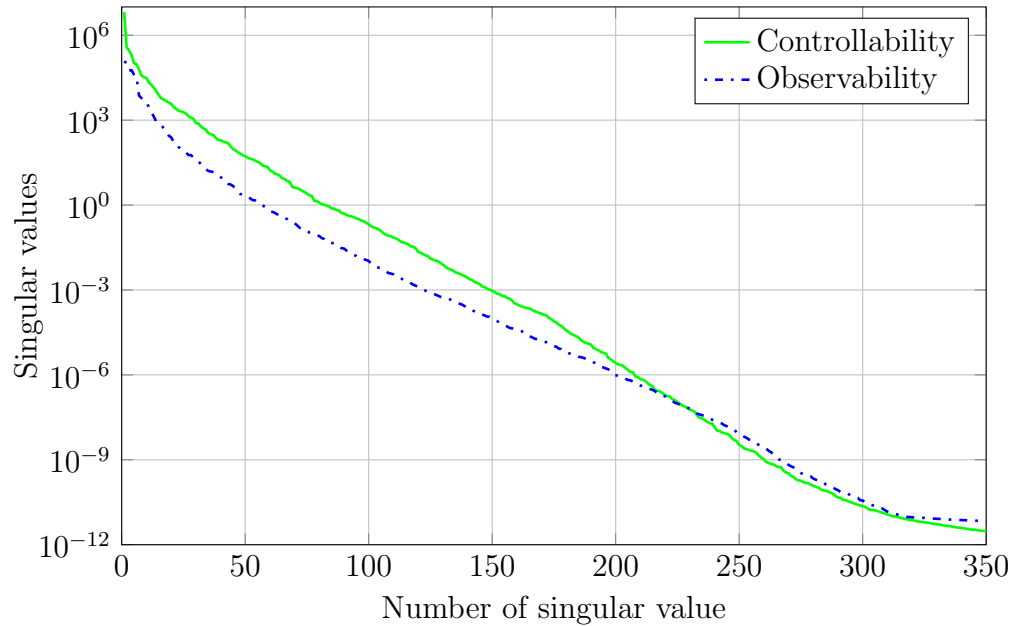


Figure. 2.12. Singular values of both gramians computed by RKSM for Mod-3.

The singular values of the controllability and observability gramians depict in same figure 2.12. It is very expensive to compute the exact gramian for Mod-3. Computational time comparison for sparse and dense linear system are depicted in the bar chart of figure 2.13.

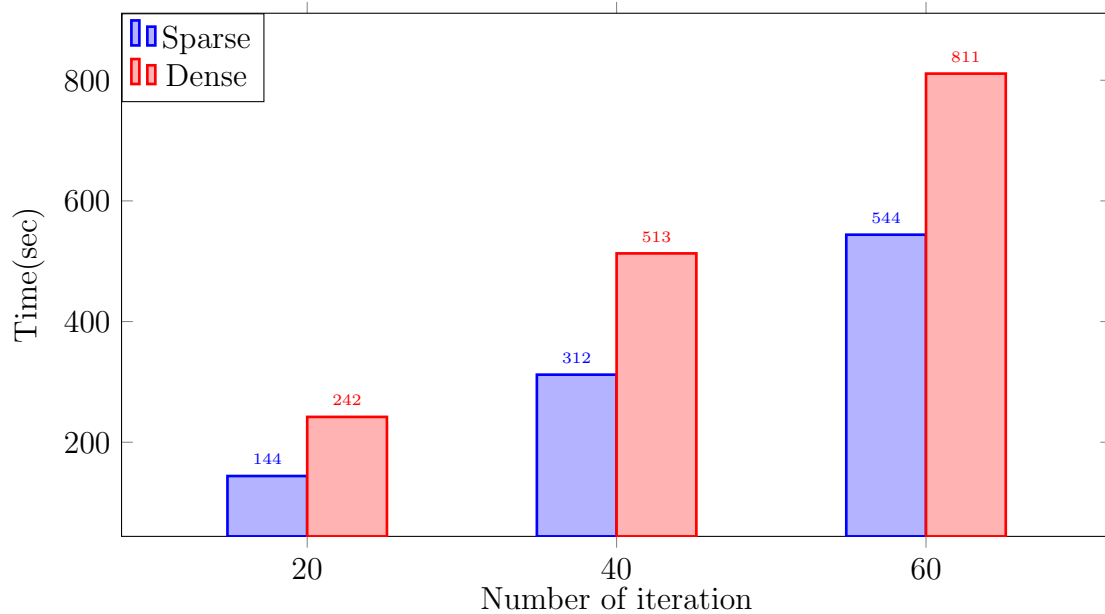


Figure. 2.13. Computational time for dense system and sparse system at different iteration by RKSM ($n = 21, 128$)

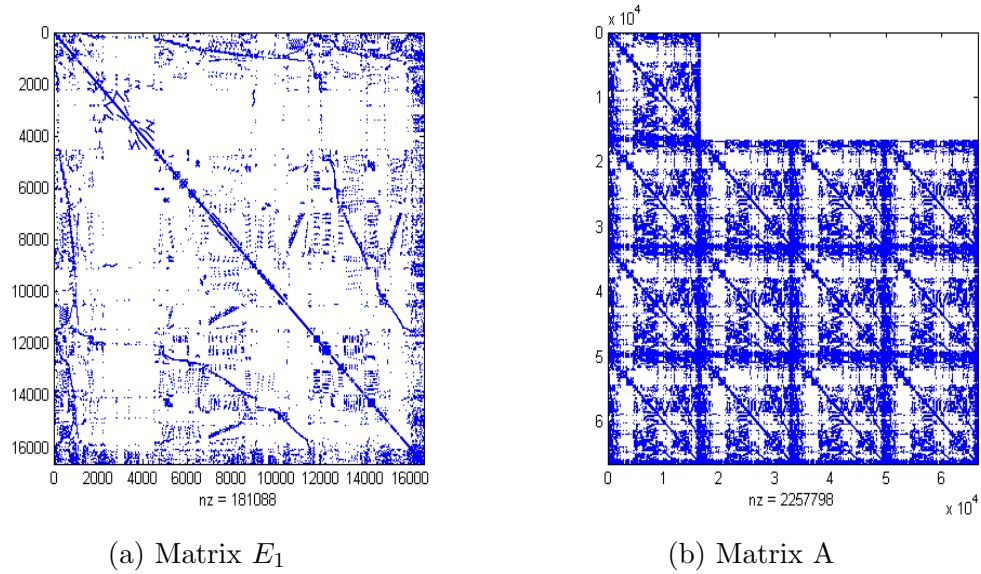


Figure. 2.14. Sparsity patterns of the matrix E_1 and the matrix A of thermo-elastic system

2.4.3 Thermo-Elastic System

In matrix vector form, the coupled thermo-elastic system [48] can represent as

$$\begin{bmatrix} E_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_a(t) \\ \dot{x}_b(t) \end{bmatrix} = \begin{bmatrix} J_1 & 0 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix} + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u(t) \quad (2.24)$$

$$y(t) = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_a(t) \\ x_b(t) \end{bmatrix}.$$

where $x_a(t) \in \mathbb{R}^{16626}$ and $x_b(t) \in \mathbb{R}^{49878}$ are state vectors. The input matrix $B \in \mathbb{R}^{20 \times 66504}$ and the output matrix $C \in \mathbb{R}^{66504 \times 27}$. The sparsity patterns of matrix E and A are shown in figure 2.14 which shows low sparsity for the large model.

Applying RKSM (algorithm 3), we compute the low rank solution of the Lyapunov equations. After 70 iterations we get the low rank factor $R \in \mathbb{R}^{16626 \times 726}$ for controllability gramian and $L \in \mathbb{R}^{16626 \times 429}$ for observability gramian.

Chapter 3

Application of Lyapunov Equation for Model Order Reduction

This chapter presents the model reduction of index-1 descriptor system. There exist various model reduction techniques for standard state space systems such as Padé approximation, rational interpolation, balanced truncation, moment matching approximation, modal truncation. Last few years, BT is used as a suitable method for MOR. This method produces a reduced model with good global accuracy. However, it requires the solution of two Lyapunov equations as well as matrix factorization and products. Chapter 2 has already discussed the low-rank approximate solutions for large-scale continuous-time algebraic Lyapunov equations.

A basic concept of MOR has been presented in the first section. Section 3.2 represents the MOR of index-1 descriptor system. Last section shows some numerical examples that illustrate the application to model reduction.

