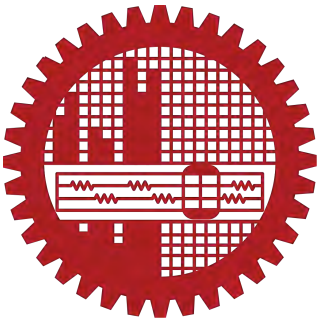
NUMERICAL STUDY ON CONTINUOUS ALGEBRAIC RICCATI EQUATION ARISING FROM LARGE-SCALE SPARSE DESCRIPTOR SYSTEMS

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

MASTER OF PHILOSOPHY
IN
MATHEMATICS

By

MAHTAB UDDIN
Student No. 1015093005P
Registration No. 10150930051, Session: Oct. 2015



Supervisor

Dr. Md. Abdul Hakim Khan
Professor
Department of Mathematics

Bangladesh University of Engineering and Technology
Dhaka-1000

February 11, 2020

The thesis entitled

NUMERICAL STUDY ON CONTINUOUS ALGEBRAIC RICCATI EQUATION ARISING FROM LARGE-SCALE SPARSE DESCRIPTOR SYSTEMS

Submitted by

MAHTAB UDDIN

Student No. 1015093005P, Registration No. 1015093005P, Session: Oct. 2015

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

Master of Philosophy in Mathematics on February 11, 2020

BOARD OF EXAMINERS

1. Dr. Md. Abdul Hakim Khan — **Chairman (Supervisor)**
Professor
Department of Mathematics
BUET, Dhaka-1000

2. Head — **Member (Ex-Officio)**
Department of Mathematics
BUET, Dhaka-1000

3. Dr. Md. Mustafizur Rahman — **Member**
Professor
Department of Mathematics
BUET, Dhaka-1000

4. Dr. Rehena Nasrin — **Member**
Professor
Department of Mathematics
BUET, Dhaka-1000

5. Dr. Mohammad Monir Uddin — **Member (External)**
Associate Professor
Department of Mathematics and Physics
North South University, Dhaka-1229.

Declaration of Authorship

I, MAHTAB UDDIN, declare that this thesis titled, 'NUMERICAL STUDY ON CONTINUOUS ALGEBRAIC RICCATI EQUATION ARISING FROM LARGE-SCALE SPARSE DESCRIPTOR SYSTEMS' and the work presented in this thesis is the outcome of the investigation carried out by me under the supervision of Dr. Md. Abdul Hakim Khan, 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: Mahtab

Date: 11.02.2020

Dedication

This thesis is dedicated

To

All of my well-wishers

Abstract

The mathematical models derived from the physical models are the pivot ingredient in science and engineering, especially, in the control theory. Most of the physical models have a large number of components with critical combinations and sophisticated designs. These models are infeasible for the computing tools according to the time dealing and memory allocation. To find the remedy of current adversity and attain desired execution results, the models are to be approximated as structure-preserving Reduced-Order Models (ROM) and machine-executable designs. Memory allocation and time management are the most eye-catching factors in the simulations of the large-scale sparse Linear Time-Invariant (LTI) systems, especially, the descriptor systems. Numerical techniques can be applied practically to control, stabilize and optimize the physical models.

In this thesis, firstly, the projection-based Rational Krylov Subspace Method (RKSM) has been proposed to compute the solution of Continuous Algebraic Riccati Equations (CARE) governed from large sparse index-1 descriptor systems. Iterative RKSM is not only time saving but also computationally feasible for memory allocation in finding the solution of the CAREs utilizing the Reduced-Order Models (ROM). The novelties of RKSM are sparsity preserving techniques and the implementation of time convenient recursive adaptive shift parameters. Secondly, the machine-independent Alternating Direction Implicit (ADI) technique based nested iterative Kleinman-Newton (KN) method has been modified and adjusted to solve the CAREs governed from large sparse index-1 descriptor systems. Then compare results achieved by the Kleinman-Newton method with that of using the RKSM.

The objective has been mainly focused on finding optimal feedback matrix for Riccati based feedback stabilization for the unstable index-1 descriptor systems applying the proposed methods. The applicability and adaptability of the proposed methods have been justified through the power system models and their transient behaviors have been analyzed.

Finally, numerical results have been shown both in tabular and graphical form to verify the robustness and accuracy of the proposed methods In addition, their comparative analysis for the target models has been illustrated in detail.

Acknowledgements

All praises for Almighty ALLAH, He is the One who has the power to fulfill the wishes of all and without His help, nothing would be a success.

I would like to express my cordial gratitude and profound honor to my thesis supervisor Prof. Dr. Md. Abdul Hakim Khan, Math, BUET for his continuous support, motivation, and guidance throughout 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 to 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 teachers for their valuable suggestions. I express my gratitude to my teachers from the Department of Mathematics, Bangladesh University of Engineering and Technology.

I am grateful to Prof. Dr. Md. Zafar Iqbal Khan, Head, Math, BUET, Prof. Dr. Md. Mustafizur Rahman, Math, BUET, and Prof. Dr. Rehena Nasrin, Math, BUET for being on my defense committee, reading my thesis and suggesting improvements.

Especially, I am extremely thankful and highly indebted to Dr. Mohammad Monir Uddin, Assoc. Prof., DMP, NSU for his generous help, encouragement, cooperation and valuable guidance during my research. Also, I deeply indebted to Md. Mamul Elahi, Asst. Prof., CSE, UIU, and Md. Abul Kalam Azad, Asst. Prof., MCE, IUT for their enthusiastic inspiration throughout the research.

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

Contents

List of Figures

List of Tables

List of Algorithms

Notations and Symbols

$\mathbb{R}$	filed of real numbers
$\mathbb{C}$	field of complex numbers
$\mathbb{C}^-$ ($\mathbb{C}^+$)	left (right) half-plane
$\mathbb{R}^{m\times n}$	set of all real matrices of order $m \times n$
$\mathbb{C}^{m\times n}$	set of all real matrices of order $m \times n$
$\subset$	subset of any matrix
$\in$	belongs to
a_{ij}	the i, j-th entry of the matrix A
$\mathrm{Re}(z)$	real part of $z \in \mathbb{C}$
$\mathrm{Im}(z)$	imaginary part of $z \in \mathbb{C}$
I_n	$n \times n$ identity matrix of order n
A^T	transpose of A
A^*	complex conjugate transpose of A
A^{-1}	inverse of A
$\subseteq$	subset
$\approx$	approximately equal to
$\ll (\gg)$	much less (greater)
$\mathrm{G}(s)$	transfer function or transfer function matrix
$\|.\|$	matrix or vector norm
$\|.\|_F$	Frobenious norm
$\|.\|_{\mathcal{H}_\infty}$	$\mathcal{H}_\infty$-norm
$\sigma_i(A)$	i-th singular value of A
$\sigma_{\max}(A)$	largest singular value of A
Σ	diagonal matrix containing singular values
$\mathrm{diag}(d_1, \cdots, d_k)$	diagonal matrix with $d_1, \cdots, d_k$ on the diagonal
$\Lambda(A, E)$	spectrum of the matrix pair (A, E)
$\lambda_j(A, E)$	j-th eigenvalue of the matrix pair (A, E)
$\mathrm{tr}(A)$	$\sum_{i=1}^{n} a_{ii}$, where a_{ii} be the diagonal entry of A
$\ker(A)$	kernel of the matrix A
$\mathcal{K}_m$	basis for the m-dimensional Krylov subspace

Abbreviations

LTI	Liner Time Invariant
CALE	Continuous Algebraic Lyapunov Equation
CARE	Continuous Algebraic Riccati Equation
CABE	Continuous Algebraic Bernoulli Equation
LQR	Linear Quadratic Regulator
MOR	Model Order Reduction
ROM	Reduced Order Model
SISO	Single Input Single Output
MIMO	Multi Input Multi Output
EVP	Eigen Value Problem
SVD	Singular Value Decomposition
ODE	Ordinary Differential Equation
PDE	Dartial Differential Equation
DAE	Differential Algebraic Equation
RSF	Real Schur Factorization
JCF	Jordan Canonical Form
BT	Balanced Truncation
RKSM	Rational Krylov Subspace Method
S-RKSM	Standard Rational Krylov Subspace Method
G-RKSM	Generalized Rational Krylov Subspace Method
G-RKSM-FB	Generalized Rational Krylov Subspace Method with Feedback
KN	Kleinman-Newton
ADI	Alternating Direction Implicit
LRCF-ADI	Low-Rank Cholesky Factor ADI
G-LRCF-ADI	Generalized LRCF-ADI
G-KN	Generalized Kleinman-Newton
G-KN-ADI	Generalized ADI based Kleinman-Newton
G-KN-LRCF-ADI	Generalized LRCF-ADI based Kleinman-Newton
IRKA	Iterative Rational Krylov Algorithm
`care`	MATLAB library command for solving CARE
`lyap`	MATLAB library command for solving CALE

Chapter 1

Introduction

1.1 Motivation

Nowadays mathematical modeling is a vital part of engineering interest e.g., control theory, system analysis, optimization, signal processing, large space flexible structures, game theory and design of physical systems. Multi-tasking systems having various components arise in many fields of engineering applications, such as microelectronics, micro-electro-mechanical systems, aerospace, computer control of industrial processes, chemical processes, communication systems, etc. are composed of branches of sub-systems and they are functioned by very large mathematical models utilizing the interrelated inner mathematical system of very large dimensions.

The Continuous Algebraic Riccati Equation (CARE) is strongly connected with the Linear Time-Invariant (LTI) system. CAREs appear in many areas of science and engineering, and in particular in control problems. The quadratic cost functional is an important part of modern mathematical models. The solution matrix of CARE used to optimize the Linear Quadratic Regulator (LQR) problem, which consists of an optimal control function that associated with the continuous LTI system, at which quadratic cost functional attains its infimum. Optimization of LQR in association with descriptor systems (unstable in particular) arise from engineering applications, there is still lacking efficient computational solvers or analytical tools.

In the conversion of the physical models into mathematical models, often their dimensions become extremely large and because of those models, analysis of the systems goes through the unsuitable approaches. The size of the matrices in CAREs is the most challenging aspect to store in computational tools. Because of large-scale matrix dimensions, simulation techniques require expensive time dealings and invade by the poor rate of convergence. Also, the accuracy of the solution reduces over time for very large and sophisticated continuous LTI systems. There are some Newton-based methods that exist, which are very complicated, time-consuming and preconditioned structures are required.

Though in the modern era of technology, faster and robust procedures for simulation, optimization and other engineering designs for systems large dimensions are available, computational complexity and wide range of memory requirements keep the techniques infeasible. Thus, the real-world models need to be replaced by some sort of lower-dimensional models. This alternative process of converting a large-scale model into a small-scale model is called Model Order Reduction (MOR), which has a wide variety of applications in engineering systems. The techniques of MOR are to approximate a large-dimensional model by a lower-dimensional Reduced-Order Model (ROM), while the system pattern is kept invariant to the largest possible extent. The reduction process does not require knowledge about the nature of the underlying systems and system properties such as stability and passivity are preserved. The algorithm must be robust and the global error bound, measured by some suitable norm must be minimized to a certain margin.

1.2 Literature Review

Depending on the characteristics of mathematical models, a number of techniques, for instance, Pade Approximation, Modal Truncation, Rational Interpolation, Optimal Hankel Norm Approximation, Singular Perturbation Approximation, Moment Matching Approximation and Balanced Truncation (BT) are commonly used as MOR approaches. Among those techniques, the theoretic BT method is one of the well-accepted methods for large sparse LTI systems [1, 2, 3, 4]. One of the great advantages of this projection-based BT method is that it does preserve the stability of the original systems. Most of the MOR techniques depend on the solution of the Continuous Algebraic Lyapunov Equation (CALE) as the projection generating tool. Moreover, it has a global error bound by choosing the prior error

tolerances. So, in the BT method, one can adapt the dimension of the ROMs. The key idea of this method is to delete the unnecessary states, which can be detected if the system is in balanced coordinates.

The implementation of Krylov subspace-based projection methods raising in the modern technologies for simulating the large-scale mathematical models arising from the complicated and automated physical systems [5]. The crucial role of these approaches to computing ROMs from the original models such that the pattern of the systems remains invariant [6]. The ROM is feasible in time-dealings and suitable for storing in the computing tools. Rational Krylov Subspace Method (RKSM) is one of the updated and efficient processes for solving matrix equations [7]. Iterative Rational Krylov Algorithm (IRKA) is an interpolatory projection method to find ROMs with invariant features such as stability, definiteness and transient behaviors as the original models and enhance the rate of computational convergence [8, 9].

Over the last decades, several iterative methods were proposed to solve large scale Riccati equation, e.g., earlier, the LQR approach applied that described an efficient way to find the optimal control for power systems. A brief discussion of the large-scale power system models subject to LTI systems have discussed and reduced-order transfer matrices from RLC network models of electric power grids subjecting the descriptor systems were analyzed by Freitas *et al.* [10, 11]. A number of MOR approaches are illustrated by Li in detail to achieve the ROMs for the large-scale LTI systems [12]. To solve large-scale CARE, Jbilou has applied the block Krylov subspace method and the work has been extended by block Arnoldi algorithm by Heyouni *et al.* [13, 14].

Fundamental discussion on the eligibility of RKSM for the large-scale LTI systems has provided by Simoncini and the application of adaptive RKSM to solve large-scale CARE for finding optimal control of the LTI systems narrated by Druskin *et al.* [15, 16]. Also, Simoncini introduced two newer approaches of RKSM by means of Newton iterates [17]. Analysis of the basic properties of RKSM for solving large-scale CAREs subject to LTI systems investigated by Simoncini, where the author briefed a new concept of shift parameters [18]. Very detailed discussion on the numerical solution of large-scale Riccati equations and LQR based optimal control problems are given in Benner *et al.*, where the author narrated the Newton methods and their extensions by means of Alternating Direction Implicit (ADI) method, the shift selection process, stopping criteria and comparison between the

ADI and RKSM techniques with supporting proofs [19]. Hylla discussed extensions of the inexact Kleinman-Newton method and their convergences for solving CARE with analogous theorems in [20]. MOR techniques of descriptor systems by interpolatory projection methods were introduced by Gugercin *et al.* [21]. The Matrix Sign function method for solving CAREs derived from descriptor systems was introduced by Huang *et al.*, which is helpful to find the LQR based optimal controls [22].

In the present decades, the works on optimal control based on descriptor systems flourished dramatically, for instance, Monir introduced a novel idea about the MOR technique for descriptor systems of higher indices using projected Cholesky factorization [23]. A balancing based MOR approach applying RKSM for the index-1 descriptor system was discussed by Monir *et al.*, and the work has been extended for second-order LTI systems by projecting onto the dominant Eigenspace by Hasan *et al.* [24, 25]. The ROM based feedback stabilization of the large-scale sparse power system models based on the descriptor systems was introduced by Hasan *et al.* [26]. Analysis of the shift parameters and their applicability were discussed by Benner *et al.*, the reformulation of in accordance with different types of shift parameters have been done with detailed numerical evidence [27].

Riccati based boundary feedback stabilization of incompressible Navier-Stokes flow was introduced by Bansch *et al.*, where a detailed analysis of mathematical model formulation, numerical conversion, ROM construction and the impact of non-dimensional parameters and the shift parameters were narrated [28]. The numerical comparison of solvers for large-scale CARE has been shown by Benner *et al.*, where several low-rank approximations of large-scale LTI systems were illustrated with numerical evidence for justifying the robustness, rate of convergence, memory allocation and pattern preservation of the ROMs in comparison to the original models [29]. Very recent work on the solution of large-scale CARE by means of projected Newton-Kleinman method has been published by Palitta, where extended forms of RKSM were introduced and extensions of Newton-Kleinman method utilizing extended RKSM were established for generalized CARE subject to LTI systems arise from engineering applications [30].

1.3 Objective

To overcome the difficulties in the conventional computations and achieve feasible outputs, projection-based techniques implementing the low-rank approximation approaches for the solution of large-scale CAREs are introduced. In this thesis, attention is mainly focused on finding Riccati based feedback stabilization for the systems subject to descriptor systems applying the projection-based RKSM. The RKSM approach is proposed to find ROMs for solving CAREs associated with very large sparse descriptor systems, using iterative techniques utilizing adaptive shifts. The computations allow the sparsity pattern and can be applied within closed-loop simulations. Also, a simplified version of the nested iterative Kleinman-Newton (KN) method combined with the Low-Rank Cholesky-Factor ADI (LRCF-ADI) technique is proposed to justify the effectiveness of RKSM technique.