3.1 Concepts of Model Order Reduction

The purpose of MOR is to replace a large model by a smaller one, which preserves the essential properties of the original model. This smaller system must approximate the larger system, in a sense that the input-output behavior of this system

is comparable to the original within a certain accuracy. Hence, it is desirable to have a method which is automatic.

Originally, MOR was developed in the area of systems and control theory for reducing the complexity of dynamical systems. Nowadays, MOR is a flourishing field of research both in systems and control theory.

For convenience, consider the large-scale generalized LTI continuous time system

$$\Sigma : \begin{cases} Ex(t) = Ax(t) + Bu(t); & x(t_0) = x_0, \quad t \geq t_0 \\ y(t) = Cx(t) + Du(t), \end{cases} \Leftrightarrow \Sigma := \left(\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right) \quad (3.1)$$

where $E, A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, $C \in \mathbb{R}^{q \times n}$ and $D \in \mathbb{R}^{q \times p}$.

The LTI system (3.1) is widely used for simulation in different applications of science and engineering such as signal processing, microelectronics, fluid dynamics, control theory, multibody dynamics with constraints, electrical circuit simulations, VLSI chip design or mechanical systems simulation. For studying complex physical phenomena, it is easy to make direct numerical simulation. Then resulting system converts to large dimension so it make difficulty to analyze the system. Hence, there is a demand for smaller models that can describe large complex systems fairly. These smaller system, called reduced order model, enable more cost efficient.

The main idea of MOR is based on projection technique. Let the trajectory of x in (3.1) is contained in a low dimensional subspace \mathcal{U} and \mathcal{V}^\perp is a complementary subspace of \mathcal{U} . Let the column of $U \in \mathbb{R}^{n \times r}$ form the basis of \mathcal{U} and the column of $V \in \mathbb{R}^{n \times r}$ be the basis of the subspace \mathcal{V} .

Then $V^T U = I$ and UV^T is a projector, which projects x onto \mathcal{U} along \mathcal{V}^\perp . The ROM is obtained by approximating x by its projector $x \approx UV^T x$.

Replacing x with the approximation $x \approx U\hat{x} =: \tilde{x}$ so that

$$V^T r = 0,$$

where $r = E\dot{\tilde{x}} - A\tilde{x} - Bu$.

Therefore, one can replace the system (3.1) into the reduced-order model

$$\hat{\Sigma} : \begin{cases} \hat{E}\dot{\hat{x}}(t) = \hat{A}\hat{x}(t) + \hat{B}u(t), \\ \hat{y}(t) = \hat{C}\hat{x}(t) + \hat{D}u(t), \end{cases} \Leftrightarrow \hat{\Sigma} := \left(\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array} \right) \quad (3.2)$$

where $\hat{x}(t) = V^T x(t) \in \mathbb{R}^r$, $\hat{E} = V^T E U \in \mathbb{R}^{r \times r}$, $\hat{A} = V^T A U \in \mathbb{R}^{r \times r}$, $\hat{B} = V^T B \in \mathbb{R}^{r \times p}$, $\hat{C} = C U \in \mathbb{R}^{q \times r}$ and $\hat{D} = D \in \mathbb{R}^{q \times p}$ ($r \ll n$).

The number of inputs and outputs are same as for the original system and the corresponding TFM is

$$\hat{G}(s) = \hat{C}(s\hat{E} - \hat{A})^{-1}\hat{B} + \hat{D}, \quad (3.3)$$

which approximates the original TFM

$$G(s) = C(sE - A)^{-1}B + D. \quad (3.4)$$

The reduced model (3.2) should be satisfied some of the following requirements:

1. The approximation error must be small *i.e.*, $r \ll n$, and the output error $\|y(t) - \hat{y}(t)\|$ should be minimized for all inputs $u(t)$ in an appropriate norm.
2. Physical or numerical properties of the original system such as stability and passivity should be preserved during the MOR process.
3. The procedure should be computed a global error bound for the reduced model.
4. The procedure must be computationally stable and efficient.
5. The reduced-order model should preserve the structure of the original model.

Applying Laplace transformation, the system (3.1) yields, $\hat{Y}(s) = \hat{G}(s)U(s)$ and $Y(s) = G(s)U(s)$.

Then the error bound of the reduced model can be represented as

$$\|Y - \hat{Y}\|_{\mathcal{L}_2} = \|GU - \hat{G}U\|_{\mathcal{L}_2} \leq \|G - \hat{G}\|_{\mathcal{H}_\infty} \|U\|_{\mathcal{L}_2},$$

where \mathcal{H}_∞ norm is defined as

$$\|G\|_{\mathcal{H}_\infty} = \sup_{w \in \mathcal{R}} \sigma|G(jw)|.$$

It is clear that $\|Y - \hat{Y}\|_{\mathcal{H}_2}$ will be minimized if $\|G - \hat{G}\|_{\mathcal{H}_\infty}$ can be minimized.

3.2 Model Order Reduction of Index-1 Descriptor System

Balanced Truncation is one of the most commonly used model-reduction schemes. The motivation of BT is that the HSVs are the invariant of the system. If a system is balanced, the smallest HSVs can be easily separated from the gramians. Therefore, one can truncate the smallest HSVs, i.e., the unimportant states from the system which are difficult to observe, and to control, so that only important information of the original system is retained in the reduced model.

To obtain a reduced order model as (3.2), one needs to perform the following steps:

Firstly, the controllability gramian $P \in \mathbb{R}^{n_g \times n_g}$ and the observability gramian $Q \in \mathbb{R}^{n_g \times n_g}$ are computed by solving the Lyapunov equations

$$APE^T + EPA^T = -BB^T \quad (3.5)$$

$$\text{and } A^TQE + E^TQA = -C^TC, \quad (3.6)$$

respectively. Since the gramian factors P and Q are symmetric and positive semi-definite, so they have symmetric decomposition $P \approx RR^*$ and $Q \approx LL^*$, where R and L are the Choleski factors.

Using the low-rank gramian factors R and L , compute the SVD of their product as follows:

$$R^TE L = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}, \quad (3.7)$$

where $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_k, \dots, \sigma_n)$ and $\sigma_i > \sigma_{i+1} \geq 0$, ($i = 1, 2, \dots, n$) are the HSVs and Σ_1 be the first $k \times k$ block.

Finally, construct left and right balancing and truncating transformations as

$$T_L := RU_1\Sigma_1^{-\frac{1}{2}}, \quad T_R := LV_1\Sigma_1^{-\frac{1}{2}}. \quad (3.8)$$

Applying transformations T_L and T_R , one can approximate the original system Σ into lower $k(\ll n)$ dimensional reduced order LTI system with the relations

$$\hat{E} = T_L^T E T_R, \quad \hat{A} = T_L^T A T_R, \quad \hat{B} = T_L^T B, \quad \hat{C} = C T_R.$$

Algorithm 4: Balanced Truncation for generalized systems

input : System matrices E, A, B, C, D_a ; a tolerance τ for reduced-order model.
output: Matrices $\hat{E}, \hat{A}, \hat{B}, \hat{C}$ of stable reduced-order model.

- 1 Compute low-rank solution factors R, L of the system Gramians;
- 2 Compute and partition a (thin) SVD

$$\text{svd}(R^T E L) =: U \Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1 & V_2 \end{bmatrix}^T;$$
- 3 Construct $T_L := R U_1 \Sigma_1^{-\frac{1}{2}}$ and $T_R := L V_1 \Sigma_1^{-\frac{1}{2}}$;
- 4 Compute the reduced order model,

$$\hat{E} := T_L^T E T_R, \hat{A} := T_L^T A T_R, \hat{B} := T_L^T B, \hat{C} := C T_R$$
 and the error bound

$$\delta = 2 \sum_{i=k+1}^n \sigma_i \leq \tau;$$

It can easily verified that $T_L^T T_R = I$, hence BT is known as Petrov-Galerkin projection method.

Preserving the k dominant HSVs by truncating the rest yields the ROM

$$\hat{\Sigma} : \begin{cases} \hat{E} \dot{\hat{x}}(t) = \hat{A} \hat{x}(t) + \hat{B} u(t), \\ \hat{y}(t) = \hat{C} \hat{x}(t) + \hat{D} u(t) \end{cases} \Leftrightarrow \hat{\Sigma} := \left(\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array} \right) \quad (3.9)$$

This model is stable with the HSVs $\sigma_1, \sigma_2, \dots, \sigma_k$. The above procedure is fruitful if it satisfies the global error bound

$$\|G(\cdot) - \hat{G}(\cdot)\|_{\mathcal{H}_\infty} \leq 2 \sum_{i=k+1}^n \sigma_i, \quad (3.10)$$

where $G(s)$ and $\hat{G}(s)$ are called the transfer functions of the full and reduced model, respectively, and $\|\cdot\|_{\mathcal{H}_\infty}$ denotes the \mathcal{H}_∞ -norm. The relation (3.10) is indeed *a priori* error bound. Hence, the reduced model can be automatically obtained by choosing k according to the desired accuracy.

Given a error tolerance, one can easily truncate the required HSVs and determine the reduced order model. That means for a given system, the method can generate a best approximate system and preserves the stability of the original systems, i.e., if the given system is stable, the method ensures a stable reduced system. The algorithm is often referred to as the *Square-Root* (SR) method for balanced truncation. The resulting algorithm is summarized in Algorithm 4.

The algorithm outlined above is applicable for a generalized system to find a reduced order model. However, the idea is extended for a index-1 descriptor system

Algorithm 5: Balanced Truncation for index-1 descriptor systems

Input : $E_1, E_2, J_1, J_2, J_3, J_4, B_1, B_2, C_1, C_2, D_a$; a tolerance τ for reduced-order model.

Output: Matrices $\hat{E}, \hat{A}, \hat{B}, \hat{C}, \hat{D}$ of the reduced system.

- 1 Compute low-rank solution factors R, L , by solving Lyapunov equations via Algorithm 3;
- 2 $L = E_1 L$;
- 3 Compute and partition a (thin) singular value decomposition
- 4 $U \Sigma V^T = \text{svd}(L^T R)$;
- 5 Truncate after k th largest singular values and compute
- 6 $U_1 = U(:, 1:k)$, $\Sigma_1 = \Sigma(:, 1:k)$, and $V_1 = V(:, 1:k)$;
- 7 Construct the transformation matrix T_L and T_R as follows:
 $T_L := R U_1 \Sigma_1^{-\frac{1}{2}}$ and $T_R := L V_1 \Sigma_1^{-\frac{1}{2}}$;
- 8 Compute

$$\hat{E}_1 = T_L^T E_1 T_R,$$

$$\hat{E}_2 = T_L^T E_2,$$

$$\hat{J}_1 = T_L^T J_1 T_R,$$

$$\hat{J}_2 = T_L^T J_2,$$

$$\hat{J}_3 = J_3 T_R,$$

$$\hat{B}_1 = T_L^T B_1,$$

$$\hat{C}_1 = C_1 T_R$$

;

- 9 Generate reduced order model

$$\hat{E} = \hat{E}_1 - \hat{E}_2 J_4^{-1} \hat{J}_3,$$

$$\hat{A} = \hat{J}_1 - \hat{J}_2 J_4^{-1} \hat{J}_3,$$

$$\hat{B} = [\hat{B}_1 - \hat{J}_2 J_4^{-1} B_2, \hat{E}_2 J_4^{-1} B_2],$$

$$\hat{C} = \hat{C}_1 - C_2 J_4^{-1} \hat{J}_3,$$

$$\hat{D} = D_a - C_2 J_4^{-1} B_2$$

as follows:

- **step-1** : Applying RKSM (algorithm 3), we compute the controllability Gramian P and the observability Gramian Q by solving the two Lyapunov equations (3.5) and (3.6), arises from the index-1 descriptor system. Then, compute the Choleski decomposition $P \approx R R^*$ and $Q \approx L L^*$. The low rank approximation of the gramians may be complex due to complex shifts. But, the Cholesky factors should be real.

- **step-2** : Following Algorithm 4, compute the SVD, and then left transformation and right transformation.
- **step-3** : Finally, construct the reduced order coefficient matrices, using these transformation matrices.

Thus, by choosing an appropriate error tolerance, we can construct the ROM for index-1 descriptor system. The resulting procedure is summarized in Algorithm 5.

3.3 Numerical Results

This section presents numerical results to illustrate the reliability of the proposed model reduction methods for descriptor systems. We apply the method to several selected test, introduced in Section 1.6 .

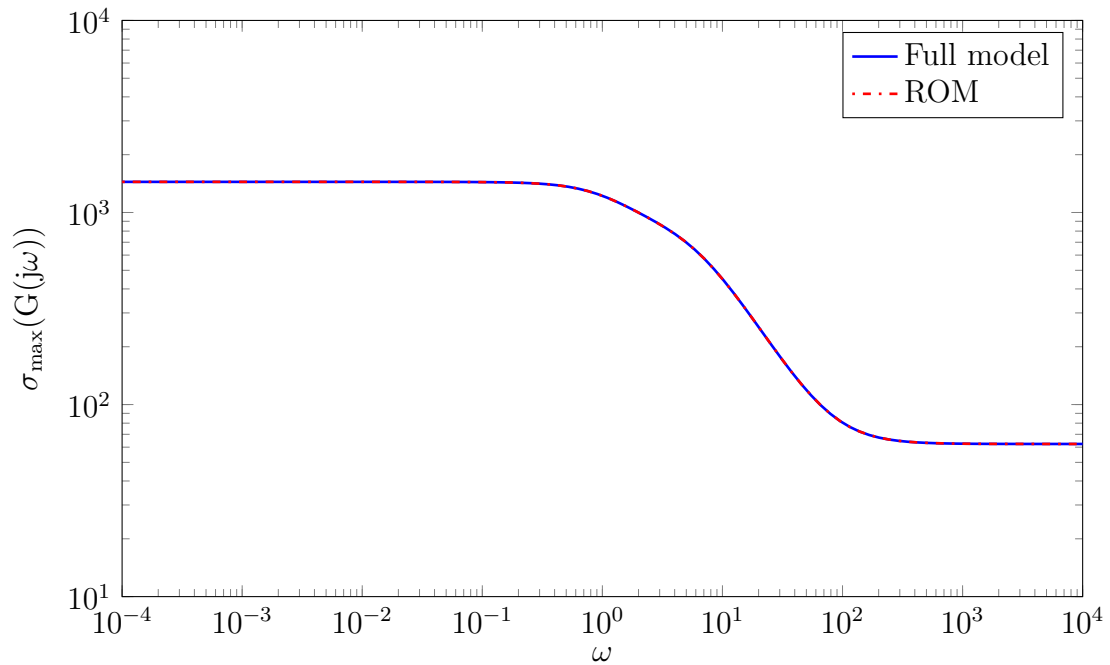
All of the following results were carried out using MATLAB (R2013b) on an Intel Core i3@1.70 GHz clock, RAM 8GB with relative machine precision $\epsilon = 1.7 \times 10^{-16}$.

3.3.1 Model Reduction of Artificial Data

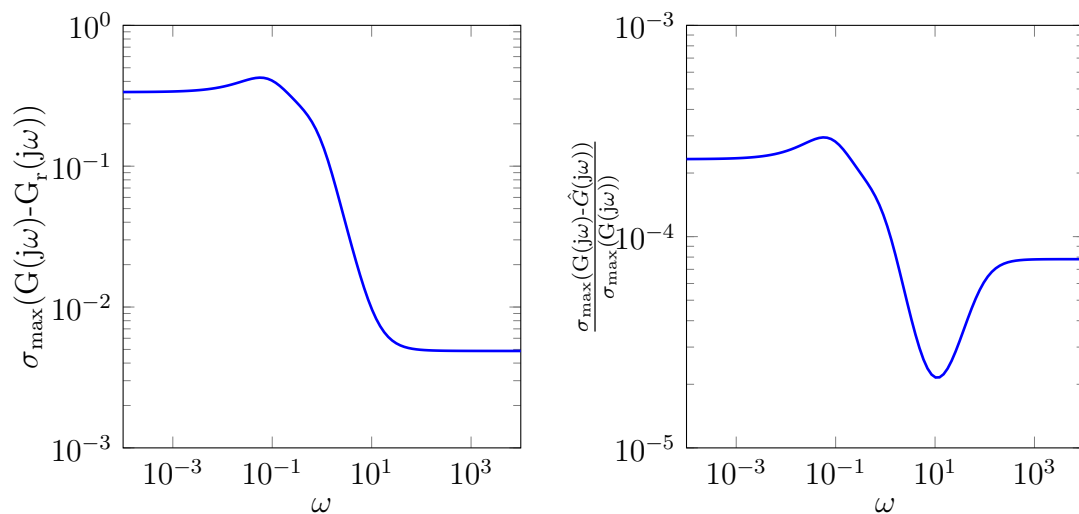
According to Section 2.4, we consider an artificial model of order 10500 for index-1 (semi-explicit) descriptor system. By eliminating the algebraic variables the system has 10000 states and the system matrices become in dense formation.