By implementing the proposed techniques the low-rank approximation approaches for large-scale continuous LTI systems are achieved, where the unstable index-1 descriptor systems are stabilized through the Riccati based feedback stabilization process. The optimal feedback matrices are gained to stabilize the target systems. A detailed analysis of the solution of CAREs with the proposed techniques is assessed by numerical computation using the MATLAB simulations. Moreover, the efficiency of the proposed algorithm is investigated by applying them to the unstable descriptor systems governed by power system models.

1.4 Outlines of the Thesis

This thesis consists of 6 chapters including this introductory Chapter-1. In Chapter-2, the derivation and fundamental concepts of the systems and control theory are discussed. The basic ideas of linear algebra, matrix equations, and instabilities are thoroughly narrated. Some existing methods for solving matrix equations and real-world models are provided in brief. The terms and concepts of this chapter are used throughout the rest of the chapters.

Chapter-3 and Chapter-4 consist of the principal work of the thesis. In Chapter-3, RKSM techniques for solving CARE derived from the standard and generalized system are discussed. The conversion of the index-1 descriptor system into the

structured generalized system is described. RKSM approaches for structured generalized CARE including adjustment adaptive shift parameters, convergence criteria, treatment for the unstable systems and sparsity pattern of the techniques are investigated in this chapter. Chapter-4 consists of the derivation of LRCF-ADI based Kleinman-Newton approaches. LRCF-ADI algorithms for generalized CALE and its modified form including real version is explained. The convergence criteria and corresponding recurrence relations are derived for the generalized CALE. The conversion of CARE to CALE and settlement of the LRCF-ADI algorithms in the Kleinman-Newton approaches in the sparse form are described.

The numerical computation of the optimal feedback matrices for the models derived from unstable index-1 descriptor systems and corresponding Riccati based feedback stabilization techniques are analyzed in the Chapter-5. Both RKSM and KN-LRCF-ADI methods are applied and the effectiveness of them is justified by means of stabilization of eigenvalues and step-responses. The comparative analysis of the proposed methods is provided in this chapter with graphical explanations.

Finally, Chapter-6 contains the conclusions of the thesis. The possibilities for the improvements and future researches are highlighted in brief.

Chapter 2

Preliminaries

2.1 Systems and Control Theory

In control theory, the state-space representation is a mathematical model governed by a physical system as a set of input, output and state variables related by first-order differential equations or difference equations. State variables are regularized by time and the values of input variables, whereas output variables can be generated by the state variables. If the dynamical system is linear, time-invariant, and finite-dimensional, then the differential and algebraic equations can be formed as matrices [31]. The state-space method is distinguished by significant algebraization of general system theory, which makes it possible to use Kronecker vector-matrix structures. The capacity of these structures can be efficiently applied to research systems with modulation or without it. The details about the state-space systems and control problems are provided in [32].

2.1.1 State-Space Systems

The space-state representation of any physical system is an important part of the analysis of controllability, observability, and stability of the system. The structure of space-state representations indicates the pattern of the target systems.

Assume a state-space system that involves n integrators combined with p inputs $u_1(t), u_2(t), \cdots, u_p(t)$ and m outputs $y_1(t), y_2(t), \cdots, y_m(t)$. Define n outputs of

the integrators as state variables $x_1(t), x_2(t), \cdots, x_n(t)$ [33]. Then the system can be described as

$$\begin{aligned} e_1(t)\dot{x}_1(t) &= f_1(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t), \\ e_2(t)\dot{x}_2(t) &= f_2(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t), \\ \vdots \qquad & \qquad \vdots \qquad \qquad \vdots \\ e_n(t)\dot{x}_n(t) &= f_n(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t). \end{aligned} \tag{2.1}$$

The outputs of the system may be given as

$$\begin{aligned} y_1(t) &= g_1(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t), \\ y_2(t) &= g_2(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t), \\ \vdots \qquad & \qquad \vdots \qquad \qquad \vdots \\ y_m(t) &= g_m(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t). \end{aligned} \tag{2.2}$$

If we define the following matrices

$$E(t) = \begin{bmatrix} e_1(t) \\ e_2(t) \\ \vdots \\ e_n(t) \end{bmatrix}, x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix}, y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \end{bmatrix}, u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_p(t) \end{bmatrix},$$

$$f(x,u,t) = \begin{bmatrix} f_1(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t) \\ f_2(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t) \\ \vdots \qquad \vdots \\ f_n(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t) \end{bmatrix}, \tag{2.3}$$

$$g(x,u,t) = \begin{bmatrix} g_1(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t) \\ g_2(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t) \\ \vdots \qquad \vdots \\ g_m(x_1, x_2, \cdots, x_n; u_1, u_2, \cdots, u_p; t) \end{bmatrix}.$$

Then equations (2.1) and (2.2) become a pair of equations as

$$\begin{aligned} E(t)\dot{x}(t) &= f(x,u,t), \\ y(t) &= g(x,u,t), \end{aligned} \tag{2.4}$$

where the equations in (2.4) represent state equation and output equation, respectively. If the functions f and g do not involve the time t explicitly, the system is called a time-invariant system. If the equations in (2.4) are linearized about the operating state, then the following linear time-invariant state-space system with input-output equations

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2.5}$$

where $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{m\times n}$ and $D \in \mathbb{R}^{m\times p}$ with very large[1] n and $p, m \ll n$; represent mass matrix, system matrix, control multiplier matrix, state multiplier matrix and direct transmission map (gain), respectively [34]. In the system (2.5), $x(t) : \mathbb{R} \mapsto \mathbb{R}^n$ and $u(t) : \mathbb{R} \mapsto \mathbb{R}^p$ are the state vector and control (input) vector, while $y(t) : \mathbb{R} \mapsto \mathbb{R}^m$ is the output vector, consider $x(t_0) = x_0$ as the initial state. In most of the state-space systems the direct transmission remain absent and because of that $D = O$.

The dimension of the system (2.5) is determined by the dimension of the state vector $x(t)$ is n. For $p = m = 1$, the system is identified as Single-Input Single-Output (SISO) system and Multi-Input Multi-Output (MIMO) system for $p, m > 1$. The system (2.5) is the arrangement of several differential and algebraic equations but for the convenience of further manipulation, the system will be treated as a compact pair of input and output equations.

2.1.1.1 Generalized and Standard System

The system (2.5) is said to be generalized for invertible and symmetric positive definite matrix E. If $E = I_n$, is the n-dimensional identity matrix, the system (2.5) is classified as standard. Again, since E is invertible, the generalized system can be converted to the standard system of the following form

$$\begin{aligned} \dot{x}(t) &= \bar{A}x(t) + \bar{B}u(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2.6}$$

where $\bar{A} = E^{-1}A$ and $\bar{B} = E^{-1}B$. Because of time consuming inversion process, the conversion (2.6) in not suitable for real-time practice.

[1] The meaning of large is changing over time due to the increasing capability of the simulation tools.

2.1.1.2 Descriptor System

In the present control theory, a special form of generalized space-state systems with singular matrix E, i.e., $\det(E) = 0$, arising from many physical models, are called descriptor systems. The descriptor systems are also known as the singular LTI systems or Differential-Algebraic Equations (DAE) [35]. Such systems are governed in the modeling of power systems, chemical engineering, and mechanical systems. A particular descriptor system is solvable if the corresponding matrix pencil is regular, i.e., $\det(\lambda E - A) \neq 0$. According to matrix algebra for the regular matrix pencil, there exist non-singular (invertible) transformation matrices T_L and T_R such that the matrices E and A have the Weierstrass canonical form as follows

$$E = T_L \begin{bmatrix} I_{n_f} & O \\ O & N \end{bmatrix} T_R \quad \text{and} \quad A = T_L \begin{bmatrix} J_1 & O \\ O & I_{n_\infty} \end{bmatrix} T_R, \tag{2.7}$$

where N is nil-potent with nil-potency v, i.e., $N^{v-1} \neq O$ but $N^v = O$, and $n_f + n_\infty = n$. The number v is referred as the algebraic index. The details of the descriptor systems and their derivation is narrated in [36].

This thesis is concerned with the special structured descriptor systems considering their applications in the fields of engineering. The descriptor systems that we focus on the block matrix form can be structured as

$$\begin{aligned} \underbrace{\begin{bmatrix} E_1 & E_2 \\ O & O \end{bmatrix}}_{E} \underbrace{\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix}}_{\dot{x}(t)} &= \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{B} u(t), \\ y(t) &= \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{C} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + D_a u(t), \end{aligned} \tag{2.8}$$

where $x_1(t) \in \mathbb{R}^{n_1}$, $x_2(t) \in \mathbb{R}^{n_2}$ with $n_1 + n_2 = n$ are state vectors and J_1, J_2, J_3 & J_4 are block matrices of A with appropriate dimensions. Here, E_1 and J_1 have full rank. If the block matrix $E_2 \neq O$, the system (2.8) is called semi-explicit descriptor system. In some particular cases, $E_2 = O$ needs to be assumed in the descriptor system (2.8).

According to physical attributes and transient behaviors, the corresponding mathematical models can be formed in various patterns. The descriptor systems (2.8) can be classified as

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

2.1.2 Input-Output Relations

The step response and the frequency response are the two most common inputs in the time domain analysis. By applying the Laplace transformation[2], the state-space system (2.5) can be expressed in the frequency domain. Then for the complex variable s the system gets the form as

$$\begin{aligned} sEX(s) - x_0 &= AX(s) + BU(s), \\ Y(s) &= CX(s) + DU(s). \end{aligned} \tag{2.9}$$

where $X(s), U(s)$ and $Y(s)$ are the Laplace transformations of $x(t), u(t)$ and $y(t)$, respectively. For $x_0 = O$, the system (2.9) can be written as

$$\begin{aligned} X(s) &= (sE - A)^{-1} BU(s), \\ Y(s) &= G(s)U(s). \end{aligned} \tag{2.10}$$

In SISO systems, the function $G(s)$ can be defined as

$$G(s) = C(sE - A)^{-1} B + D. \tag{2.11}$$

In MIMO systems, $G(s)$ is the $p \times m$ matrix, can be written as

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1m}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2m}(s) \\ \vdots & \vdots & \cdots & \vdots \\ G_{p1}(s) & G_{p2}(s) & \cdots & G_{pm}(s) \end{bmatrix} \tag{2.12}$$

where $G_{ij} = C(i,:)(sE - A)B(:,j) + D(i,j)$ with the indices $i = 1, 2, \cdots, p$ and $j = 1, 2, \cdots, m$.

[2]The Laplace transformation of the function $f(t)$ is defined as for all $t \geq 0 \in \mathbb{R}$, is the function $F(s) = \mathcal{L}[f(t)] = \int_0^\infty f(t)e^{-st}dt$ for the number $s \in \mathbb{C}$.

2.1.2.1 Transfer Function

The function $G(s)$ introduced in (2.10) and defined in (2.11) and (2.12) are called the transfer functions of the system (2.5) for SISO and MIMO structures, respectively. The transfer function indicates the input-output relations of the state-space systems. In the control theory, the error bounds of the reduced-order models are established by means of their transfer functions [37].

The transfer function $G(s)$ is called proper if $\lim_{x\to\infty} G(s) < \infty$ and called strictly proper if $\lim_{x\to\infty} G(s) = 0$, otherwise $G(s)$ is called improper. The point s_p at which $G(s_p) \to \infty$ is called the pole of the system.

2.1.2.2 Impulse Response and Frequency Response

The relationship between input and output is known as the impulse response and denoted by $h(t)$, which is the transfer function of the system (2.5) in the time domain. The impulse response corresponding to (2.5) can be defined as

$$h(t) = y(t)u(t)^{-1}; \quad t \in \mathbb{R}. \tag{2.13}$$

If the input and the impulse response of a system are available, the system output can be estimated by the following convolution[3] operation

$$y(t) = h(t) * u(t). \tag{2.14}$$

Again, the frequency response of a system is as the same as the transfer function but the input-output relations of the system in the complex Fourier domain, i.e., $s = j\omega;\ \omega \in \mathbb{R}$. Thus, the frequency response of a system can be found from the transfer function (2.11), by setting $s = j\omega;\ \omega \in \mathbb{R}$.

The frequency response of the state-space system (2.5) can be defined as

$$G(j\omega) = C(j\omega E - A)^{-1}B + D. \tag{2.15}$$

where $\omega \in \mathbb{R}$ is the frequency of the system that provides the value of transfer function on the imaginary axis.

[3]The convolution of two function, $f_1(t)$ and $f_2(t)$ is $(f_1 * f_2)(t) = \int_{-\infty}^{\infty} f_1(\tau) f_2(t-\tau) d\tau$

2.1.3 Reduced Order Model

The large-scale space-state systems are governed from very large dimensional real-world engineering models with sophisticated ingredients and have very complex arrangements. In this situation, the dimensions of the differential coefficient and system matrices become very high. Simulation techniques for these systems require highly expensive time dealings and invade by the infeasible rate of convergence [38].

The size of the matrices in large-scale state-space systems is the most challenging aspect to store in computational tools. Though the faster technologies and robust simulation techniques for large dimensional systems are available, computational complexity and a wide range of memory requirements keep the computations infeasible.

So, the large-scale real-world models need to be converted into the Reduced-Order Models (ROM) through the iterative techniques, e.g., ADI, RKSM and IRKA [39], which has a wide variety of applications in engineering systems.

The ROM corresponding to the system (2.5) can be derived as

$$\begin{aligned} \hat{E}\dot{\hat{x}}(t) &= \hat{A}\hat{x}(t) + \hat{B}u(t), \\ \hat{y}(t) &= \hat{C}\hat{x}(t) + \hat{D}u(t), \end{aligned} \tag{2.16}$$

where the reduced order matrices can be obtained by proper transformation provided by the simulation techniques.

In the techniques, system pattern is kept invariant to the largest possible extent and size of the ROMs should be allocable in the sense of memory and time dealings [40]. The algorithm must be robust and have the global error bound, measured by suitable norm must be minimized to a certain margin. Also, the transfer functions of (2.5) and (2.16) must be same.

2.2 Matrix Equations

In this section, some linear and quadratic matrix equations will be introduced, those have important applications in control theory. Also, the concept of the linear quadratic regulator problem will be discussed in short.

2.2.1 Lyapunov Equation

In the MOR approaches, Lyapunov equations are the key tools for the large-scale state-space systems. The Continuous Algebraic Lyapunov Equations (CALE) are the essential parts of controllability and observability analysis [41]. The generalized CALEs can be structured in the following ways

$$APE^T + EPA^T + BB^T = O. \tag{2.17}$$

$$A^TQE + E^TQA + C^TC = O. \tag{2.18}$$

where P and Q are the controllability and observability Gramians, respectively, and can be defined as

$$P = \int_0^\infty e^{At}BB^Te^{A^Tt}dt \tag{2.19}$$

$$Q = \int_0^\infty e^{A^Tt}C^TCe^{At}dt \tag{2.20}$$

2.2.2 Riccati Equation

In the present advancement of engineering fields, such as in control theory, the continuous LTI system is a vital part of interest. The Continuous Algebraic Riccati Equation (CARE) has a wide range of applications in the control problems [42]. Riccati based feedback matrix has a pivotal role in the state-space stabilization techniques. The generalized CARE is defined as

$$A^TXE + E^TXA - E^TXBR^{-1}B^TXE + C^TC = O. \tag{2.21}$$

Here E, A, B, and C, are defined in (2.5), whereas the control cost matrix[4] $R \in \mathbb{R}^{p\times p}$ is symmetric positive definite.

For the standard state-space system the CARE can be defined as

$$A^TX + XA - XBR^{-1}B^TX + C^TC = O. \tag{2.22}$$

[4]For the power system models, the cost matrix R is considered as the identity matrix.

2.2.3 Bernoulli Equation

To find a special form of the CARE (2.21), consider the term C^TC is not present, i.e., $C^TC = O$. This form is called the Continuous Algebraic Bernoulli Equation (CABE) [43]. The generalized CABE is defined as

$$A^TXE + E^TXA - E^TXBR^{-1}B^TXE = O. \tag{2.23}$$

The Bernoulli equation has various applications of stability analysis in the control theory [44]. The trivial solution $X = O$ is always a solution of the CABE (2.23), whereas this solution is not a matter of interest.

2.2.4 Linear Quadratic Regulator Problem

Linear Quadratic Regulator (LQR) problem is an essential tool of the control problems [45]. The setup of a self-generated regulator or controller governing from an engineering application, especially in mechanical and electrical systems, is utilized by applying a mathematical algorithm to minimize a cost functional and controls run-time perturbations by means of appropriate constraints.