We apply RKSM (algorithm 3) to solve Lyapunov equations. We get the low rank gramian factor $R \in \mathbb{C}^{3078 \times 315}$ for controllability gramian and $L \in \mathbb{C}^{3078 \times 300}$ for observability gramian. And then applying BT the system reduces to 19 dimensional reduced model.

Figure 3.1a shows the frequency response of the full system and its 19 order reduced system. Both the systems have same largest singular values in frequency domain 10^{-4} to 10^4 . The absolute errors and relative errors are shown in figure 3.1b and figure 3.1c, respectively. The errors show better accuracy of RKSM.



(a) Sigma plots



(b) Absolute errors

(c) Relative errors

Figure. 3.1. The frequency responses of original system and 19 reduced-order systems in frequency domain and its errors.

3.3.2 Model Reduction of Power System Data

In this section, we discuss the model reduction of power system models. Section 2.4 (in table 2.2) shows the summary of them. Since the numerical results of all data sets are same, we only explain the results for largest model Mod-3.

Applying RKSM (Algorithm 3), we compute the low rank controllability and observability gramian factors for all the models by solving their respective Lyapunov

Table. 3.1. Comparisons of original and reduced models for all data.

Model	Dimension		Error	
	Original	Reduced	Absolute	Relative
Mod-1	7 135	72	2×10^{-2}	1.1×10^{-3}
Mod-2	9 735	70	9×10^{-3}	3×10^{-3}
Mod-3	21 128	70	3.4×10^{-2}	5.4×10^{-2}

equations. Then, the reduced models are computed for all data by BT method.

Table. 3.2. BT tolerances and dimensions of reduced model Mod-3.

Model	Tolerance	Dimension of ROM
Mod-3	10^{-4}	143
	10^{-3}	118
	10^{-2}	94
	10^{-1}	70
	10^0	48

The comparisons of original and reduced models are depicted in table 3.1 with their absolute and relative deviations. The truncation tolerance for all data set is considered by 10^{-1} . However, the dimension of the ROM can be decreased or increased by changing the tolerance if desired or required.

To compute the low rank controllability and observability gramian factors, we solve two Lyapunov equations. We apply RKSM (algorithm 3) with shifted linear system in sparse form. We get the solution of low rank gramian factor $R \in \mathbb{C}^{3078 \times 315}$ for controllability gramian and $L \in \mathbb{C}^{3078 \times 300}$ for observability gramian. 6 proper shift parameters are considered in each cycle.

For the initial shift, we compute the largest eigen value of the pair $(-A, E)$, using MATLAB `eigs`, which is 10778. Note that each shifted linear system $(A - \mu_i E)$ is solved using a sparse direct solver (backslash in MATLAB). To compute R and L the GRKSM is performed with normalized residual norm tolerance 10^{-13} . The computed gramians are applied to MOR (Algorithm 5) with BT tolerance 10^{-1} . Using the 70 HSVs, we are formed the transformation matrices $T_L \in \mathbb{C}^{3078 \times 70}$ and $T_R \in \mathbb{C}^{3078 \times 70}$. Finally, we construct the reduced system of 70 order, by computing the reduced system matrices.

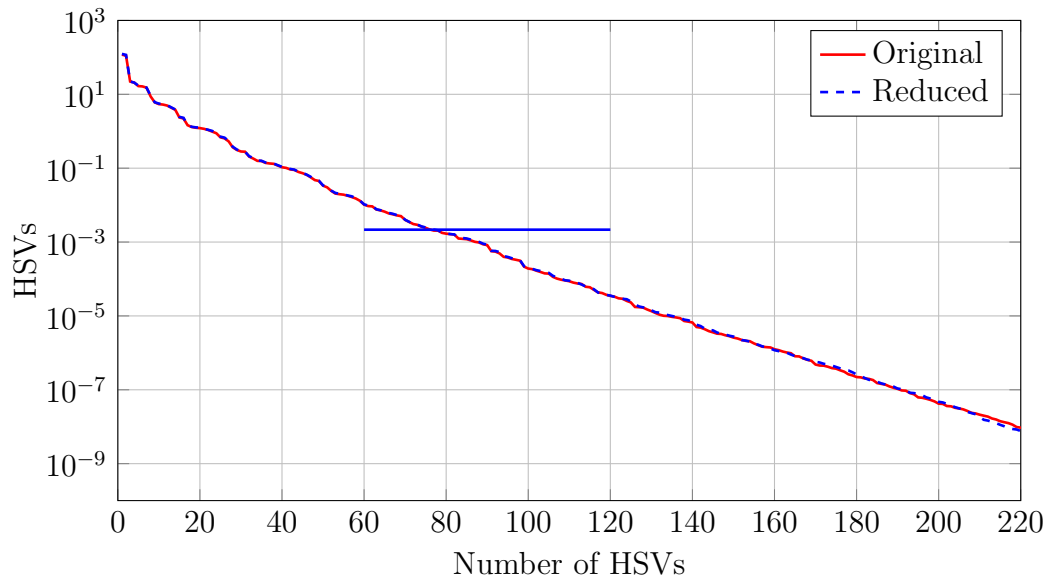


Figure. 3.2. Comparisons of Hankel singular values of original system and reduced system.

Table 3.2 shows the different ROM for Mod-3 by changing BT tolerance. For Mod-3, figure 3.2 shows that the largest HSVs of 3078 state system and 70 order reduced model are approximately same. In figure 3.3, one can observe that the reduced system is asymptotically stable i.e., all the eigenvalues lie in the left complex half plane and the eigenvalues are closed to the imaginary axis.

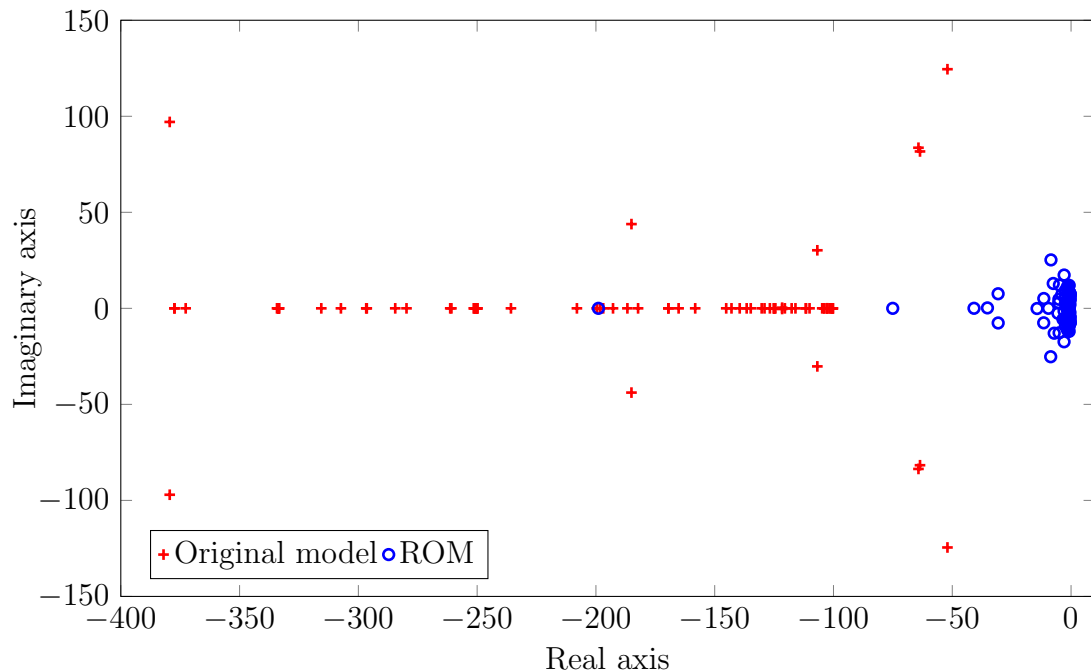
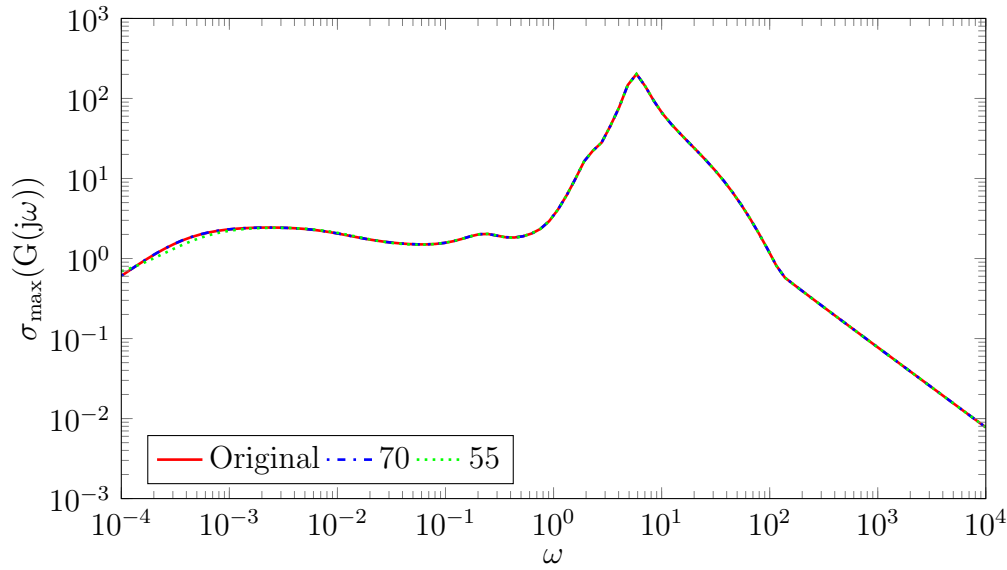
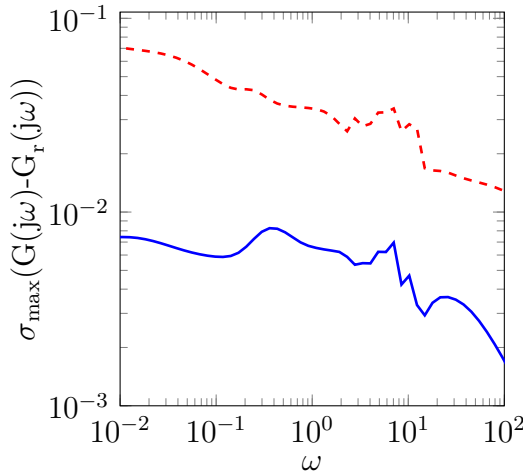