The LQR algorithm minimizes the work-load of the control systems to optimize the controller. The system needs to be initiated with raw parameters and test the efficiency of the control system for the desired aim. The execution process is iterative and needs to find an optimal control through simulation and then rearrange the parameters to achieve a feasible output of the objective system. The quadratic cost functional is defined as

$$J(u, x_0) = \int_0^\infty (x^T(t)C^TCx(t) + u^T(t)Ru(t))dt. \tag{2.24}$$

The function (2.24) can be optimized as $J(u^0, x_0) = {x_0}^T X x_0$ by applying an optimal control $u^o = -K^o x(t)$ generated by the optimal feedback matrix $K^o = R^{-1}B^TXE$ associated with the solution matrix X of the CARE (2.21).

2.3 Stability and Related Topics

System stability is one of the most important performance specifications of a control system. Some of the basic concepts of the stability of a system are discussed below.

2.3.1 Stable and Unstable System

The stability of a system relates to its response to inputs or disturbances. A system that remains in a constant state unless affected by an external action and which returns to a constant state when the external action is removed can be considered to be stable.

A system is said to be stable if its output is under control. Otherwise, it is said to be unstable. A stable system produces a bounded output for a given bounded input. In control theory, stability is defined as a measure of the tendency of a system's response return to zero after being disturbed.

The matrix pair (A, E) is said to be Hurwitz-stable if all of its eigenvalues are lie in the open left half of the complex plane, i.e., $\lambda \in \mathbb{C}^-$ and the system is unstable if any eigenvalues of matrix pair (A, E) lie in the open right half of the complex plane, i.e., $\lambda \in \mathbb{C}^+$. The Hamiltonian matrix plays an important role in determining the attributes of system stability.

Moreover, if a few eigenvalues of the matrix pair (A, E) lie in the open right half of the complex plane but very close to the imaginary axis and rest of the eigenvalues lie in the open left half of the complex plane, the matrix pair (A, E) is said to be semi-stable [46].

2.3.2 Hamiltonian Matrix

Hamiltonian matrices corresponding to the CARE (2.21) and CABE (2.23) are defined as

$$H_R = \begin{bmatrix} AE^{-1} & -BR^{-1}B^T \\ -E^{-T}C^TCE^{-1} & -E^{-T}A^T \end{bmatrix} \tag{2.25}$$

$$H_B = \begin{bmatrix} AE^{-1} & -BR^{-1}B^T \\ O & -E^{-T}A^T \end{bmatrix} \tag{2.26}$$

where E is a non-singular matrix. For the standard state-space systems $E = I_n$, i.e., identity matrix.

If, the Hamiltonian matrix (2.25) has no pure imaginary eigenvalues, the matrix triple $(A, B; E)$ is called stabilizable, the matrix $A-(BR^{-1}B^T)XE$ is called stable and the CARE (2.21) has a unique solution X [47]. In contrast, if some eigenvalues of the Hamiltonian matrix (2.25) lie on the imaginary axis, the CARE (2.21) has no unique or finite solution. As like as CARE, the stabilizing solution X of the CABE (2.23) provides $A - (BR^{-1}B^T)XE$ is stable.

Detectability is the dual concept of the stabilizability. The matrix triple $(C, A; E)$ is said to be detectable if the matrix triple $(A^T, C^T; E)$ is stabilizable.

The following theorems depict the criterion of the stabilizability and detectability in brief [48].

Theorem 2.1 (**Stabilizability Characterization**). *The following statements are equivalent,*

- $(A, B; E)$ *is stabilizable,*
- $Rank\Big[\lambda E - A, B\Big] = n$ *for all* $Re(\lambda) \geq 0$,
- *For all* λ *and* $x \neq O$ *such that* $x^*A = \lambda x^*E$ *and* $Re(\lambda) \geq 0$, *provided* $x^*B \neq O$.

Theorem 2.2 (**Detectability Characterization**). *The following statements are equivalent,*

- $(C, A; E)$ *is detectable,*
- *The matrix* $\begin{bmatrix} A - \lambda E \\ C \end{bmatrix}$ *has full column rank for all* $Re(\lambda) \geq 0$,
- *For all* λ *and* $x \neq O$ *such that* $Ax = \lambda Ex$ *and* $Re(\lambda) \geq 0$, *provided* $Cx \neq O$,
- $(A^T, C^T; E)$ *is stabilizable.*

The following theorem illustrates the characteristics of the system stability according to the Hamiltonian matrix [49].

Theorem 2.3 (**Unique Stabilizing Solution of CARE**). *Suppose* $(A, B; E)$ *is stabilizable and* $(C, A; E)$ *is detectable, assume* $R > O$. *Then there exists a unique positive semi-definite stabilizing solution* X *of the CARE (2.5). This solution is given by* $X = X_2 X_1^{-1}$, *where the columns of the matrix* $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$ *span the invariant subspace of the Hamiltonian matrix (2.25) associated with its stable eigenvalues.*

2.3.3 Feedback Stabilization

The control system in which the output has an effect on the input quantity in such a manner that the input quantity will adjust itself based on the output generated is called a closed-loop control system. Feedback is a common and powerful tool when designing a control system. The feedback stabilization is a concept that consisted of a feedback element that gives information on the present state of the system and then adjustments to the system's present operation. In any control system, the output is affected due to the change in environmental conditions or any kind of disturbance. So the feedback element constantly receives updates from the output and is return-back to the input [50].

The fundamental point remains, however, that it is the power of feedback to combat uncertainty which makes it so useful for the purposes of control.

2.3.3.1 Bernoulli Stabilization

If the matrix pencil (A, E) is not stable in general but to solve the CARE (2.21), a stable pencil is required. Then an initial feedback matrix K_0 is needs to be attained in the way provided in [51]. Most of the $n_v - n_p$ finite eigenvalues of the matrix pencil (A, E) are stable and only n_{us} eigenvalues are unstable with $n_{us} \ll n_v$. Initially, all unstable finite eigenvalues $\lambda_{us}^{(i)} \in \mathbb{C}^+$ of the matrix pencil (A, E) are required together with their corresponding left and right eigenvalues $\omega^{(i)}, \eta^{(i)} \in \mathbb{C}^{n_v+n_p}$ for $i = 1, 2, ..., n_{us}$.

To solve large-scale generalized eigenvalue problems, implicitly restarted shift-and-invert Arnoldi method can be implemented in the `eigs` command of MATLAB with matrix shifting approach. In this technique, all infinite eigenvalues will be transferred to fixed finite eigenvalues, whereas other eigenvalues will remain unchanged.

By means of the left and right eigenvectors, the following projection matrices can be defined as

$$\begin{aligned} W &:= \left[\omega^{(1)}, \omega^{(2)}, \cdots, \omega^{n_{us}}\right] \in \mathbb{C}^{(n_v+n_p)\times n_{us}}, \\ V &:= \left[\eta^{(1)}, \eta^{(2)}, \cdots, \eta^{n_{us}}\right] \in \mathbb{C}^{(n_v+n_p)\times n_{us}}. \end{aligned} \tag{2.27}$$

The using the matrices W and V, the n_{us}-dimensional generalized CABE can be written as

$$\hat{A}^T X_0 \hat{E} + \hat{E}^T X_0 \hat{A} - \hat{E}^T X \hat{B} R^{-1} \hat{B}^T X_0 \hat{E} = O, \tag{2.28}$$

where $\hat{E} = W^T E V, \hat{A} = W^T A V$ and $\hat{B} = W^T B$. After solving (2.28) for X_0, the initial feedback matrix $K_0 = B^T (W X_0 W^T) E \in \mathbb{R}^{n_r \times n_v}$ can be estimated.

Using the initial feedback matrix K_0, the desired stabilized closed-loop matrix pencil can be written as $(A - BK_0, E)$, and all the initially unstable eigenvalues $\lambda_{us}^{(i)}$ converted (mirrored image) to the stabilized eigenvalues as

$$\lambda_{stab}^{(i)} = -\mathrm{Re}(\lambda_{us}^{(i)}) + j\mathrm{Im}(\lambda_{us}^{(i)}) \in \mathbb{C}^-; \quad \forall i = 1, 2, \cdots, n_{us}. \tag{2.29}$$

2.3.3.2 Riccati Stabilization

Riccati-based feedback stabilization is the most rising approach for the stabilization in large-scale system simulations. The convergence of computationally gained feedback matrix depends on the actual model [52]. In recent researches, the stabilization of the unstable systems around a stationary solution using a Riccati-based feedback matrix has achieved significant attention regarding control theory as well as numerical methods [53, 54].

The difficulty in the LQR approach for the target model under investigation is to compute the feedback matrix K_f, such that the stabilized system has some specific forms. In the Riccati-based feedback stabilization technique, to solve the CARE (2.21) arising from the large-scale model is the most challenging task. So, the reduced-order model (2.16) will be employed to compute an approximation to the optimal feedback matrix of the full system by means of the LQR approach.

Based on the ROM (2.16) the generalized CARE can be written in the form

$$\hat{A}^T \hat{X} \hat{E} + \hat{E}^T \hat{X} \hat{A} - \hat{E}^T \hat{X} \hat{B} R^{-1} \hat{B}^T \hat{X} \hat{E} + \hat{C}^T \hat{C} = O, \tag{2.30}$$

The generalized CARE (2.30) is feasible in the sense of matrix dimensions, that can be solved efficiently for $\hat{X}$ using any conventional solvers as, e.g., the MATLAB `care` command.

Then, the stabilizing feedback matrix for the ROM (2.16) can be computed as $\hat{K}_f = \hat{B}^T \hat{X} \hat{E}$. The ROM based approximation to the stabilizing feedback matrix for the full order model can now be retrieved as $K_f = \hat{K}_f V^T E$, where V is the transformation matrix used to compute the ROM (2.16).

2.4 Background of Linear Algebra

To understand the theoretical concept of system and control theory, basic ideas of the linear algebra are essential. In this section we will discuss some fundamentals of the linear algebra.

2.4.1 Eigenvalue Problem

For the matrix pair (A, E), where $A, E \in C^{n\times n}$, an eigenvalue $\lambda \in \mathbb{C}$ and its right eigenvector $x \in \mathbb{C}^n \setminus \{0\}$ and the left eigenvector $y \in \mathbb{C}^n \setminus \{0\}$ together form an eigen-triple (λ, x, y) of the matrix pair (A, E), which satisfies the generalized Eigenvalue Problem (EVP) is defined as

$$Ax = \lambda Ex, \quad y^*A = \lambda y^*E. \tag{2.31}$$

The eigenvalues are the roots of the characteristic polynomial $p(\lambda) = \det(A - \lambda E)$ and the spectrum is the set of all eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ corresponding to the matrix pair (A, E), denoted by $\Lambda(A, E)$ [55]. If E is singular, $\Lambda(A, E)$ contains eigenvalues at infinity and the finite spectrum $\Lambda_f(A, E)$ denotes the set of all finite eigenvalues of the matrix pair (A, E).

A matrix pair (A, E) is called singular if $A - \lambda E$ is singular for all $\lambda \in \mathbb{C}$, otherwise it called regular.

Theorem 2.4 (**Eigenvalue Criteria**). *Let the matrix pair (A, E) with $A, E \in C^{n\times n}$ and $\lambda \in \mathbb{C}$. Then the following statements are equivalent,*

- *λ is an eigenvalue of (A, E) if and only if $\frac{1}{\lambda}$ is an eigenvalue of (E, A),*
- *∞ is an eigenvalue of (A, E) if and only if E is singular matrix,*
- *∞ is an eigenvalue of (A, E) if and only if 0 is an eigenvalue of (E, A),*
- *If E is non-singular, the eigenvalues of (A, E) are exactly the eigenvalues of AE^{-1} and $E^{-1}A$.*

The algebraic multiplicity $\alpha(\lambda)$ of a particular eigenvalue is the number of times λ appears as the root of $p(\lambda)$. The number of linearly independent right and left eigenvectors x, y associated to λ is called the geometric multiplicity and denoted by $\zeta(\lambda)$, which satisfies $1 \leq \zeta(\lambda) := \dim\{\ker(A - \lambda E)\} \leq \alpha(\lambda)$. If $\zeta(\lambda) = \alpha(\lambda)$ then λ is called the simple eigenvalue and the corresponding matrix pair (A, E) is called diagonalizable.

The following lemmas illustrate the properties of the diagonalizable matrix pair (A, E) [56].

Lemma 2.5. *A matrix pair (A, E) with $A, E \in C^{n\times n}$ is diagonalizable if and only if there exists a non-singular matrix $X \in \mathbb{C}^{n\times n}$ and $\lambda_1, \lambda_2, \cdots, \lambda_n$ are the eigenvalues of the matrix pair (A, E) such that $X^{-1}AX = diag(\lambda_1, \lambda_2, \cdots, \lambda_n)$, where the columns of X are eigenvectors of the matrix pair (A, E).*

Lemma 2.6. *Let the matrix pair (A, E) with $A, E \in C^{n\times n}$ be diagonalizable having distinct eigenvalues $\Lambda(A, E) = \{\lambda_1, \lambda_2, \cdots, \lambda_{\hat{n}}\}$ with $n \leq \hat{n}$. Then for the all $i = 1, 2, \cdots, \hat{n}$, the relation $\zeta(\lambda_i) = \alpha(\lambda_i)$ holds.*

A matrix pair (A, E), where $A, E \in C^{n\times n}$ with non-singular E, is called normal matrix pair if it is diagonalizable and its left eigenvectors coincide with the right eigenvectors. The following theorem depicts the properties of the normal matrix pair [57].

Theorem 2.7 (Normal Matrix Pair). *Let the matrix pair (A, E) with $A, E \in C^{n\times n}$ be the regular matrices with non-singular E. Then the following statements are true*

- *(A, E) is a normal matrix pair,*
- *There exists $Q \in \mathbb{C}^{n\times n}$ such that $Q^T AQ = diag(\lambda_1, \lambda_2, \cdots, \lambda_n)$ and satisfies $Q^T EQ = I_n$.*

If $E = I_n$ eigenvalue problem can be re-defined for the matrix $A \in \mathbb{C}^{n\times n}$ as

$$Ax = \lambda x, \quad y^*A = \lambda y^*. \tag{2.32}$$

Here, the characteristic polynomial can be defined as $p(\lambda) = \det(A - \lambda I_n)$ and the corresponding spectrum can be denoted by $\Lambda(A)$. The other properties will remain the same as the generalized EVP.

2.4.2 Matrix Definiteness

A symmetric matrix $M \in \mathbb{R}^{n\times n}$ is said to be positive definite if the scalar $z^T Mz$ is strictly positive for every non-zero column vector z of n real numbers. When interpreting Mz as the output of an operator M, z is acting on an input. The property of positive definiteness implies that the output always has a positive inner product with the input, as often observed in physical processes [58].

Again, a Hermitian matrix $M \in \mathbb{C}^{n\times n}$ is said to be positive definite if the scalar z^*Mz is strictly positive for every non-zero column vector z of n complex numbers, where z^* denotes the conjugate transpose of z.

Positive semi-definite matrices are defined similarly, except that the above scalars $z^T Mz$ or z^*Mz must be non-negative. Negative definite and negative semi-definite matrices are defined analogously. A matrix that is not positive semi-definite and not negative semi-definite is called indefinite.

2.4.3 Structure of the Matrices

The structure of the matrices has an important impact on system adaptation and computational convergence. Different types if matrix structure can be generated by the governing models. To apply convenient simulation techniques the matrices of a target system can be converted to the user-defined structure [59].

2.4.3.1 Sparse Matrix

A matrix is sparse if many of its coefficients are zero and there is no memory allocation for those coefficients. The interest in sparsity arises because its exploitation can lead to enormous computational savings and because many large matrix problems that occur in engineering applications are sparse.

In practice, most large matrices are sparse, i.e., almost all entries are zeros. The number of zero values in the matrix divided by the total number of elements in the matrix is called the sparsity of a matrix, which is useful in combinations and application areas such as network theory.

Large sparse matrices are common in general and especially in applied machine learning, such as in data that contains counts, data encoding that map categories to counts, and even in whole sub-fields of machine learning such as natural language processing. The sparse matrices are feasible for memory allocation and computation speed on the computer simulations. It is beneficial and often necessary to use specialized algorithms and data structures that take advantage of the sparse structure of the matrices.

2.4.3.2 Dense Matrix

A matrix is said to be dense if it is not sparse. In other words, if most of the elements are nonzero, then the matrix is considered as dense. In the dense matrices, a large number of elements are zero values and this is a waste of memory resources as those zero values do not contain any information. Some very large matrix systems are infeasible to manipulate using standard dense matrix algorithms.

2.4.3.3 Projection Matrix

The projection on a vector space V is a linear operator $P : V \mapsto V$ such that $P^2 = P$. Then the square matrix P is called a projection matrix. The projection matrix P is called an orthogonal projection matrix if $P^2 = P = P^T$ for a real matrix, and $P^2 = P = P^*$ for a complex matrix, where P^* denotes the Hermitian transpose of P. By definition, a projection matrix P is idempotent and corresponding eigenvalues must be 0 or 1.