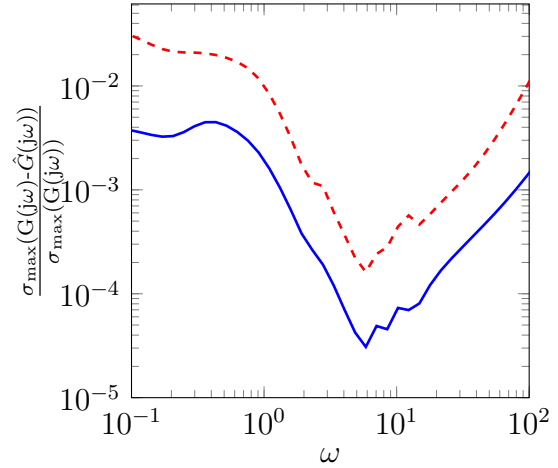
Figure. 3.3. Eigenvalues of the original system and reduced model.



(a) Sigma plots.



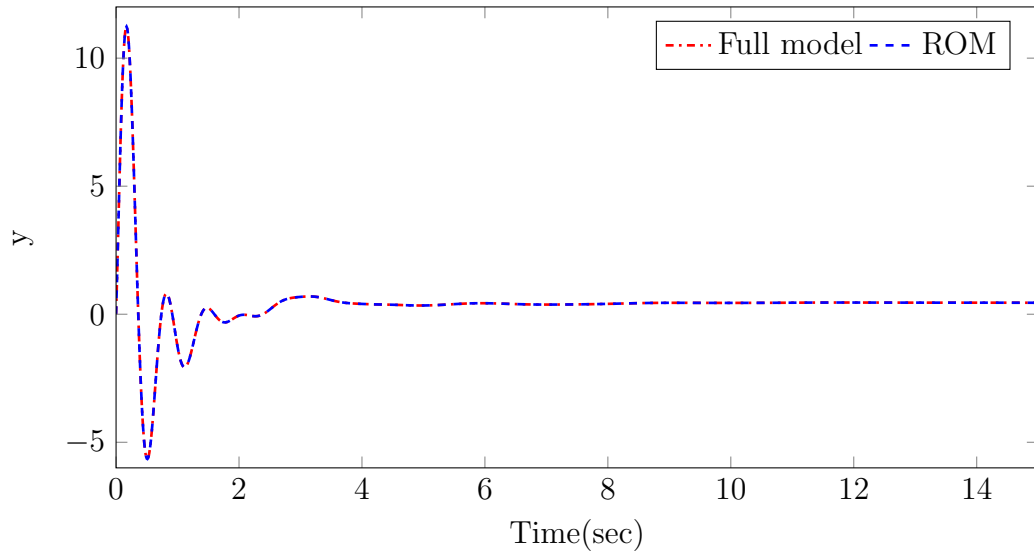
(b) Absolute errors



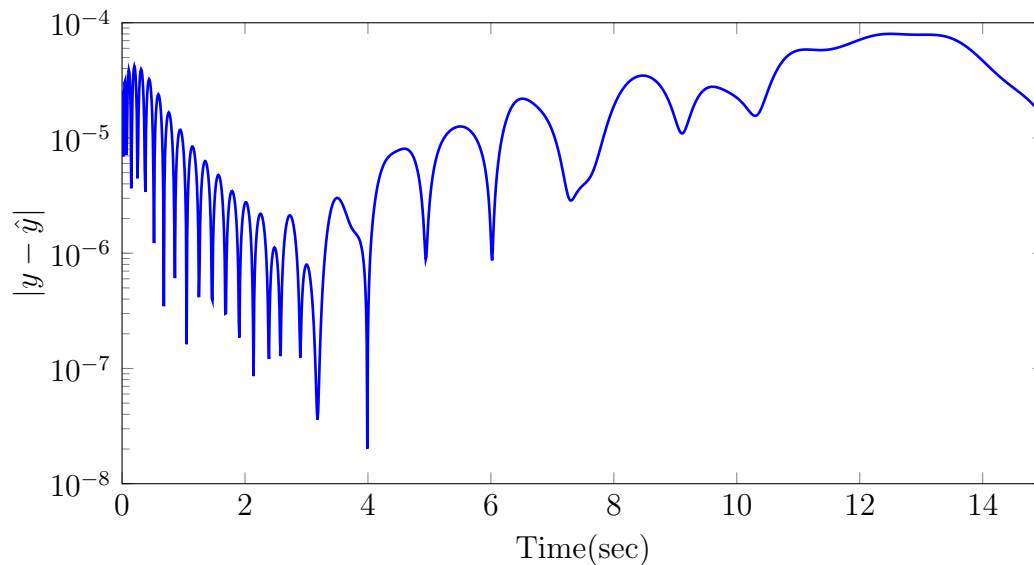
(c) Relative errors

Figure. 3.4. Errors between the original model, 70 and 55 dimensional reduced systems computed by RKSM using the system Mod-3.

The comparisons of largest singular values for original model, 70 state and 55 state reduced systems are shown in figure 3.4 in frequency domain over the frequency (ω) range 10^{-4} to 10^4 . Figure 3.4a shows the frequency responses of full system, 70 order and 55 order reduced systems. The absolute errors and the relative errors of the frequency responses of full and reduced systems are depicted in figure 3.4b and figure 3.4c, respectively, which shows that these are below to the truncation tolerance 10^{-4} . One can also compute lower dimensional ROMs if they are required or desired. We see that the model below 55 dimension plays a little bit worse match in high and low frequency.



(a) Step responses.



(b) Absolute deviations.

Figure. 3.5. Time domain responses for full system and reduced model, and their absolute deviations.

For time response, we use an implicit Euler method with fixed time step size 10^{-2} . Figure 3.5a shows the step response from 1st input to 1st output in time domain for the original system and 70 state reduced order models, respectively and the absolute deviations are depicted in figure 3.5b. and the of the frequency responses for the individual component of the transfer function.