The following lemma represents the properties of an orthogonal projector [60].

Lemma 2.8. *A projector projects onto a subspace S_1 along a subspace S_2, it said to be orthogonal projector if and only if, $S_1, S_2 \in \mathbb{C}^n$ are orthogonal sub-spaces such that $S_1 \cap S_2 = \{0\}$ and $S_1 + S_2 = C^n$, where $S_1 + S_2$ denotes the span of S_1 and S_2, i.e., the set of vectors $s_1 + s_2$ with $s_1 \in S_1$ and $s_2 \in S_2$, respectively.*

If P is a projector, $I - P$ is also a projector, called the complementary projector that satisfies $(I - P)^2 = I - P$.

2.4.4 Matrix Decomposition Techniques

In the control theory, matrix decomposition (factorization) is a computing tool for deriving ROMs of the large-scale systems. There are several matrix decomposition approaches and in this section, some of the leading techniques will be discussed.

2.4.4.1 Eigenvalue Decomposition

Eigenvalue decomposition is the factorization of a matrix into a canonical form, whereas the matrix is expressed by means of eigenvalues and eigenvectors. eigenvalue decomposition also called spectral decomposition [61].

If the columns $V \in \mathbb{C}^{n\times n}$ contain linearly independent eigenvectors of a square matrix $A \in \mathbb{C}^{n\times n}$, the eigenvalue decomposition of A can be defined as

$$A = V\Lambda V^{-1}, \tag{2.33}$$

where $\Lambda \in \mathbb{C}^{n\times n}$ is a diagonal matrix whose elements are the eigenvalues of A.

2.4.4.2 Singular-Value Decomposition

Singular-Value Decomposition (SVD) is one of the most useful matrix decomposition tools applied in the control systems, signal processing, and statistics. It can be efficiently used to generate ROMs. It is the generalization of the eigenvalue decomposition of a positive semi-definite normal matrix to any $m \times n$ matrix via an extension of the polar decomposition [62].

For a matrix $A \in \mathbb{C}^{m\times n};\ m, n \in \mathbb{R}$, the SVD of A as the matrix factorization can be defined as

$$A = U\Sigma V^*, \tag{2.34}$$

where $U \in \mathbb{C}^{m\times m}$ and $V \in \mathbb{C}^{n\times n}$ are unitary, and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix. The diagonal elements $\sigma_j; j = 1, 2, \cdots, k$ of Σ are the singular values of A, which are non-negative and in decreasing order, i.e., $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0$, where $k = \min(m, n)$.

The SVD obtained by taking only the first m singular values of A is called the thin SVD.

The following theorem represents the properties of the SVD [60].

Theorem 2.9 (**Properties of SVD**). *For the singular-value decomposition of a matrix A, the following statements are true,*

- *The singular-values σ_j of A are the square roots of the eigenvalues of the symmetric positive semi-definite matrix A^TA,*
- *The right singular-vectors are the eigenvectors of the matrix A^TA, and the left singular-vectors are the eigenvectors of the matrix AA^T,*
- *The rank of A is r, the number of non-zero singular-values and A is the sum of rank-one matrices,*
- *If $A = A^*$, then the singular-values of A are the absolute values of the eigenvalues of A,*
- *For $A \in \mathbb{C}^{n\times n}$, $det(A) = \prod_{j=1}^{n} \sigma_j$,*
- $\|A\|_2 = \sigma_1$ *and* $\|A\|_F = \sqrt{\sum_{k=1}^{r} \sigma_k^2}$.

2.4.4.3 Schur Decomposition

Schur decomposition is the matrix decomposition that allows writing any arbitrary matrix as unitarily equivalent to an upper triangular matrix whose diagonal elements are the eigenvalues of the original matrix [63]. It also called the Schur triangulation.

For a square matrix $A \in \mathbb{C}^{n\times n}$, the Schur decomposition can be defined as

$$A = UTU^*, \tag{2.35}$$

where $U \in \mathbb{C}^{n\times n}$ is a unitary matrix and T is an upper triangular matrix, which is called a Schur form of A. Since T is triangular and similar to A, it has the same spectrum and eigenvalues of A are the diagonal entries of T.

2.4.4.4 QR Decomposition

QR decomposition is a tool to find an orthogonal matrix with respect to some given matrices. QR decomposition is often used to solve the linear least square problems and is the basis for a particular eigenvalue algorithm [64].

QR decomposition is a factorization of a matrix $A \in \mathbb{C}^{m\times n}$ into a matrix product defined as

$$A = QR \tag{2.36}$$

where Q is an orthogonal matrix, i.e., $QQ^T = I = Q^TQ$ and R is an upper triangular matrix. For an invertible matrix A the factorization is unique and the diagonal elements of R are positive definite.

There are several approaches for computing QR decomposition, such as the modified Gram-Schmidt process and Householder transformations.

2.4.4.5 Cholesky Decomposition

Cholesky decomposition is the factorization of a Hermitian, positive-definite matrix into the product of a lower triangular matrix and its conjugate transpose, which is useful for several efficient numerical computations, for example, Monte-Carlo simulations [65].

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

$$A = LL^* \tag{2.37}$$

where L is a lower triangular matrix with real and positive diagonal entries and L^* is its conjugate transpose. Every Hermitian positive-definite matrix has a unique Cholesky decomposition.

Cholesky decomposition can be used to solve the system of linear equations $Ax = b$, where A is the real symmetric and positive-definite matrix.

2.4.4.6 Arnoldi Decomposition

Arnoldi decomposition is an efficient iterative solver and a vital tool for generating the basis for the Krylov subspace. Typically, it is a large sparse matrix algorithm that does not execute the matrix elements directly, instead utilizes the matrix map vectors. To compute the eigenvalues of large sparse matrices, the Krylov-based Arnoldi process is one of the most powerful tools.

Choose $A \in \mathbb{R}^{n\times n}$ and an orthogonal projector $V \in \mathbb{R}^{n\times p}$. Then the m-th dimensional Krylov matrix based on A and V is defined as

$$\mathcal{K}_m(A, V) = \left[V, AV, A^2V, \cdots, A^{m-1}V\right]. \tag{2.38}$$

Matrix-vector products play the key role in generating the Krylov subspace (2.38) by a recursive technique. The orthogonal columns of the matrix $V_{m+1} = \left[V_m, v_{m+1}\right]$ form an orthogonal basis for the Krylov subspace $\mathcal{K}_m$.

There exists an unreduced upper Hesenberg matrix $\hat{H}_m \in \mathbb{R}^{(m+1)\times m}$ such that $AV_m = V_{m+1}\hat{H}_m$ [66]. By a suitable partition of $\hat{H}_m$, we can write

$$\begin{aligned} AV_m &= \begin{bmatrix} V_m & v_{m+1} \end{bmatrix} \begin{bmatrix} H_m \\ h_{m+1,m}e_m^T \end{bmatrix}, \\ &= V_m H_m + h_{m+1,m} v_{m+1} e_m^T. \end{aligned} \tag{2.39}$$

Here, H_m can be obtained from $\hat{H}_m$ by removing the last row and e_m in the matrix of the last p columns of the mp-th order identity matrix $I_{m\times p}$ and after m steps $h_{m+1,m}$ will be vanished. So that, after a certain number of iterations the second term of (2.39) will be converged to zero.

Thus, by the orthogonality property of v_{m+1}, (2.39) provides the projection

$$H_m = V_m^T A V_m. \tag{2.40}$$

Algorithm 1: Arnoldi decomposition (Modified Gram-Schimdt).

Input : A, C, orthogonal matrix V_m.
Output: Matrix $Z_m \in \mathbb{R}^{n \times m}$ such that $X_m \approx Z_m Z_m^T$.

Compute $C^T = QR$ (QR factorization).
Assume $V_1 = Q = v_1$.
for $j \leftarrow 1$ *to* m **do**
    Compute $w_j = Av_j$.
    for $i \leftarrow 1$ *to* j **do**
        Compute $h_{i,j} = {v_i}^T w_j$.
        Update $w_j = w_j - h_{i,j} v_i$.
    end for
    Compute $h_{j+1,j} = \|W_j\|_2$.
    Update $v_{j+1} = \frac{w_j}{h_{j+1,j}}$.
    Compute $H_j = \begin{bmatrix} H_{j-1} & h_j \\ O & h_{j+1,j} \end{bmatrix}$.
    Update $V_{j+1} = \begin{bmatrix} V_j, v_{j+1} \end{bmatrix}$.
    Partition $\hat{H}_m = \begin{bmatrix} H_m \\ h_{m+1,m} e_m^T \end{bmatrix}$.
end for

Hence, the term H_m represents the projection A onto the Krylov subspace $\mathcal{K}_m(A, V)$ [67]. The Arnoldi decomposition is summarized in the Algorithm-1.

The eigenvalues λ_i of a projection matrix H_m in the Krylov subspace $\mathcal{K}_m(A, V)$, are known as the Ritz values and if χ is an eigenvector of H_m associate with λ , then $V_m \chi$ is called the Ritz vector belong to λ [68].

2.5 System Norms

The norms of the vectors and matrices are useful in the stability analysis and corresponding applications, i.e., stopping criteria and convergence analysis of the iterative techniques. In this section, some vector and matrix norms will be discussed with their properties.

2.5.1 Vector Norms

For the vector space $X \in \mathbb{R}^n$, a real valued function $\|.\| : \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be norm of X if for any $x, y \in X$ and $\alpha \in \mathbb{R}$ [69]. The norm of $\|.\|$ it satisfies the

following properties

- $\|x\| \geq 0$ and $\|x\| = 0$ if and only if $x = O$,
- $\|x + y\| \leq \|x\| + \|y\|$,
- $\|\alpha x\| = |\alpha|\|x\|$.

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

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}}; \quad 1 \leq p \leq \infty \tag{2.41}$$

In particular, when $p = 1, 2, \cdots, \infty$, the norm can be defined as

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|,$$
$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|} = \sqrt{x^T x},$$
$$\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|.$$

2.5.2 Matrix Norms

Let $A = [a_{ij}] \in \mathbb{R}^{n \times n}$, then the matrix norm $\|.\| : \mathbb{R}^{n \times n} \to \mathbb{R}$ induced by a vector p-norm is defined as

$$\|A\|_p = \sup_{x \neq O} \frac{\|Ax\|_p}{\|x\|_p}. \tag{2.42}$$

The most important matrix norm which is not induced by a vector norm is called Frobenius norm. The Frobenius norm of a matrix $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ is denoted by $\|A\|_F$ and defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2} = \sqrt{\text{tr}(AA^*)} = \sqrt{\text{tr}(A^*A)}. \tag{2.43}$$

The matrix norms induced by vector p-norms are sometimes called induced p-norms satisfying the properties of the vector norm [70].

In particular, the column-sum norm, spectral norm, and row-sum norm can be defined as

$$\|A\|_1 = \max_j \sum_{i=1}^{n} |a_{ij}|;$$
$$\|A\|_2 = \sqrt{\lambda_{max}(A^*A)},$$
$$\|A\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}|.$$

2.6 Existing Methods

In control theory, the CARE plays a leading role in the study of system stability and structural phenomena. To analyze the transient behaviors of many branches of engineering fields, the solution of CARE governed by corresponding system matrices is required [71]. Due to the gradual increase in the system size and complexity, the simulation techniques are upgrading over the years. Projection-based iterative methods have shown very effective for large-scale systems as they permit the sparsity pattern and provide low-rank approximated systems preserving properties of original systems [72]. A number of iterative methods employ the Galerkin projection technique to find a feasible solution for CARE.

In this section, some fundamental and newly developed methods for solving CARE will be introduced.

2.6.1 Schur Decomposition Method

Schur decomposition method based on Real Schur Factorization (RSF) of the Hamiltonian matrix for the CARE is one of the fundamental and oldest methods. Consider the converted to the standard system (2.6) and corresponding CARE can be defined as

$$\bar{A}^T X + X\bar{A} - X\bar{B}R^{-1}\bar{B}^T X + C^T C = O, \tag{2.44}$$

where $\bar{A} \in \mathbb{R}^{n\times n}$, $\bar{B} \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{m\times n}$ and $R \in \mathbb{R}^{p\times p}$.

Algorithm 2: Schur decomposition method.

Input : $\bar{A}, \bar{B}, C$ and R.

Output: The unique stabilizing solution X of the CARE.

1 Form the Hamiltonian matrix $H = \begin{bmatrix} \bar{A} & -\bar{B}R^{-1}\bar{B}^T \\ -C^TC & -\bar{A}^T \end{bmatrix}$.

2 Transform H to the RSF $S = U^THU = \begin{bmatrix} S_{11} & S_{12} \\ O & S_{22} \end{bmatrix}$.

3 Partition U comformably $U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix}$.

4 Compute $X = U_{21}U_{11}^{-1}$.

Assuming (A, B) is a stabilizable pair and (C, A) is detectable pair, whereas both of the pairs have full rank. So, the CARE (2.44) has a unique non-negative definite solution X. For the CARE (2.44) the Hamiltonian matrix can be written as

$$H = \begin{bmatrix} \bar{A} & -\bar{B}R^{-1}\bar{B}^T \\ -C^TC & -\bar{A}^T \end{bmatrix} \in \mathbb{R}^{2n\times 2n}. \tag{2.45}$$

To find the finite solution X of (2.44), it should be ensured that H has no pure imaginary eigenvalues [73]. Thus an orthogonal transformation matrix $U \in \mathbb{R}^{2n\times 2n}$ need to be found that puts H in ordered RSF as

$$S = U^THU = \begin{bmatrix} S_{11} & S_{12} \\ O & S_{22} \end{bmatrix}, \tag{2.46}$$

where $S_{ij} \in \mathbb{R}^{n\times n}$. The eigenvalues of H with negative real parts have been stacked in S_{11} and those with positive real parts are stacked in S_{22}. Let U is comfortably partitioned into four $n \times n$ blocks as

$$U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix}, \tag{2.47}$$

where the following relation is true

$$H \begin{bmatrix} U_{11} \\ U_{21} \end{bmatrix} = \begin{bmatrix} U_{11} \\ U_{21} \end{bmatrix} S_{11}. \tag{2.48}$$

Then, $X = U_{21}U_{11}^{-1}$ is the unique stabilizing solution of the CARE (2.44). The Schur decomposition method is summarized in the Algorithm-2.

2.6.2 Matrix-Sign Function Method

Matrix-sign function of a Hamiltonian matrix $H_{n\times n}$ is denoted by sign(H) and indicates the stable invariant subspace of H [74]. If H has no zero or purely imaginary eigenvalues, the Jordan Canonical Form (JCF) of H can be defined as

$$J = X^{-1}HX = D + N, \tag{2.49}$$

where $D = \text{diag}\{d_1, d_2, \cdots, d_n\}$ and N is nil-potent and commutes with D. Then, sign(H) can be defined as

$$\text{sign}(H) = X\text{diag}[\text{sign}(d_1), \text{sign}(d_2), \cdots, \text{sign}(d_n)]X^{-1}, \tag{2.50}$$

where $\text{sign}(d_i) = \pm 1$ for $\text{Re}(d_i) > 0$ and $\text{Re}(d_i) < 0$, respectively.