On the other hand, frequency responses and their relative deviations are shown in figure 3.6 for the individual component of the transfer function. For example figure 3.6a shows the SISO relation (i.e., 1st input to 1st output) of original and

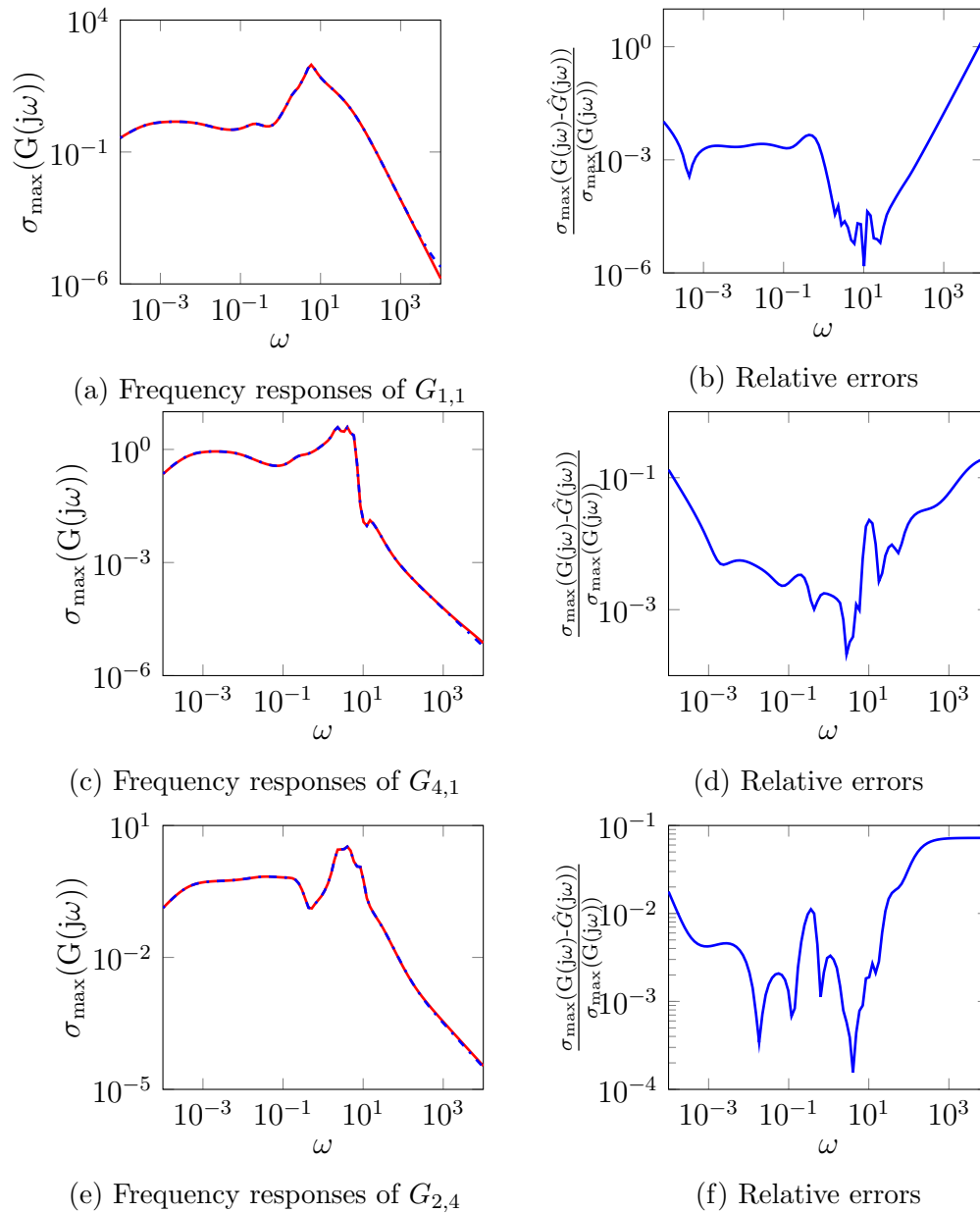


Figure 3.6. Each rows shows the relation of 1st input to 1st output, 4th input to 1st output and 2nd input to 4th output between full and reduced order models with respective relative errors.

70 order reduced model. The relative errors between original and reduced model of the respective relation are also shown in the same figure.

From these figures we can notice that the generated ROM ensures good quality and hence they can be used for controller design, simulation or optimization.

3.3.3 Reduced Order Model for Mod-1 and Mod-2

The ROM for Mod-1 and Mod-2 are computed analogously. We only show major results for these two models. Table 3.3 shows the different ROM for Mod-1 with different tolerances. Similarly, the different ROM of Mod-2 are shown in table 3.4 with different tolerances.

Table. 3.3. BT tolerances and dimensions of reduced model Mod-1.

Model	Tolerance	Dimension of ROM
Mod-1	10^{-4}	121
	10^{-3}	102
	10^{-2}	87
	10^{-1}	72
	10^0	54

Table. 3.4. BT tolerances and dimensions of reduced model Mod-2.

Model	Tolerance	Dimension of ROM
Mod-2	10^{-4}	136
	10^{-3}	111
	10^{-2}	89
	10^{-1}	70
	10^0	48

3.3.4 Model Reduction of Thermo-Elastic System

In subsection 2.4.3 we have computed the low rank gramian factors of Thermo-Elastic System. Applying BT method, we get 65 order reduce model. Figure 3.7a shows the frequency responses of full system and 65 order reduced system. The absolute errors and the relative errors of the frequency responses of original and reduced system are depicted in figure 3.7b and figure 3.7c, respectively.

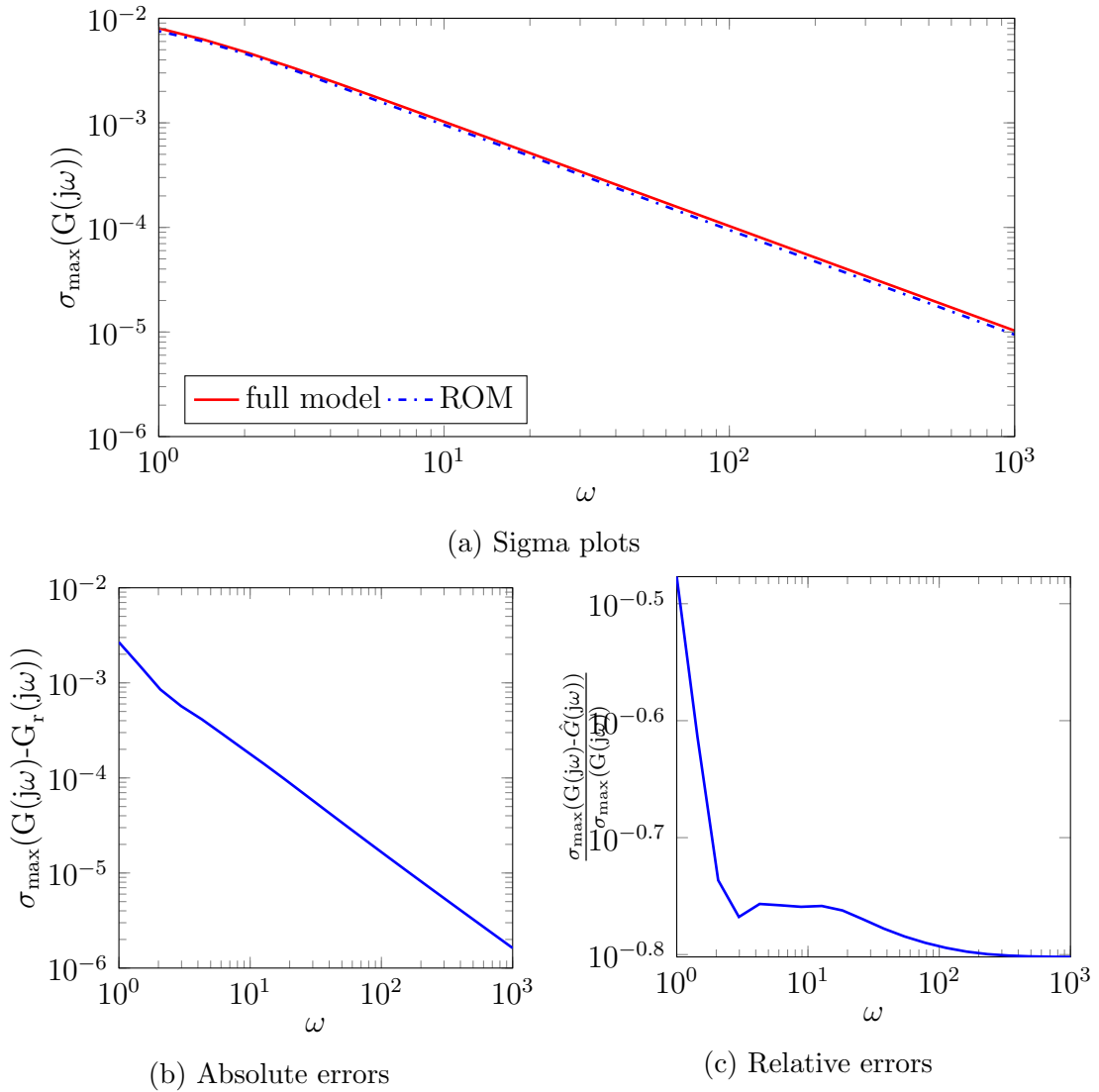


Figure. 3.7. Frequency responses of original system and 65 reduced-order systems in frequency domain and its errors.