Using the matrix sign function method, the positive definite solution of generalized CARE can be obtained. Consider the generalized CARE (2.21) corresponding to the generalized state-space system (2.5) with E is non-singular. Then the corresponding Hamiltonian matrix can be factorized as

$$\begin{aligned}\bar{H} &= \begin{bmatrix} AE^{-1} & -BR^{-1}B^T \\ -E^{-T}C^TCE^{-1} & -E^{-T}A^T \end{bmatrix}, \\ &= \begin{bmatrix} I & O \\ O & E^{-T} \end{bmatrix} \begin{bmatrix} A & -BR^{-1}B^T \\ -C^TC & -A^T \end{bmatrix} \begin{bmatrix} E^{-1} & O \\ O & I \end{bmatrix}, \\ &= E_1^{-1}HE_2^{-1}. \end{aligned} \tag{2.51}$$

The matrix pencil $H - \lambda E_1 E_2 = H - \lambda L$ for $L = \begin{bmatrix} E & O \\ O & E^T \end{bmatrix}$ needs to be considered. Now, the sign$(\bar{H})$ can be defined as

$$\text{sign}(\bar{H}) = \bar{W} = \begin{bmatrix} \bar{W}_{11} & \bar{W}_{12} \\ \bar{W}_{21} & \bar{W}_{22} \end{bmatrix}. \tag{2.52}$$

Then, the positive definite solution X of the CARE (2.21) can be achieved from the matrix equation in the form

$$\begin{bmatrix} \bar{W}_{12} \\ \bar{W}_{22} + I \end{bmatrix} X = - \begin{bmatrix} \bar{W}_{11} + I \\ \bar{W}_{21} \end{bmatrix}. \tag{2.53}$$

Since E is non-singular, (2.53) can be written as

$$\begin{bmatrix} \bar{W}_{12} \\ E^T \bar{W}_{22} + E^T \end{bmatrix} XE = - \begin{bmatrix} \bar{W}_{11} E + E \\ E^T \bar{W}_{21} E \end{bmatrix}. \tag{2.54}$$

Assume $\bar{X} = XE$ and consider the following matrix decomposition

$$W = E_1 \left[\text{sign}(\bar{H}) \right] E_2 = \begin{bmatrix} \bar{W}_{11} E & \bar{W}_{12} \\ E^T \bar{W}_{21} E & E^T \bar{W}_{22} \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}. \tag{2.55}$$

Thus, equation (2.54) can be written as

$$\begin{bmatrix} W_{12} \\ W_{22} + E^T \end{bmatrix} \bar{X} = - \begin{bmatrix} W_{11} + E \\ W_{21} \end{bmatrix}. \tag{2.56}$$

Finally, the desired solution X of the generalized CARE (2.21) can be found as $X = \bar{X} E^{-1}$. The matrix sign function method is summarized in the Algorithm-3.

2.6.3 Rational Krylov Subspace Method

The Krylov subspace and the methods based on Krylov subspace were invented by the famous Russian naval engineer and applied mathematician **A. N. Krylov** (1863–1945) [75]. In this method, the aim is to approximate a large-scale system with a lower-dimensional system that has invariant characteristics with comparison to the original system. The ROMs can be used to replace the original system, as a component in larger simulations that are suitable for real-time applications. Several MOR techniques for state-space systems were developed recently, Rational Krylov Subspace Method (RKSM) is one of the most efficient approaches [76].

Algorithm 3: Matrix sign function method.

Input : E, A, B, C, and R.

Output: The unique positive definite solution X of the CARE.

Choose $W_0 = JH$ such that $J = \begin{bmatrix} O & I \\ -I & O \end{bmatrix}$ and $H = \begin{bmatrix} A & -BR^{-1}B^T \\ -C^TC & -A^T \end{bmatrix}$.

for $k \leftarrow 1$ *to* m **do**

  Compute $c = \left(\frac{|\det(W_k)|}{|\det(E_1E_2)|}\right)$, where E_1 & E_2 are defined in (2.51).

  Compute $W_{k+1} = \frac{1}{2c}\left[W_k + c^2(JE_1E_2)W_k^{-1}(JE_1E_2)\right]$.

end for

Partition $W = JE_1\left[\text{sign}(E_1^{-1}HE_2^{-1})\right]E_2 = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix}$.

For $\bar{X}$ solve $\begin{bmatrix} W_{12} \\ W_{22} + E^T \end{bmatrix}\bar{X} = -\begin{bmatrix} W_{11} + E \\ W_{21} \end{bmatrix}$.

Compute $X = \bar{X}E^{-1}$.

RKSM is a projection based iterative approach via block Arnoldi or Lanczos process. This iterative technique becomes competitive with ADI based techniques due to the recent developments and extensions. According to the real-world applications, RKSM has better results with the flexibility of the selection of interpolation points with comparison to the conventional methods for solving linear matrix equations [77].

In RKSM, an appropriate solution to the CARE is determined in the form of $X \approx V\hat{X}V^T$ for $X \in \mathbb{R}^{n\times m}$, $V \in \mathbb{R}^{n\times m}$, and $\hat{X} \in \mathbb{R}^{m\times m}$. At first, the columns of the projector V need to be determined, which span an orthonormal basis for the mp-dimensional Krylov subapace defined by

$$\mathcal{K}_m(A, B) = \text{span}\left(B, AB, A^2B, \cdots, A^{m-1}B\right). \tag{2.57}$$

The orthogonal basis $V_m = [v_1, v_2, \cdots, v_m]$ can be computed by the Arnoldi algorithm based on modified Gram-Schimdt process given in Algorithm-(1) from the Krylov subspace $\mathcal{K}_m$. Now, the main task is to derive a reduced-order Riccati equation

$$H_m^TY_mG_m + G_m^TY_mH_m - G_m^TY_mV_m^TBB^TV_mY_mG_m + V_m^TC^TCV_m = O, \tag{2.58}$$

which has a unique solution if and only if $\lambda_i + \lambda_j \neq 0$ for every pair of eigenvalues λ_i and λ_j for the real matrix H_m.

If A is sparse, computing the orthogonal columns of V_m by modified Gram-Schmidt, is the toughest part of the computation. The convergence of the Arnoldi method for solving CARE depends on the residual R_m corresponding to Y_m is defined by

$$R_m = A^T V_m Y_m V_m^T E + E^T V_m Y_m V_m^T A - E^T V_m Y_m V_m^T B B^T V_m Y_m V_m^T E + C^T C. \quad (2.59)$$

By the techniques of RKSM, the solution Y_m of (2.58) can be computed such that the Galerkin condition $V_m^T R_m V_m = O$ will be satisfied.

The detailed analysis of the Arnoldi decomposition-based rational Krylov subspace method will be discussed in Chapter-(3).

2.6.4 Kleinman-Newton Method

The solution of CARE is a laborious and complicated task, especially for equations arising from large-scale control systems. The CARE is often solved by means of well known Kleinman-Newton method. To reduce the simulation time, the execution of iterative solvers for the solution of the linear systems occurring at each Newton step is obvious. In the computation, control the accuracy of the solution of the linear systems and to gain efficiency without losing rate of convergence are the vital achievements [78]. The Kleinman-Newton iterative approach provides the desired goals and keeps the suitable inner iteration termination, which is time demanding [79].

In the Klenman-Newton method, the generalized CARE (2.21) is needed to be converted to the following generalized CALE as

$$\tilde{A}^T X E + E^T X \tilde{A} = -W W^T, \quad (2.60)$$

where $\tilde{A} = A - B B^T X E$ and $W = \begin{bmatrix} C^T & E^T X B \end{bmatrix}$. The generalized CALE (2.60) can be solved by any existing Lyapunov solvers, i.e., Bartels-Stewart's method [80].

Due to the gradually increased size of the control system matrices, the simulations of the generalized CALE (2.60) are not affordable for the direct solvers. So that, iterative solvers including MOR techniques are needed to be introduced. ADI method is one of the widely used techniques for solving the generalized CALE (2.60) by an iterative process.

The detailed analysis of Kleinman-Newton method implementing LRCF-ADI iterations will be discussed in Chapter-(4).

2.7 Shift Parameters

Shift parameters are the numerical weapon for the compensation of the system perturbations. They are the pre-conditioned and system-oriented random constants. For the quick and smooth convergence of the simulation process, adjustable shift selection is crucial. Usually, conventional Penzl's heuristic shifts and Wachspress's optimal shifts are commonly used for the large-scale descriptor systems. Nowadays, adaptive ADI shift selection approach has been adopted for sophisticated and larger descriptor systems. The ADI min-max problem is the key tool for generating shift parameters [81], which is defined as

$$\min_{\mu_1,\ldots,\mu_j\in\mathbb{C}^-} \left(\max_{1\leq l\leq n} \left| \prod_{i=1}^{J} \frac{\overline{\mu_i} - \lambda_l}{\mu_i + \lambda_l} \right| \right) ; \quad \lambda_l \in \Lambda(A,E), \tag{2.61}$$

where $\Lambda(A,E)$ represents the spectrum of the matrix pencil $\lambda E - A$.

In this section, we will introduce some of the techniques for finding useful shift parameters.

2.7.1 Penzl Heuristic Shifts

The heuristic approach is the most frequent approach to obtain shift parameters. In this case, the spectrum $\Lambda(A,E)$ is replaced recursively by a smaller set consisting k_+ Ritz and k_- reciprocal Ritz values with respect to $E^{-1}A$ and $A^{-1}E$, respectively by Arnoldi steps [82].

Penzl heuristic shifts are computationally unstable as even small changes in any of the parameters can cause significant alteration in the system performance. Computed Ritz values can have positive real parts if $A^TE + E^TA$ is indefinite and these must be neglected.

2.7.2 Wachspress Optimal and Sub-optimal Shifts

An analytic solution for the min-max problem (2.61) is proposed in [83], which uses the values $a := \min_i \mathrm{Re}(\lambda_i), b := \max_i \mathrm{Re}(\lambda_i)$ and $\Phi := \max_i \left(\tan^{-1}\left|\frac{\mathrm{Im}(\lambda_i)}{\mathrm{Re}(\lambda_i)}\right|\right)$ for $\lambda_i \in \Lambda(A,E)$ to generate the structure of the spectrum $\Lambda(A,E)$. The computation of the Wachspress optimal shifts through the elliptic integral consisting the tolerance τ and the spectral data a, b and Φ. If the spectrum $\Lambda(A,E)$ is real or real part dominant

always provides real shift parameters, whereas for imaginary dominant a modification will be done to produce complex shift parameters [84].

For large-scale matrices, it is suitable to obtain sub-optimal shift parameters from the spectrum $\Lambda(A, E)$ by a small number of k_+ Ritz and k_- harmonic Ritz values with respect to $E^{-1}A$ and $A^{-1}E$, respectively [85]. These Ritz values can be attained by Arnoldi or Lanczos processes and then the other computation will remain the same as before.

2.7.3 Adaptive ADI Shifts

In the case of adaptive shifts, the shifts are initially investigated by the eigenvalues of the matrix pencil $\lambda E - A$ projected to the span of C^T, where E, A and C are sparse and of appropriate dimensions. Once, all the shifts in the set have been used, the pencil is needed to be projected to the span of the current basis V_i and the current eigenvalues are used as the next set of shifts. The process required to be recursive and in each step the subspace to all the bases V_i generated with the current set of shifts will be extended [86]. note that, it can not be guaranteed that all of the projected eigenvalues will be contained in $\mathbb{C}^-$.

Let us consider W to be the orthogonal basis of the extended subspace and evaluate the eigenvalues of $\lambda W^T EW - W^T AW$. Choose a number of optimal shifts $\{\mu_i\}_{i=1}^J$ by solving the min-max problem (2.61) in the similar approach of the heuristic process. The recursive process will continue until the algorithm converged to the desired tolerance.

Chapter 3

Rational Krylov Subspace Method for the CARE Arising from Index-1 Descriptor System

3.1 Rational Krylov Subspace Method

For a very large dimensional system, the procedure of solving the CARE is highly time-consuming and needs an immense memory for storing it. Therefore, a low-rank approximate solution needs to be computed. The process can be manipulated by projecting the system onto a lower-dimensional rational Krylov subspace, which is an iterative approach and a crucial numerical weapon to get adequate efficiency. To do this, the rational Krylov subspace method (RKSM) can be used as an efficient tool.

Projection-based methods, such as RKSM, yield low-rank approximations and have applied effectively for solving linear matrix equations. The linearization strategy by ignoring the quadratic term and initial ad-hoc parameter implementation enhance the rapid convergence of the RKSM approach [87]. RKSM is a multi-step and recursive algorithm due to the span large approximation spaces to get a feasible solution but this iterative technique can be applied efficiently in the perturbed systems.

3.1.1 RKSM for Solving Standard CARE

Simoncini[5] applied RKSM approach for solving the CARE

$$A^T X + XA - XBR^{-1}B^T X + C^T C = O, \tag{3.1}$$

associated with the standard continuous-time LTI system

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

If the eigenvalues of the matrix A satisfy $\lambda_i + \bar{\lambda}_j \neq 0, \forall i, j = 1, 2, \cdots, n$, that ensures the solution X of the CARE (3.1) exists and unique. Through RKSM, the low-rank factor Z can be estimated for finding approximate solution $\hat{X}$ of the CARE (3.1), such that

$$X \approx V\hat{X}V^T; \quad V \in \mathbb{R}^{n\times r}, \quad \hat{X} \in \mathbb{R}^{r\times r}.$$

The orthogonal projector V spanned by the m-dimensional rational Krylov subspace for a set of shift parameters $\mu_i \in \mathbb{C}^+; i = 1, 2, \cdots, m$ is defined as

$$\mathcal{K}_m = \text{span}\left(C^T, (A^T - \mu_1 I)^{-1}C^T, \cdots, \prod_{i=1}^{m}(A^T - \mu_i I)^{-1}C^T\right).$$

If θ_j are the eigenvalues of $V_m^T A V_m$; $\mathbb{S}_m \in \mathbb{C}^+$ approximates the mirrored eigen-space of $A - BB^T X_m$ and $\delta\mathbb{S}_m$ is its border, the shifts are computed from

$$\mu_{m+1} = \arg\left(\max_{\mu\in\delta\mathbb{S}_m}\left|\frac{\prod_{i=1}^{m}(\mu - \mu_j)}{\prod_{i=1}^{m}(\mu - \theta_j)}\right|\right).$$

According to the Galerkin condition

$$V^T(A^T X + XA - XBR^{-1}B^T X + C^T C)V = O.$$

After simplification, the low-rank CARE can be obtained as

$$\hat{A}^T\hat{X} + \hat{X}\hat{A} - \hat{X}\hat{B}R^{-1}\hat{B}^T\hat{X} + \hat{C}^T\hat{C} = O, \tag{3.3}$$

[5]http://www.dm.unibo.it/ simoncin/welcome.html

where $\hat{X} = V^T XV, \hat{A} = V^T AV, \hat{B} = V^T B$ and $\hat{C} = CV$. The equation (3.3) is an approximated low-rank CARE and can be solved by MATLAB `care` command.

Here $\hat{X}$ is taken as the low-rank approximation of X, corresponding to the low-rank CARE (3.3). Then residual of the $(m+1)$-th iteration is

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

where S is a block upper triangular matrix in the QR factorization of the matrix U derived as

$$U = \begin{bmatrix} v_{m+1}\mu_{m+1} \\ V_m \hat{X} H_m^{-1} e_m h_{m+1,m}^T \\ -(I - V_m V_m^T) A^T v_{m+1} \end{bmatrix}^T,$$

where H_m is a block upper Hessenberg matrix and e_m is the matrix formed by the last p columns of the mp-order identity matrix.

For $C^T = QR$ such that $R = \beta$, the relative-residual can be estimated as

$$\|\mathcal{R}_m\|_{(\text{relative})} = \frac{\|\mathcal{R}_m\|_F}{\|\beta^T \beta\|_F}.$$

The low-rank solution $\hat{X}$ of (3.3) is symmetric, positive definite and can be factorized as $\hat{X} = YY^T$. Using the matrix property, the original solution X can be reproduced as $X = V\hat{X}V^T$.

By the eigenvalue decomposition to the approximate solution $\hat{X}$ and truncating the negligible eigenvalues, the possible lowest order factor Z of X can be estimated. The factorization will be carried out as

$$\begin{aligned} X = V\hat{X}V^T &= V(S\Lambda S^T)V^T \\ &= V \begin{bmatrix} S_1 & S_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & O \\ O & \Lambda_2 \end{bmatrix} \begin{bmatrix} S_1^T \\ S_2^T \end{bmatrix} V^T \\ &= VS_1\Lambda_1 S_1^T V^T = (VS_1\Lambda_1^{\frac{1}{2}})(VS_1\Lambda_1^{\frac{1}{2}})^T \\ &= ZZ^T. \end{aligned}$$

Here the truncated Λ_2 contains the negligible eigenvalues. The summary of the above process is given in the Algorithm-4.

Algorithm 4: S-RKSM.

Input : A, B, C, R, i_{max} (number of iterations) and μ_i (initial shifts).
Output: Low-rank factored solution Z such that $X \approx ZZ^T$.

Compute $QR = C^T$ (QR factorization).
Choose $V_1 = Q$.
while *not converged* *or* $m \leq i_{max}$ **do**
  Solve $v_m = (A^T - \mu_{m+1} I)^{-1} V_m$.
  Compute shift for the next iteration.
  Using Arnoldi algorithm orthogonalize v_m against V_m to obtain v_{m+1}, such that $V_{m+1} = \begin{bmatrix} V_m, v_{m+1} \end{bmatrix}$.
  Assuming $\hat{A} = V_{m+1}^T A V_{m+1}$, $\hat{B} = V_{m+1}^T B$ and $\hat{C} = C V_{m+1}$, for $\hat{X}$ solve the reduced order Riccati equation $\hat{A}^T \hat{X} + \hat{X}\hat{A} - \hat{X}\hat{B}R^{-1}\hat{B}^T\hat{X} = -\hat{C}^T\hat{C}$.
  Compute $\|\mathcal{R}_m\|_{(\text{relative})}$ for convergence.
end while
Compute eigenvalue decomposition $\hat{X} = T\Lambda T^T = \begin{bmatrix} T_1 & T_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & O \\ O & \Lambda_2 \end{bmatrix} \begin{bmatrix} T_1^T \\ T_2^T \end{bmatrix}$.
For negligible eigenvalues truncate Λ_2 and compute $Z = V_{m+1} T_1 \Lambda_1^{\frac{1}{2}}$.

3.1.2 RKSM for Solving Generalized CARE

The generalized CARE can be defined as

$$A^T X E + E^T X A - E^T X B R^{-1} B^T X E + C^T C = O, \tag{3.4}$$

and the corresponding generalized continuous-time LTI system is

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{3.5}$$

In this case, the existence of the unique solution X of the generalized CARE (3.4) can be ensured if eigenvalues of the matrix pair (A, E) satisfy $\lambda_i + \bar{\lambda}_j \neq 0, \forall i, j = 1, 2, \cdots, n$. Also, θ_j are the eigenvalues of $(V_m^T A V_m, V_m^T E V_m)$.

Then the orthogonal projector V spanned by the m dimensional rational Krylov subspace for a set of shift parameters $\mu_i \in \mathbb{C}^+; i = 1, 2, \cdots, m$ can be generalized as

$$\mathcal{K}_m = \text{span}\left(C^T, (A^T - \mu_1 E)^{-1} C^T, \cdots, \prod_{i=1}^{m} (A^T - \mu_i E)^{-1} C^T \right).$$

Algorithm 5: G-RKSM.

Input : E, A, B, C, R, i_{max} (number of iterations) and μ_i (initial shifts).
Output: Low-rank factored solution Z such that $X \approx ZZ^T$.

Compute $QR = C^T$ (QR factorization).
Choose $V_1 = Q$.
while *not converged* *or* $m \leq i_{max}$ **do**
  Solve $v_m = (A^T - \mu_{m+1}E^T)^{-1}V_m$.
  Compute shift for the next iteration.
  Using Arnoldi algorithm orthogonalize v_m against V_m to obtain v_{m+1}, such that $V_{m+1} = \begin{bmatrix} V_m, v_{m+1} \end{bmatrix}$.
  Assuming $\hat{E} = V_{m+1}^T E V_{m+1}$, $\hat{A} = V_{m+1}^T A V_{m+1}$, $\hat{B} = V_{m+1}^T B$ and $\hat{C} = C V_{m+1}$, for $\hat{X}$ solve the reduced order Riccati equation $\hat{A}^T\hat{X}\hat{E} + \hat{E}^T\hat{X}\hat{A} - \hat{E}^T\hat{X}\hat{B}R^{-1}\hat{B}^T\hat{X}\hat{E} = -\hat{C}^T\hat{C}$.
  Compute $\|\mathcal{R}_m\|_{(\text{relative})}$ for convergence.
end while
Compute eigenvalue decomposition $\hat{X} = T\Lambda T^T = \begin{bmatrix} T_1 & T_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & O \\ O & \Lambda_2 \end{bmatrix} \begin{bmatrix} T_1^T \\ T_2^T \end{bmatrix}$.
For negligible eigenvalues truncate Λ_2 and compute $Z = V_{m+1}T_1\Lambda_1^{\frac{1}{2}}$.

As before, assuming $\hat{X} = V^T X V, \hat{E} = V^T E V, \hat{A} = V^T A V, \hat{B} = V^T B$ and $\hat{C} = CV$, the approximated low-rank CARE for the generalized CARE (3.4) can be formed as

$$\hat{A}^T\hat{X}\hat{E} + \hat{E}^T\hat{X}\hat{A} - \hat{E}^T\hat{X}\hat{B}R^{-1}\hat{B}^T\hat{X}\hat{E} + \hat{C}^T\hat{C} = O. \tag{3.6}$$

The matrix U needs to be rearranged as

$$U = \begin{bmatrix} v_{m+1}\mu_{m+1} \\ E^T V_m \hat{X} H_m^{-1} e_m h_{m+1,m}^T \\ -(I - V_m V_m^T)A^T v_{m+1} \end{bmatrix}^T,$$

Because of the above extensions, Algorithm-(4) needs to be updated for the generalized CARE, which is given in Algorithm-(5).

3.2 Structured Generalized System Derived from Index-1 Descriptor System

The index-1 descriptor system needs to be structured into the generalized system and apply RKSM to find the low-rank solution of the corresponding CARE. The semi-explicit

form of a index-1 descriptor system can be represented in sparse form as

$$\begin{aligned}\underbrace{\begin{bmatrix} E_1 & E_2 \\ O & O \end{bmatrix}}_{E}\underbrace{\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix}}_{\dot{x}(t)} &= \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_{A}\underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{B} u(t), \\ y(t) &= \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{C}\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + D_a u(t).\end{aligned} \tag{3.7}$$

Here $x_1(t) \in \mathbb{R}^{n_1}$, $x_2(t) \in \mathbb{R}^{n_2}$ with $n_1 + n_2 = n$ are state vectors called differential and algebraic variables, respectively. The sub-matrices $E_1, E_2, J_1, J_2, J_3, B_1, B_2, C_1, C_2$ and D_a are sparse in appropriate dimensions with E_1 and J_1 are of full rank. Since J_4 is non-singular (i.e. $\det(A) \neq 0$), the system (3.7) is an index-1 descriptor system.

The equations in (3.7) are equivalent to the DAEs given below

$$\begin{aligned} E_1\dot{x_1} + E_2\dot{x_2} &= J_1x_1(t) + J_2x_2(t) + B_1u(t), \\ O &= J_3x_1(t) + J_4x_2(t) + B_2u(t), \\ y(t) &= C_1x_1(t) + C_2x_2(t) + D_au(t). \end{aligned} \tag{3.8}$$

From the algebraic (second) equation of the (3.8), X_2 can be eliminated as

$$x_2(t) = -J_4^{-1}J_3x_1(t) - J_4^{-1}B_2u(t). \tag{3.9}$$

By proper substitution of (3.9) in the DAEs (3.8) can be structured to the generalized system as

$$\begin{aligned} \mathcal{E}\dot{x}(t) &= \mathcal{A}x(t) + \mathcal{B}\bar{u}(t), \\ y(t) &= \mathcal{C}x(t) + \mathcal{D}\bar{u}(t), \end{aligned} \tag{3.10}$$

where, the following `Schur` complements have been considered

$$\begin{aligned} &\mathcal{E} := E_1 - E_2{J_4}^{-1}J_3, \quad \mathcal{A} := J_1 - J_2{J_4}^{-1}J_3, \\ &\mathcal{B} := [B_1 - J_2{J_4}^{-1}B_2, E_2{J_4}^{-1}B_2], \quad \mathcal{C} := C_1 - C_2{J_4}^{-1}J_3, \\ &\mathcal{D} := [D_a - C_2{J_4}^{-1}B_2, O], \quad x := x_1, \quad \bar{u}(t) := [u^T(t), \dot{u}^T(t)]^T. \end{aligned} \tag{3.11}$$

Therefore, the index-1 descriptor system (3.7) can be structured as the generalized system (3.10) by proper elimination as substitution as above. It to be noted that the structured matrices in (3.10) are formed in dense form.

Following lemma shows that the structured generalized system (3.10) is equivalent to the corresponding index-1 descriptor system (3.7) in the sense of their transfer functions and finite spectrum.

Lemma 3.1 (**Equivalence of Transfer Functions**). *Assume the transfer functions* $G(s) = C(sE - A)^{-1} + D_a$ *and* $\mathcal{G}(s) = \mathcal{C}(s\mathcal{E} - \mathcal{A})^{-1} + \mathcal{D}$ *are obtained from the semi-explicit index-1 descriptor system (3.7) and the structured generalized system (3.10), respectively. Then, the transfer functions* $G(s)$ *and* $\mathcal{G}(s)$ *are identical.*

Then, the generalized CARE corresponding to (3.10) can be defined as

$$\mathcal{A}^T X \mathcal{E} + \mathcal{E}^T X \mathcal{A} - \mathcal{E}^T X \mathcal{B} R^{-1} \mathcal{B}^T X \mathcal{E} + \mathcal{C}^T \mathcal{C} = O, \tag{3.12}$$

In case of power system model $E_2 = O$, then the system (3.7) can be written as

$$\begin{aligned} \underbrace{\begin{bmatrix} E_1 & O \\ O & O \end{bmatrix}}_{E} \underbrace{\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix}}_{\dot{x}(t)} &= \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{B} u(t), \\ y(t) &= \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{C} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + D_a u(t). \end{aligned} \tag{3.13}$$

In this case, the DAEs (3.8) can be re-defined as

$$\begin{aligned} E_1 \dot{x_1} &= J_1 x_1(t) + J_2 x_2(t) + B_1 u(t), \\ O &= J_3 x_1(t) + J_4 x_2(t) + B_2 u(t), \\ y(t) &= C_1 x_1(t) + C_2 x_2(t) + D_a u(t). \end{aligned} \tag{3.14}$$

In the similar manner, the index-1 descriptor system (3.13) defined for power system model can be structured into the generalized system (3.10) considering the following `Schur` complements

$$\begin{aligned} &\mathcal{E} := E_1, \quad \mathcal{A} := J_1 - J_2 {J_4}^{-1} J_3, \\ &\mathcal{B} := B_1 - J_2 {J_4}^{-1} B_2, \quad \mathcal{C} := C_1 - C_2 {J_4}^{-1} J_3, \\ &\mathcal{D} := D_a - C_2 {J_4}^{-1} B_2, \quad x := x_1, \quad \bar{u}(t) := u(t). \end{aligned} \tag{3.15}$$

3.3 RKSM for Solving Structured Generalized CARE

Let us introduce an orthogonal projector V spanned by the m dimensional rational Krylov subspace for a set of given adaptive shift parameters $\mu_i \in \mathbb{C}; i = 1, 2, \cdots, m$ defined as

$$\mathcal{K}_m = \text{span}\left(\mathcal{C}^T, (\mathcal{A}^T - \mu_1\mathcal{E})^{-1}\mathcal{C}^T, \cdots, \prod_{i=1}^{m}(\mathcal{A}^T - \mu_i\mathcal{E})^{-1}\mathcal{C}^T\right).$$

According to the Galerkin condition, we have

$$V^T(\mathcal{A}^T X\mathcal{E} + \mathcal{E}^T X\mathcal{A} - \mathcal{E}^T X\mathcal{B}R^{-1}\mathcal{B}^T X\mathcal{E} + \mathcal{C}^T\mathcal{C})V = O.$$

After simplification, the low-rank CARE can be written as

$$\hat{\mathcal{A}}^T\hat{X}\hat{\mathcal{E}} + \hat{\mathcal{E}}^T\hat{X}\hat{\mathcal{A}} - \hat{\mathcal{E}}^T\hat{X}\hat{\mathcal{B}}R^{-1}\hat{\mathcal{B}}^T\hat{X}\hat{\mathcal{E}} + \hat{\mathcal{C}}^T\hat{\mathcal{C}} = O. \tag{3.16}$$

Here, $\hat{X} = V^T XV, \hat{\mathcal{E}} = V^T\mathcal{E}V, \hat{\mathcal{A}} = V^T\mathcal{A}V, \hat{\mathcal{B}} = V^T\mathcal{B}$ and $\hat{\mathcal{C}} = \mathcal{C}V$. The equation (3.16) is the low-rank CARE and can be solved by any existing method, such as `Schur`-decomposition or MATLAB `care` command. $\hat{X}$ is taken as the low-rank approximation of X, corresponding to the low-rank CARE (3.16).

3.3.1 Convergence Criteria and Related Theorems

Arnoldi relation is a very essential tool for the computation of residual of the RKSM iterations. Following lemma highlights the formulation of the Arnoldi relation for the shifts $\mu_i \in \mathbb{C}; i = 1, 2, \cdots, m$ [88, 89].

Lemma 3.2 (**Arnoldi Relation**). *The rational Krylov subspace $\mathcal{K}_m = span(V_m)$ for $m \geq 1$ satisfies the Arnoldi relation as follows*

$$\mathcal{A}^T V_m = V_m T_m + v_{m+1}g_m^T; \quad V_m^T V_m = I, \tag{3.17}$$

where $\hat{v}_{m+1}\beta = v_{m+1}\mu_m - (I - V_m V_m^T)\mathcal{A}^T v_{m+1}$ is the QR decomposition of the right hand side matrix with $g_m^T = \beta h_{m+1,m}e_m^T H_m^{-1}$.

To avoid extra matrix-vector multiplies per iteration, the computation of T_m can be performed more efficiently than the explicit product $T_m = V_m^T \mathcal{A} V_m$. The following lemma describes the procedure suggested by Ruhe, where an extra multiplication by $\mathcal{A}$ is only performed to build the projected matrix [90].

Lemma 3.3 (**Building the Projected Matrix**). *Let the columns of V_m be an orthonormal basis of the rational Krylov subspace $\mathcal{K}_m$ with $D_m = diag(\mu_1, \mu_2, \cdots, \mu_m)$. Then the following relation holds for the projected matrix T_m.*

$$T_m = V_m^T \mathcal{A} V_m = \left(I + H_m D_m - V_m^T \mathcal{A} v_{m+1} h_{m+1,m} e_m^T\right) H_m^{-1}. \tag{3.18}$$

In the Lemma-(3.2) and Lemma-(3.3), the matrix $H_{m+1,m} = \begin{bmatrix} H_m \\ h_{m+1,m} e_m^T \end{bmatrix}$ contains the orthogonalization co-efficient that generates the orthogonal columns of V_{m+1} and e_m is the last p columns of the mp-th order identity matrix.

Theorem 3.4 (**Residual of the RKSM Iterations**). *Let V_m be the orthogonal projector spanned by the rational Krylov subspace $\mathcal{K}_m$ and $X \approx V \hat{X} V^T$ is the solution of the continuous algebraic Riccati equation using the low-rank solution $\hat{X}$. Then, the residual of m-th iteration can be computed as*

$$\|\mathcal{R}_m\|_F = \|SJS^T\|_F; \quad J = \begin{bmatrix} O & I & O \\ I & O & I \\ O & I & O \end{bmatrix}, \tag{3.19}$$

where $\|.\|_F$ denotes the Frobenius norm and S is a block upper triangular matrix in the QR factorization of the matrix U defined as

$$U = \begin{bmatrix} v_{m+1}\mu_{m+1} & \mathcal{E}^T V_m \hat{X} H_m^{-1} e_m h_{m+1,m}^T & -(I - V_m V_m^T)\mathcal{A}^T v_{m+1} \end{bmatrix}. \tag{3.20}$$

Proof. Assume $g = (I - V_m V_m^T)\mathcal{A}^T v_{m+1}$ and consider the reduced QR factorization $\mathcal{C}^T = V_1 \beta_0$. Then, by putting the relations in equation (3.17) of Lemma-(3.2), the relation can be defined

$$\mathcal{A}^T V_m = V_m T_m + v_{m+1}\mu_m h_{m+1,m} e_m^T H_m^{-1} - g h_{m+1,m} e_m^T H_m^{-1}. \tag{3.21}$$

The residual of the CARE (3.12) can be written as

$$\mathcal{R} = \mathcal{A}^T X \mathcal{E} + \mathcal{E}^T X \mathcal{A} + \mathcal{E}^T X \mathcal{B} R^{-1} \mathcal{B}^T X \mathcal{E} + \mathcal{C}^T \mathcal{C}. \tag{3.22}$$