3.4 Comparison with LRCF-ADI

In this section, we briefly discuss the comparison of our proposed method with the existing LRCF-ADI method. In [24], authors have discussed a balancing based method for the model reduction of index-1 descriptor system. They have applied the LRCF-ADI method to solve the Lyapunov equation. We know that LRCF-ADI can only be applied to the asymptotically stable system. Note that the power system models are an asymptotically unstable. To circumvent the problem, authors used so called α -shift approach to stabilize the system. That is, the matrix A is replaced by the shifted matrix $\tilde{A} = A - \alpha I$, where I is an identity matrix

Table. 3.5. Comparisons of SLRCF-ADI method and RKSM

Model	SLRCF-ADI		RKSM	
	Iterations	ROM	Iterations(P+Q)	ROM
Mod-1	100	92	(35 + 35)	72
Mod-2	-	-	(40 + 40)	70
Mod-3	150	106	(60 + 60)	70

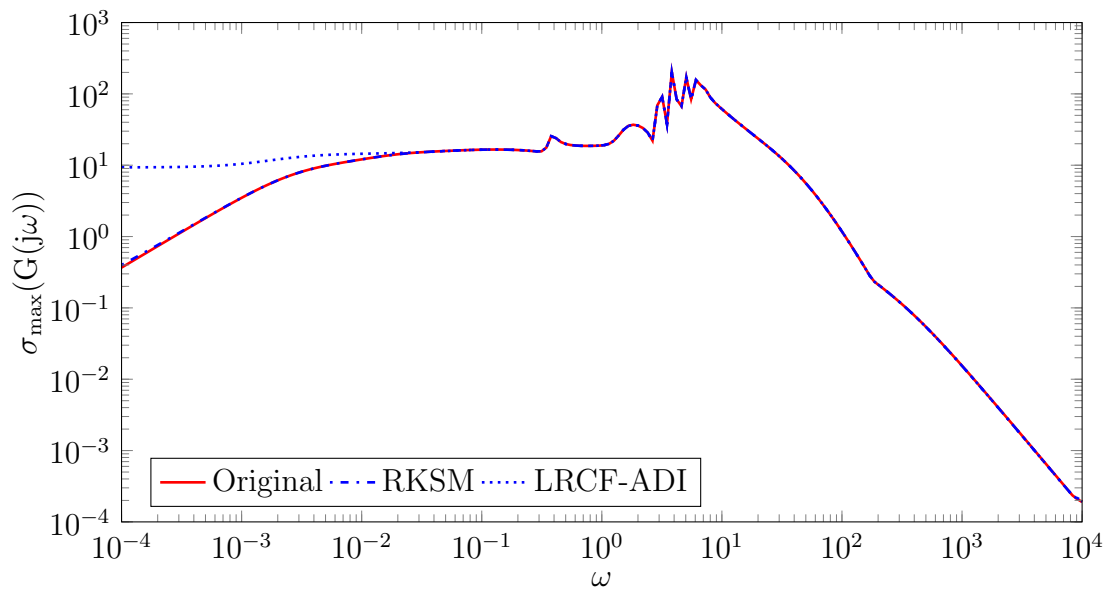


Figure. 3.8. Comparisons between full model, 67 order ADI and 67 order RKSM.

and α be a positive real number. In the case of ROM, $\hat{A} = T_L^T \tilde{A} T_R + \alpha I$, where $I = T_L^T T_R$.

But, RKSM do not require to stabilize the system. In addition, we are used efficient technique to compute the shift parameters for better convergence of the RKSM. Table 3.5 shows that our proposed method is better than the LRCF-ADI.

To compare the RKSM with the LRCF-ADI method, we compute 67 dimensional reduced model using the algorithm LRCF-ADI for Mod-1. The frequency responses of original system, 67-order system by RKSM and 67-order system by LRCF-ADI are depicted in figure 3.8. This shows that the LRCF-ADI gives worst match with the original model but in this case RKSM shows good accuracy.

Chapter 4

Conclusions and Future Work

4.1 Summary

Two important issues have been presented in this thesis. Krylov subspace based projection technique has been discussed for the solution of Lyapunov equations in Chapter 2. And in Chapter 3 the low rank solution has been applied for balancing based model reduction .

A projection technique to compute the solution of Lyapunov equation has been developed. The proposed algorithm has been applied for the large-scale index-1 descriptor systems. The method was closely related to the well established technique based for standard system. The idea for structured descriptor system has been discussed in Chapter 2. It is observed that the technique is efficient for highly sparse, large systems and also the computational cost is cheap. A recycling technique has been implemented to produce the columns of the projection matrix. A new shift computation strategy has been performed to construct rational Krylov subspace. It was shown that this new version has been making a satisfactory to compute rational Krylov subspace iteratively. A normalized residual technique is used to stop the proposed methods.

On the other hand, the balancing based projection technique has been constructed for model reduction of descriptor systems. But, the most expensive part of the proposed MOR approach was to solve two Generalized Continuous-time Algebraic Lyapunov Equations (GCALE). RKSM has been applied to find the low rank solution of GCALE efficiently.

Finally, the accuracy of the proposed method is discussed by Hankel singular values, frequency responses and time domain responses for several test-systems. It is observed that the proposed method produces a very good approximation of the original system. A global error bound is also applied for the approximate system.

4.2 Future Work

The work may be extended for future research possibilities. This thesis opens the path to a wide range of system and control theory. Some of them are rather theoretic whereas others lie more in the range of computational scientific aspects. The technique can be used to solve many problems as well as for higher index DAEs. We have observed that the relative changes in the low-rank factors are very slow for the supersonic inlet flow model and need to be investigated further. In RKSM algorithm, at each iteration we used direct sparse solvers to solve the linear system. Future research would be conducted to find the better way for the solution of the linear systems. In this case we can exploit the shift computation strategy.

In future research, this techniques would be used to solve the algebraic Riccati equations. This work has unlocked some new aspects to stabilize the unstable large-scale dynamical systems. But the fact of preserving the cyclic structure of system matrices in the iteration is again the key issue that should be investigated.

References

- [1] A. C. Antoulas, D. C. Sorensen, and S. Gugercin, “A survey of model reduction methods for large-scale systems,” *Contemp. Math.*, vol. 280, pp. 193–219, 2001.
- [2] P. Benner, V. Mehrmann, and D. Sorensen, Eds., *Dimension reduction of large-scale systems*, ser. Lecture Notes in Computational Science and Engineering. Berlin/ Heidelberg, Germany: Springer-Verlag, 2005, vol. 45.
- [3] W. H. A. Schilders, H. A. Vorst, and J. Rommes, *Model order reduction: theory, research aspects and applications*. Berlin: Springer-Verlag, 2008, vol. 13.
- [4] M. M. Uddin, “Model reduction for piezo-mechanical system using balanced truncation,” Master’s thesis, Stockholms University, Stockholm, Sweden, 2011. [Online]. Available: http://www.qucosa.de/fileadmin/data/qucosa/documents/7822/Master_Thesis_Uddin.pdf
- [5] A. C. Antoulas, *Approximation of large-scale dynamical systems*. Advances in Design and Control. SIAM, Philadelphia, PA, 2005, vol. 6.
- [6] V. Mehrmann and T. Stykel, “Balanced truncation model reduction for large-scale systems in descriptor form,” in *Dimension Reduction of Large-Scale Systems*, ser. Lecture Notes in Computational Science and Engineering. Berlin/Heidelberg, Germany: Springer-Verlag, 2005, pp. 83–115.
- [7] B. C. Moore, “Principal component analysis in linear systems: Controllability, observability, and model reduction,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 17–32, 1981.
- [8] P. Feldmann and R. W. Freund, “Efficient linear circuit analysis by Padé approximation via the Lanczos process,” vol. 14, pp. 639–649, 1995.

-
- [9] J.-R. Li, “Model reduction of large linear systems via low rank system Gramians,” Ph.D. Thesis, Massachusetts Institute of Technology, 2000.
- [10] E. J. Davison, “A method for simplifying linear dynamic systems,” vol. AC-11, pp. 93–101, 1966.
- [11] E. J. Grimme, “Krylov projection methods for model reduction,” Ph.D. dissertation, University of Illinois at Urbana, Champaign, 1997.
- [12] P. Benner, J.-R. Li, and T. Penzl, “Numerical solution of large Lyapunov equations, Riccati equations, and linear-quadratic control problems,” *Num. Alg. Appl.*, vol. 15, no. 9, pp. 755–777, 2008.
- [13] K. Jbilou, “ADI preconditioned Krylov methods for large Lyapunov matrix equations,” vol. 432, no. 10, pp. 2473–2485, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0024379509006582>
- [14] J.-R. Li and J. White, “Low rank solution of Lyapunov equations,” *SIAM Matrix*, vol. 24, no. 1, pp. 260–280, 2002.
- [15] S. Gugercin, D. Sorensen, and A. Antoulas, “A modified low-rank Smith method for large-scale Lyapunov equations,” *Numer. Algorithms*, vol. 32, no. 1, pp. 27–55, 2003.
- [16] T. Penzl, “A cyclic low rank Smith method for large, sparse Lyapunov equations with applications in model reduction and optimal control,” Chemnitz University of Technology, 09107 Chemnitz, FRG, Tech. Rep. SFB393/98-6, 1998. [Online]. Available: <http://www.tu-chemnitz.de/sfb393/sfb98pr.html>
- [17] I. Jaimoukha and E. Kasenally, “Krylov subspace methods for solving large Lyapunov equations,” *SIAM J. Matrix Anal.*, vol. 31, pp. 227–251, 1994.
- [18] K. Jbilou and A. J. Riquet, “Projection methods for large Lyapunov matrix equations,” vol. 415, no. 2-3, pp. 344–358, 2006.
- [19] V. Simoncini, “A new iterative method for solving large-scale Lyapunov matrix equations,” *SIAM Sci. Comp.*, vol. 29, no. 3, pp. 1268–1288, 2007.
- [20] Y. Saad, “Numerical solution of large Lyapunov equation,” in *Signal Processing, Scattering, Operator Theory and Numerical Methods*, 1990, pp. 503–511.

- [21] U. Baur, “Control-Oriented Model Reduction for Parabolic Systems,” PhD Thesis, Inst. f. Mathematik, Technische Universität Berlin, Berlin, 2008. [Online]. Available: <http://www.nbn-resolving.de/urn:nbn:de:kobv:83-opus-17608>
- [22] P. Benner, E. Quintana-Ortí, and G. Quintana-Ortí, “Balanced truncation model reduction of large-scale dense systems on parallel computers,” *Math. Comput. Model. Dyn. Syst.*, vol. 6, no. 4, pp. 383–405, 2000.
- [23] C. Kenney and A. J. Laub, “Rational iterative methods for the matrix sign function,” vol. 12, pp. 273–291, 1991.
- [24] F. Freitas, J. Rommes, and N. Martins, “Gramian-based reduction method applied to large sparse power system descriptor models,” *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 1258–1270, 2008.
- [25] M. S. Hossain, M. M. Uddin, and M. F. Uddin, “Iterative methods for solving large sparse lyapunov equations and application to model reduction of index-1 differential-algebraic equations,” *Numerical Algebra, Control and Optimization (AIMS)*, under review, 2017.
- [26] M. M. Uddin, M. S. Hossain, and M. F. Uddin, “Rational krylov subspace method (rksm) for solving the lyapunov equations of index-1 descriptor systems and application to balancing based model reduction,” in *9th International Conference on Electrical & Computer Engineering (ICECE) 2016*. IEEE, 2016, pp. 451–454.
- [27] V. Druskin and V. Simoncini, “Adaptive rational Krylov subspaces for large-scale dynamical systems,” *Systems and Control Letters*, vol. 60, no. 8, pp. 546–560, 2011. [Online]. Available: <https://doi.org/10.1016/j.sysconle.2011.04.013>
- [28] M. M. Uddin, “Computational methods for model reduction of large-scale sparse structured descriptor systems,” PhD Thesis, Otto-von-Guericke Universität Magdeburg, Magdeburg, 2006.
- [29] P. Benner, P. Kürschner, and J. Saak, “Self-generating and efficient shift parameters in ADI methods for large Lyapunov and Sylvester equations,” vol. 43, pp. 142–162, 2014.

-
- [30] T. Stykel, “Analysis and numerical solution of generalized Lyapunov equations,” Ph.D. thesis, Institut für Mathematik, Technische Universität Berlin, Berlin, Germany, 2002, available from <http://www.math.tu-berlin.de/~stykel/Publications/disser.ps>.
- [31] P. Kunkel and V. Mehrmann, *Differential-Algebraic Equations: Analysis and Numerical Solution*. European Mathematical Society, 2006.
- [32] B. Datta, *Numerical methods for linear control systems*. Elsevier Academic Press, London, UK, 2004.
- [33] R. Bartels and G. Stewart, “Solution of the matrix equation $AX+XB=C$ [F4],” *Communications of the ACM*, vol. 15, no. 9, pp. 820–826, 1972.
- [34] D. C. Sorensen and Y. Zhou, “Direct methods for matrix Sylvester and Lyapunov equations,” *J. of Appl. Math*, vol. 6, pp. 277–303, 2003.
- [35] D. Kressner, “Block variants of hammarling methods for solving lyapunov equations,” *ACM Trans. Math. Softw.*, vol. 34, no. 1, pp. 1–15, 2008.
- [36] T. Penzl, “Numerical solution of generalized Lyapunov equations,” *Adv. Comp. Math.*, vol. 8, pp. 33–48, 1997.
- [37] C. Kenney and A. J. Laub, “The matrix sign function,” vol. 40, no. 8, pp. 1330–1348, 1995.
- [38] P. Benner and E. S. Quintana-Ortí, “Solving stable generalized Lyapunov equations with the matrix sign function,” vol. 20, no. 1, pp. 75–100, 1999.
- [39] U. Baur and P. Benner, “Factorized solution of Lyapunov equations based on hierarchical matrix arithmetic,” vol. 78, no. 3, 2006.
- [40] A. Lu and E. Wachspress, “Solution of Lyapunov equations by alternating direction implicit iteration.” *Comput. Math. Appl.*, vol. 21, no. 9, pp. 43–58, 1991.
- [41] J. R. Li and J. White, “Low-rank solution of Lyapunov equations,” *SIAM Review*, vol. 46, no. 4, pp. 693–713, 2004.
- [42] P. Benner, P. Kürschner, and J. Saak, “Efficient handling of complex shift parameters in the low-rank Cholesky factor ADI method,” *Numerical Algorithms*, vol. 62, no. 2, pp. 225–251, 2013. [Online]. Available: <http://dx.doi.org/10.1007/s11075-012-9569-7>

-
- [43] P. Benner, P. Kürschner, and J. Saak, “An improved numerical method for balanced truncation for symmetric second order systems,” vol. 19, no. 6, pp. 593–615, 2013.
- [44] V. Simoncini and V. Druskin, “Convergence analysis of projection methods for the numerical solution of large Lyapunov equations,” *SIAM J. Numer. Anal.*, vol. 47, no. 2, pp. 828–843, 2009.
- [45] D. Kressner, “Memory efficient krylov subspace techniques for solving large-scale equations,” *In IEEE International Conference on Computer Aided Control Systems*, pp. 613–618, 2008.
- [46] S. Gugercin, T. Stykel, and S. Wyatt, “Model reduction of descriptor systems by interpolatory projection methods,” vol. 35, no. 5, p. B1010–B1033, 2013.
- [47] Y. Saad, *Iterative methods for sparse linear systems*. Philadelphia, PA: SIAM, 2003.
- [48] N. Lang, J. Saak, and P. Benner, “Model order reduction for systems with moving loads,” vol. 62, no. 7, pp. 512–522, 2014.