Consider the approximate solution using the low-rank solution $\hat{X}$ as $X = V_m \hat{X} V_m^T$ and equation-(3.18) in Lemma-(3.3), then applying (3.21) in (3.22), we get

$$\begin{aligned}
\mathcal{R}_m &= v_{m+1}\mu_m h_{m+1,m} e_m^T H_m^{-1} \hat{X}_m V_m^T \mathcal{E} - g h_{m+1,m} e_m^T H_m^{-1} \hat{X}_m V_m^T \mathcal{E} \\
&+ \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T \mu_m^T v_{m+1}^T - \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T g^T + V_m T_m^T \hat{X}_m V_m^T \mathcal{E} \\
&+ \mathcal{E}^T V_m \hat{X}_m T_m V_m^T + \mathcal{E}^T X \mathcal{B} R^{-1} \mathcal{B}^T X \mathcal{E} + \mathcal{C}^T \mathcal{C}, \\
&= \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T \mu_m^T v_{m+1}^T + (v_{m+1}\mu_m - g) h_{m+1,m} e_m^T H_m^{-1} \hat{X}_m^T V_m^T \mathcal{E} \\
&- \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T g^T + V_m V_m^T \mathcal{A}^T V_m \hat{X}_m V_m^T \mathcal{E} + \mathcal{E}^T V_m \hat{X}_m V_m^T \mathcal{A} V_m V_m^T \\
&+ \mathcal{E}^T X \mathcal{B} R^{-1} \mathcal{B}^T X \mathcal{E} + \mathcal{C}^T \mathcal{C}, \\
&= \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T \mu_m^T v_{m+1}^T + (v_{m+1}\mu_m - g) h_{m+1,m} e_m^T H_m^{-1} \hat{X}_m^T V_m^T \mathcal{E} \\
&- \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T g^T + (\mathcal{A} V_m V_m^T)^T X \mathcal{E} + \mathcal{E}^T X (\mathcal{A} V_m V_m^T) \\
&+ \mathcal{E}^T X \mathcal{B} R^{-1} \mathcal{B}^T X \mathcal{E} + \mathcal{C}^T \mathcal{C}, \\
&= \begin{bmatrix} \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T & v_{m+1}\mu_m - g & \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T \end{bmatrix} \\
&\times \begin{bmatrix} \mu_m^T v_{m+1}^T \\ h_{m+1,m} e_m^T H_m^{-1} \hat{X}_m^T V_m^T \mathcal{E} \\ -g^T \end{bmatrix} + O, \\
&= \begin{bmatrix} v_{m+1}\mu_m & \mathcal{E}^T V_m \hat{X}_m H_m^{-T} e_m h_{m+1,m}^T & -g \end{bmatrix} \begin{bmatrix} O & I & O \\ I & O & I \\ O & I & O \end{bmatrix} \begin{bmatrix} \mu_m^T v_{m+1}^T \\ h_{m+1,m} e_m^T H_m^{-1} \hat{X}_m^T V_m^T \mathcal{E} \\ -g^T \end{bmatrix} \\
&= S J S^T.
\end{aligned} \tag{3.23}$$

Thus, the proof follows from equation (3.23). □

Then, using the reduced QR factorization $\mathcal{C}^T = V_1 \beta_0$ the relative-residual can be estimated as follows

$$\|\mathcal{R}_m\|_{(\text{relative})} = \frac{\|\mathcal{R}_m\|_F}{\|\beta_0^T \beta_0\|_F}. \tag{3.24}$$

The computation of the norm of the approximate solution is high time expensive and setting stopping criteria based on direct simulation is impractical due to several constraints. To overcome these difficulties and have smooth convergence, the relative residual (3.24) can be used efficiently.

Thus, the simulation process must be stopped, if the value of relative residual under a certain margin of tolerance, i.e. $\|\mathcal{R}_m\|_{(\text{relative})} \leq tol$.

3.3.2 Treatment for the Unstable Systems

If system (3.10) is unstable an initial feedback stabilization is required by using the initial-feedback matrix $K = K_0$ and for each iteration, it needs to be updated as $K = (\mathcal{B}^T \hat{X})V\mathcal{E}$ by the solution of (3.16). Then, $\mathcal{A}$ needs to be replaced by $\mathcal{A}_f = \mathcal{A} - \mathcal{B}K$. Thus, the rational Krylov subspace for the projector V needs to be redefined as

$$\mathcal{K}_m = \text{span}\left(\mathcal{C}^T, (\mathcal{A}_f^T - \mu_1\mathcal{E})^{-1}\mathcal{C}^T, \cdots, \prod_{i=1}^{m}(\mathcal{A}_f^T - \mu_i\mathcal{E})^{-1}\mathcal{C}^T\right).$$

Furthermore, if the power system that governs the descriptor system has very critical components of large dimensions, it is a very time consuming and complex process to find the initial feedback matrix K_0. In this circumstance, introduce $K_0 = O$ initially and update the value of K at each iteration as $K = (\mathcal{B}^T \hat{X})V\mathcal{E}$, where the other steps will remain same.

3.3.3 Sparse Form of the Basis of the Krylov Subspace

The matrix $\mathcal{A}$ in (3.10) is in dense form and the convergence of the structured system becomes slow and has ill-conditioned efficiency. To bypass these drawbacks at each iteration, a shifted linear system needs to be solved in the form

$$\begin{aligned} (\mathcal{A}^T - \mu_i\mathcal{E})v_i &= V_{i-1}, \\ or, ((J_1 - J_2{J_4}^{-1}J_3)^T - \mu_i E_1)v_i &= V_{i-1}. \end{aligned} \tag{3.25}$$

Thus, for v_i the following linear system needs to be solved

$$\begin{bmatrix} J_1 - \mu_i E_1 & J_2 \\ J_3 & J_4 \end{bmatrix}^T \begin{bmatrix} v_i \\ * \end{bmatrix} = \begin{bmatrix} V_{i-1} \\ O \end{bmatrix}. \tag{3.26}$$

Here $*$ is the truncated term, which is connected to the algebraic part of the descriptor system. Though the linear system (3.26) is higher dimensional, it is sparse and can be solved by the sparse-direct solver very efficiently [91].

The whole procedure to solve CARE using the low-rank factored solution is summarized in the Algorithm-6.

Algorithm 6: G-RKSM (Structured).

Input : $E_1, J_1, J_2, J_3, J_4, B_1, B_2, C_1, C_2, R, i_{max}$ (number of iterations) and μ_i (initial shifts).

Output: Low-rank factored solution Z such that $X \approx ZZ^T$ and optimal feedback matrix K^o.

Compute $QR = (C_1 - C_2 {J_4}^{-1} J_3)^T$ (QR factorization).
Choose $V_1 = Q$.
while *not converged* *or* $m \leq i_{max}$ **do**
  Solve the linear system (3.26) for v_m.
  Compute adaptive shifts for the next iterations (if store is empty).
  Using Arnoldi algorithm orthogonalize v_m against V_m to obtain v_{m+1}, such that $V_{m+1} = \begin{bmatrix} V_m, v_{m+1} \end{bmatrix}$.
  Assuming $\hat{\mathcal{E}}$, $\hat{\mathcal{A}}$, $\hat{\mathcal{B}}$ and $\hat{\mathcal{C}}$ are defined in (3.29), for $\hat{X}$ solve the reduced order Riccati equation $\hat{\mathcal{A}}^T \hat{X} \hat{\mathcal{E}} + \hat{\mathcal{E}}^T \hat{X} \hat{\mathcal{A}} - \hat{\mathcal{E}}^T \hat{X} \hat{\mathcal{B}} R^{-1} \hat{\mathcal{B}}^T \hat{X} \hat{\mathcal{E}} = -\hat{\mathcal{C}}^T \hat{\mathcal{C}}$.
  Compute $\|\mathcal{R}_m\|_{(\text{relative})}$ for convergence.
end while
Compute eigenvalue decomposition $\hat{X} = T \Lambda T^T = \begin{bmatrix} T_1 & T_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & O \\ O & \Lambda_2 \end{bmatrix} \begin{bmatrix} T_1^T \\ T_2^T \end{bmatrix}$.
For negligible eigenvalues truncate Λ_2 and construct $Z = V_{m+1} T_1 \Lambda_1^{\frac{1}{2}}$.
Compute the optimal feedback matrix $K^o = R^{-1}(B_1 - J_2 {J_4}^{-1} B_2)^T (ZZ^T) E_1$.

For the stabilized system using the initial-feedback matrix K_0, the expressions (3.25) and (3.26) can be written as

$$\begin{aligned} (\mathcal{A}_f^T - \mu_i \mathcal{E}) v_i &= V_{i-1}, \\ or, (((J_1 - B_1 K_0) - J_2 {J_4}^{-1} (J_3 - B_2 K_0))^T - \mu_i E_1) v_i &= V_{i-1}, \end{aligned} \tag{3.27}$$

$$\begin{bmatrix} (J_1 - B_1 K_0) - \mu_i E_1 & J_2 \\ J_3 - B_2 K_0 & J_4 \end{bmatrix}^T \begin{bmatrix} v_i \\ * \end{bmatrix} = \begin{bmatrix} V_{i-1} \\ O \end{bmatrix}. \tag{3.28}$$

To evaluate the shifted linear system (3.27), explicit inversion of $(\mathcal{A} - \mathcal{B}K)$ (usually dense) should be avoided in practice, instead the *Sherman-Morrison-Woodbury* formula needs to be used as follows

$$(\mathcal{A} - \mathcal{B}K)^{-1} = \mathcal{A}^{-1} + \mathcal{A}^{-1} \mathcal{B} (I - K \mathcal{A}^{-1} \mathcal{B})^{-1} K \mathcal{A}^{-1}.$$

This iterative process will be continued until the desired convergence achieved. The stabilization approach with the implementation of the initial feedback matrix K_0 is summarized in the Algorithm-7.

Algorithm 7: G-RKSM-FB (Structured).

Input : $E_1, J_1, J_2, J_3, J_4, B_1, B_2, C_1, C_2, R, K_0$ (initial feedback matrix) i_{max} (number of iterations) and μ_i (initial shifts).

Output: Low-rank factored solution Z such that $X \approx ZZ^T$ and optimal feedback matrix K^o.

Compute $QR = (C_1 - C_2 {J_4}^{-1} J_3)^T$ (QR factorization).
Choose $V_1 = Q$.
Choose $K = K_0$.
while *not converged* *or* $m \leq i_{max}$ **do**
  Solve the linear system (3.28) for v_m.
  Compute adaptive shifts for the next iterations (if store is empty).
  Using Arnoldi algorithm orthogonalize v_m against V_m to obtain v_{m+1}, such that $V_{m+1} = \begin{bmatrix} V_m, v_{m+1} \end{bmatrix}$.
  Assuming $\hat{\mathcal{E}}$, $\hat{\mathcal{A}}$, $\hat{\mathcal{B}}$ and $\hat{\mathcal{C}}$ are defined in (3.29), for $\hat{X}$ solve the reduced order Riccati equation $\hat{\mathcal{A}}^T \hat{X} \hat{\mathcal{E}} + \hat{\mathcal{E}}^T \hat{X} \hat{\mathcal{A}} - \hat{\mathcal{E}}^T \hat{X} \hat{\mathcal{B}} R^{-1} \hat{\mathcal{B}}^T \hat{X} \hat{\mathcal{E}} = -\hat{\mathcal{C}}^T \hat{\mathcal{C}}$.
  Update $K = (\hat{\mathcal{B}}^T \hat{X}) V_{m+1} E_1$.
  Compute $\|\mathcal{R}_m\|_{(\text{relative})}$ for convergence.
end while
Compute eigenvalue decomposition $\hat{X} = T \Lambda T^T = \begin{bmatrix} T_1 & T_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & O \\ O & \Lambda_2 \end{bmatrix} \begin{bmatrix} T_1^T \\ T_2^T \end{bmatrix}$.
For negligible eigenvalues truncate Λ_2 and construct $Z = V_{m+1} T_1 \Lambda_1^{\frac{1}{2}}$.
Compute the optimal feedback matrix $K^o = R^{-1}(B_1 - J_2 {J_4}^{-1} B_2)^T (ZZ^T) E_1$.

3.3.4 Sparse Form of the Matrices in the Reduced Order System

The matrices in the structured generalized system are in the dense form and this is contradictory to the aim of the work. To resolve this issue, explicit form of the reduced order matrices will not be used to construct reduced order system. The sparsity preserving reduced order matrices can be attained by following way

$$\begin{aligned} &\hat{\mathcal{E}} = V^T E_1 V, \quad \hat{\mathcal{A}} = V^T J_1 V - (V^T J_2) {J_4}^{-1} (J_3 V), \\ &\hat{\mathcal{B}} = V^T B_1 - (V^T J_2) {J_4}^{-1} B_2, \quad \hat{\mathcal{C}} = C_1 V - C_2 {J_4}^{-1} (J_3 V). \end{aligned} \tag{3.29}$$

For the improvement of the consistency of the proposed algorithms, the matrix formulations in (3.29) will be implemented.

3.3.5 Optimally Stabilized System

The low-rank solution $\hat{X}$ is symmetric and positive definite and can be factorized as $\hat{X} = YY^T$. Using the matrix property, the original solution can be reproduced as $X = V\hat{X}V^T$. Then, the desired low-rank factored solution $Z = VY$ of the CARE (3.12) can be stored.

At the end, the optimal feedback matrix $K^o = R^{-1}\mathcal{B}^T X\mathcal{E} = R^{-1}\mathcal{B}^T(ZZ^T)\mathcal{E}$ can be achieved from the final low-rank factored solution Z and assuming $\mathcal{A}_s = \mathcal{A} - \mathcal{B}K^o$ the system (3.10) can be optimally stabilized as

$$\begin{aligned} \mathcal{E}\dot{x}(t) &= \mathcal{A}_s x(t) + \mathcal{B}u(t), \\ y(t) &= \mathcal{C}x(t) + \mathcal{D}u(t). \end{aligned} \tag{3.30}$$

Chapter 4

Kleinman-Newton Method Based on LRCF-ADI for the CARE Arising from Index-1 Descriptor System

4.1 Alternative Direction Implicit Method

Alternative Direction Implicit (ADI) iteration was developed by **Peaceman *et al.*** in 1955, for solving the linear system $MX = b$, where the matrix $M = M_1 + M_2 \in \mathbb{R}^{n\times n}$ is symmetric, positive definite arising in the numerical solution of partial differential equations [92]. If M represents a centered finite difference discretization in two-dimensional space, then M_1 and M_2 can be chosen as finite difference dicretizations with respect to x and y direction, respectively. The ADI iteration is for $i = 1, 2, \cdots$ (when converges). defined in terms of double steps as

$$\begin{aligned}(M_1 + \mu_i I_n)X_{i-\frac{1}{2}} &= (\mu_i I_n - M_2)X_{i-1} + b,\\ (M_2 + \mu_i I_n)X_i &= (\mu_i I_n - M_1)X_{i-\frac{1}{2}} + b,\end{aligned} \tag{4.1}$$

where $\mu_i \in \mathbb{R}^+$ are the shift parameters such that the above ADI iteration converges with super-linear convergence rate [93].

4.1.1 Single-step ADI Scheme for Solving Generalized CALE

Consider the generalized CALE for the ADI model problem defined as

$$A^T X E + E^T X A = -C^T C, \tag{4.2}$$

associated with the generalized continuous-time LTI system

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{4.3}$$

Then, the original ADI scheme (4.1) can be redefined as

$$\begin{aligned} (A^T + \mu_i E^T) X_{i-\frac{1}{2}} &= -C^T C - X_{i-1}(A - \mu_i E), \\ (A^T + \mu_i E^T) X_i^T &= -C^T C - X_{i-\frac{1}{2}}^T (A - \mu_i E), \end{aligned} \tag{4.4}$$

where the shift parameters $\mu_i \in \mathbb{C}^-$ are allowed and the initial iteration is taken as $X_0 = {X_0}^T \in \mathbb{R}^{n\times n}$.

After elimination and simplification of the equations in (4.4), the single-step ADI scheme can be derived as

$$\begin{aligned} X_i = {} & (A^T + \mu_i E^T)^{-1}(A^T - \bar{\mu_i} E^T) X_{i-1} (A^T - \bar{\mu_i} E^T)^T (A^T + \mu_i E^T)^{-T} \\ & - 2\mathrm{Re}(\mu_i)(A^T + \mu_i E^T)^{-1} C^T C (A^T + \mu_i E^T)^{-T}. \end{aligned} \tag{4.5}$$

4.1.2 LRCF-ADI Method for Solving Generalized CALE

Let $Z_i \in \mathbb{R}^{n\times ip}$ is a low-rank Cholesky-factor of $X_i \in \mathbb{R}^{n\times n}$ such that $X_i = Z_i Z_i^T$. The matrix Z_i does not have to be a square matrix nor have a lower triangular structure. Then by applying the Cholesky factorization in (4.5), the low-rank ADI scheme can be found as

$$\begin{aligned} Z_0 &= O_{n\times p}, \\ Z_i Z_i^T &= \{(A^T + \mu_i E^T)^{-1}(A^T - \bar{\mu_i} E^T) Z_{i-1}\}\{((A^T + \mu_i E^T)^{-1}(A^T - \bar{\mu_i} E^T) Z_{i-1}\}^T \\ & \quad - 2\mathrm{Re}(\mu_i)\{(A^T + \mu_i E^T)^{-1} C^T\}\{(A^T + \mu_i E^T)^{-1} C^T\}^T \in \mathbb{R}^{n\times ip}. \end{aligned} \tag{4.6}$$

So, Z_i on the left side of (4.5) can be obtained by the combination of two factors on the right as

$$Z_i = \left[\sqrt{-2\mathrm{Re}(\mu_i)}(A^T + \mu_i E^T)^{-1}C^T \quad (A^T + \mu_i E^T)^{-1}(A^T - \bar{\mu}_i E^T)Z_{i-1}\right]. \tag{4.7}$$

Thus, the ADI algorithm can be reformulated in terms of Cholesky-factor Z_i of X_i and there will be no need to estimate or store X_i at each iteration as only Z_i is required. The initial form of low-rank ADI in terms of Cholesky-factor, which computes the Cholesky-factor Z_i of X_i as

$$\begin{aligned} Z_1 &= \sqrt{-2\mathrm{Re}(\mu_1)}(A^T + \mu_1 E^T)^{-1}C^T \in \mathbb{R}^{n\times p}, \\ Z_i &= \left[\sqrt{-2\mathrm{Re}(\mu_i)}(A^T + \mu_i E^T)^{-1}C^T \quad (A^T + \mu_i E^T)^{-1}(A^T - \bar{\mu}_i E^T)Z_{i-1}\right] \in \mathbb{R}^{n\times ip}. \end{aligned} \tag{4.8}$$

Following lemma depicts one of the vital aspects for the formulation of the LRCF-ADI iterates [84].

Lemma 4.1 (**Commutative Identity**). *For all matrices $M, N \in \mathbb{C}^{n\times n}$ and α, β, the matrices $(M \pm \alpha N)^{\pm 1}$ and $(M \pm \beta N)^{\pm 1}$ commute, provided the inverses exist.*

Consider the following assumption

$$\begin{aligned} \gamma_i &= \sqrt{-2\mathrm{Re}(\mu_i)}, \\ F_i &= (A^T + \mu_i E^T)^{-1}C^T, \\ G_i &= (A^T + \mu_i E^T)^{-1}(A^T - \bar{\mu}_i E^T), \\ H_{i,j} &= (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1}. \end{aligned} \tag{4.9}$$

Then the low-rank Cholesky-factor Z_i in the above scheme can be written as

$$Z_i = \left[\gamma_i F_i, \gamma_{i-1}G_i F_{i-1}, \gamma_{i-2}G_i G_{i-1}F_{i-2}, \cdots, \gamma_1 G_i G_{i-1}\cdots G_1 F_1\right]. \tag{4.10}$$

Using $G_i F_i = H_{i,j}F_i$, $G_i G_j = H_{i,j}H_{j,i}, \forall i, j$ and Lemma-(4.1), the equation (4.10) can be re-written as

$$Z_i = \left[\gamma_i F_i, \gamma_{i-1}H_{i-1,i}F_i, \gamma_{i-2}H_{i-2,i-1}H_{i-1,i}F_i, \cdots, \gamma_1 H_{1,2}\cdots H_{i-1,i}F_i\right]. \tag{4.11}$$

Algorithm 8: G-LRCF-ADI.

Input : E, A, C, i_{max} (number of iterations) and shift parameters $\{\mu_j\}_{j=1}^{i_{max}}$.
Output: Low-rank Cholesky-factor Z such that $X \approx ZZ^T$.

Consider $Z_0 = [\,]$.
for $i \leftarrow 1$ *to* i_{max} **do**
  if $i = 1$ **then**
    Solve $V_1 = (A^T + \mu_1 E^T)^{-1} C^T$.
  else
    Compute $V_i = V_{i-1} - (\mu_i + \bar{\mu}_{i-1})(A^T + \mu_i E^T)^{-1} E^T V_{i-1}$.
  end if
  Update $Z_i = \begin{bmatrix} Z_{i-1} & \sqrt{-2\mathrm{Re}(\mu_i)} V_i \end{bmatrix}$.
end for

Thus, considering $i \geq 1$ and reversing the order of the shift parameters LRCF-ADI iterations yields the form as

$$\begin{aligned} V_1 &= (A^T + \mu_1 E^T)^{-1} C^T, \\ Z_1 &= \gamma_1 V_1 = \sqrt{-2\mathrm{Re}(\mu_1)} (A^T + \mu_1 E^T)^{-1} C^T, \\ V_i &= H_{i-1,i} V_{i-1} = V_{i-1} - (\mu_i + \bar{\mu}_{i-1})(A^T + \mu_i E^T)^{-1} E^T V_{i-1}, \\ Z_i &= \begin{bmatrix} Z_{i-1} & \gamma_i V_i \end{bmatrix} = \begin{bmatrix} Z_{i-1} & \sqrt{-2\mathrm{Re}(\mu_i)} V_i \end{bmatrix}. \end{aligned} \tag{4.12}$$

Summary of the above process summarized in Algorithm-(8).

4.1.3 Modified Form of LRCF-ADI Method for Solving Generalized CALE

To find the modified form of the LRCF-ADI method for the generalized system, consider the following theorem.

Theorem 4.2 (**Modified LRCF-ADI Iterates**). *Assuming a set of adjustable shift parameters, for two subsequent block iterates V_i, V_{i+1} of Algorithm-(8) related to the pair of complex conjugated shifts $\{\mu_i, \mu_{i+1} := \bar{\mu}_i\}$ it holds*

$$V_{i+1} = \bar{V}_i + 2\delta_i Im(V_i), \tag{4.13}$$

with $\delta_i = \frac{Re(\mu_i)}{Im(\mu_i)}$ and iterates associated to real shifts are always purely real.

Proof. See [86] and references therein. □

Algorithm 9: Modified G-LRCF-ADI.

Input : E, A, C, i_{max} (number of iterations) and shift parameters $\{\mu_j\}_{j=1}^{i_{max}}$.
Output: Low-rank Cholesky-factor Z such that $X \approx ZZ^T$.

Consider $Z_0 = [\,]$.
for $i \leftarrow 1$ *to* i_{max} **do**
  if $i = 1$ **then**
    Solve $V_1 = (A^T + \mu_1 E^T)^{-1} C^T$.
  else
    Compute $V_i = V_{i-1} - (\mu_i + \bar{\mu}_{i-1})(A^T + \mu_i E^T)^{-1} E^T V_{i-1}$.
  end if
  if $Im(\mu_i) = 0$ **then**
    Update $Z_i = \begin{bmatrix} Z_{i-1} & \sqrt{-2\mu_i} V_i \end{bmatrix}$.
  else
    Assume $\gamma_i = \sqrt{-2\mathrm{Re}(\mu_i)}$, $\delta_i = \frac{\mathrm{Re}(\mu_i)}{\mathrm{Im}(\mu_i)}$,
    Update $Z_{i+1} = \begin{bmatrix} Z_{i-1} & \gamma_i(\mathrm{Re}(V_i) + \delta_i \mathrm{Im}(V_i)) & \gamma_i \sqrt{\delta_i^2 + 1}\mathrm{Im}(V_i) \end{bmatrix}$,
    Compute $V_{i+1} = \bar{V}_i + 2\delta_i\ \mathrm{Im}(V_i)$.
    $i = i + 1$
  end if
end for

Applying Theorem-(4.2), the following matrix can be obtained

$$\begin{bmatrix} V_i & V_{i+1} \end{bmatrix} = \begin{bmatrix} \sqrt{-2\mathrm{Re}(\mu_i)}(\mathrm{Re}(V_i) + \delta_i \mathrm{Im}(V_i)) & \sqrt{-2\mathrm{Re}(\mu_i)}\sqrt{\delta_i^2 + 1}\ \mathrm{Im}(V_i) \end{bmatrix}. \quad (4.14)$$

Then, for a pair of complex conjugate shifts at any iteration, the low-rank Cholesky-factor Z_i can be computed as

$$Z_{i+1} = \begin{bmatrix} Z_{i-1} & \sqrt{-2\mathrm{Re}(\mu_i)}(\mathrm{Re}(V_i) + \delta_i \mathrm{Im}(V_i)) & \sqrt{-2\mathrm{Re}(\mu_i)}\sqrt{\delta_i^2 + 1}\ \mathrm{Im}(V_i) \end{bmatrix}. \quad (4.15)$$

Summary of the above process summarized in Algorithm-(9).

4.1.4 Convergence Criteria and Recurrence Relations

The computation of residuals of the LRCF-ADI iterations can be achieved using a simplified technique implementing the shift parameters. This approach will be efficient for time and memory allocation.

Theorem 4.3 (**Residual of the ADI Iterations**). *The residual for CALE (4.2) at i-th step of LRCF-ADI method is of rank at most m and given by*

$$\mathcal{R}(Z_i) = A^T Z_i Z_i^T E + E^T Z_i Z_i^T A + C^T C = W_i W_i^T, \quad (4.16)$$

with $W_i = \left(\prod_{i=1}^{m} (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1}\right) C^T$, *such that* $\mu_i \notin \Lambda(A, E)$ *for all* i *then the rank is exactly* m.

Proof. Consider the residual of the CALE (4.2) in terms of the low-rank Cholesky-factor Z_i in the i-th iteration

$$\mathcal{R}(Z_i) = A^T Z_i Z_i^T E + E^T Z_i Z_i^T A + C^T C. \tag{4.17}$$

For all $\mu_i \notin \Lambda(A, E)$, the equation (4.2) is equivalent to the Stein equation

$$ZZ^T = (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1} Z_i Z_i^T (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-T} + Z_1 Z_1^T, \tag{4.18}$$

where, Z_1 is the first low-rank Cholesky-factor of the solution of (4.2) obtained with Algorithm-(9) and the complex Cayley type transformation $H_{i,j}$.

Using (4.18) and Lemma-5.2 & Lemma-5.3 in [94], (4.17) can be written as

$$\begin{aligned}
\mathcal{R}(Z_i) &= A^T (Z_i Z_i^T - ZZ^T) E + E^T (Z_i Z_i^T - ZZ^T) A, \\
&= \left(\prod_{i=1}^{m} (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1}\right) C^T C \left(\prod_{i=1}^{m} (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1}\right)^T, \\
&= W_i W_i^T,
\end{aligned} \tag{4.19}$$

with $W_i = \left(\prod_{i=1}^{m} (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1}\right) C^T$. □

Thus, for $W_i W_i^T \leq \tau$ the ADI iterations in the LRCF-ADI algorithm needs to be stopped, where τ is a given margin of tolerance.

Now, apply Lemma-(4.1) and Theorem-(4.3) to find V_i in terms of the residual factor W_{i-1} by the following relation

$$\begin{aligned}
V_i &= (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1} V_{i-1}, \\
&= (A^T + \mu_i E^T)^{-1} \left(\prod_{j=1}^{i-1} (A^T - \bar{\mu}_j E^T)(A^T + \mu_j E^T)^{-1}\right) C^T, \\
&= (A^T + \mu_i E^T)^{-1} W_{i-1}.
\end{aligned} \tag{4.20}$$

Algorithm 10: Modified G-LRCF-ADI (updated).

Input : E, A, C, τ (tolerance), i_{max} (number of iterations) and shift parameters $\{\mu_j\}_{j=1}^{i_{max}}$.

Output: Low-rank Cholesky-factor Z such that $X \approx ZZ^T$.

Consider $W_0 = C^T$, $Z_0 = [\,]$ and $i = 1$.
while $\|W_{i-1}W_{i-1}^T\| \geq \tau$ *or* $i \leq i_{max}$ **do**
  Solve $V_i = (A^T + \mu_i E^T)^{-1} W_{i-1}$.
  if $Im(\mu_i) = 0$ **then**
    Update $Z_i = \begin{bmatrix} Z_{i-1} & \sqrt{-2\mu_i} V_i \end{bmatrix}$,
    Compute $W_i = W_{i-1} - 2\mu_i E^T V_i$.
  else
    Assume $\gamma_i = \sqrt{-2\mathrm{Re}(\mu_i)}$, $\delta_i = \frac{\mathrm{Re}(\mu_i)}{\mathrm{Im}(\mu_i)}$,
    Update $Z_{i+1} = \begin{bmatrix} Z_{i-1} & \gamma_i(\mathrm{Re}(V_i) + \delta_i \mathrm{Im}(V_i)) & \gamma_i\sqrt{\delta_i^2 + 1}\ \mathrm{Im}(V_i) \end{bmatrix}$,
    Compute $W_{i+1} = W_{i-1} - 4\mathrm{Re}(\mu_i) E^T [\mathrm{Re}(V_i) + \delta_i \mathrm{Im}(V_i)]$.
    $i = i + 1$
  end if
  $i = i + 1$
end while

Then, using (4.20) the residual factor W_i can be derived in a recursive form as

$$\begin{aligned} W_i &= \left(\prod_{i=1}^{m} (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1} \right) C^T, \\ &= (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1} \left(\prod_{j=1}^{i-1} (A^T - \bar{\mu}_j E^T)(A^T + \mu_j E^T)^{-1} \right) C^T, \\ &= (A^T - \bar{\mu}_i E^T)(A^T + \mu_i E^T)^{-1} W_{i-1}, \\ &= \left[I - (\mu_i + \bar{\mu}_i) E^T (A^T + \mu_i E^T)^{-1} \right] W_{i-1}, \\ &= W_{i-1} - 2\mathrm{Re}(\mu_i) E^T V_i. \end{aligned} \tag{4.21}$$

In the case of real setting, $\mu_{i+1} := \bar{\mu}_i$ needs to be considered and using Theorem-(4.2) the following form can be defined

$$\begin{aligned} W_{i+1} &= W_i - 2\mathrm{Re}(\mu_{i+1}) E^T V_{i+1}, \\ &= W_{i-1} - 2\mathrm{Re}(\mu_i) E^T V_i - 2\mathrm{Re}(\mu_{i+1}) E^T V_{i+1}, \\ &= W_{i-1} - 2\mathrm{Re}(\mu_i) E^T \left[V_i + \bar{V}_i + 2\delta_i \mathrm{Im}(V_i) \right], \\ &= W_{i-1} - 4\mathrm{Re}(\mu_i) E^T \left[\mathrm{Re}(V_i) + \delta_i \mathrm{Im}(V_i) \right]. \end{aligned} \tag{4.22}$$

The summary is given in Algorithm-(10).

4.2 LRCF-ADI Based Kleinman-Newton Method for Solving Structured Generalized CARE

Since the LRCF-ADI method is mainly derived for the generalized CALE, the LRCF-ADI based Kleinman-Newton method needs to be divided into two steps. Initially, the generalized CARE needs to be converted to the generalized CALE and then can be solved by Kleinman-Newton method. At each Newton iterations, the LRCF-ADI technique needs to be applied for once.

4.2.1 Conversion from CARE to CALE for the Structured Generalized System

The structured generalized CARE corresponding to the system (3.10) is

$$\mathcal{A}^T X\mathcal{E} + \mathcal{E}^T X\mathcal{A} - \mathcal{E}^T X\mathcal{B}R^{-1}\mathcal{B}^T X\mathcal{E} + \mathcal{C}^T\mathcal{C} = O. \tag{4.23}$$

The residual $\mathcal{R}(X)$ of the CARE (4.23) is

$$\mathcal{R}(X) = \mathcal{A}^T X\mathcal{E} + \mathcal{E}^T X\mathcal{A} - \mathcal{E}^T X\mathcal{B}R^{-1}\mathcal{B}^T X\mathcal{E} + \mathcal{C}^T\mathcal{C}. \tag{4.24}$$

From the Fre'chet derivative, we have

$$\mathcal{R}'(Y) = (\mathcal{A} - \mathcal{B}\mathcal{B}^T X\mathcal{E})^T Y\mathcal{E} + \mathcal{E}^T Y(\mathcal{A} - \mathcal{B}\mathcal{B}^T X\mathcal{E}). \tag{4.25}$$

Again, consider the Newton iteration and apply (4.25), we have

$$\mathcal{R}'(\Delta X_i) + \mathcal{R}(X_i) = O. \tag{4.26}$$

Assuming $\Delta X_i = X_{i+1} - X_i$ to (4.26) and simplification of (4.26) gives

$$(\mathcal{A} - \mathcal{B}\mathcal{B}^T X_i\mathcal{E})^T X_{i+1}\mathcal{E} + \mathcal{E}^T X_{i+1}(\mathcal{A} - \mathcal{B}\mathcal{B}^T X_i\mathcal{E}) = -\mathcal{C}^T\mathcal{C} - \mathcal{E}^T X_i\mathcal{B}R^{-1}\mathcal{B}^T X_i\mathcal{E}. \tag{4.27}$$

Now consider $\tilde{\mathcal{A}}_i = \mathcal{A} - \mathcal{B}\mathcal{B}^T X_i\mathcal{E}$ and $\mathcal{W}_i = \begin{bmatrix}\mathcal{C}^T & \mathcal{E}^T X_i\mathcal{B}\end{bmatrix}$, then equation (4.27) reduces to a generalized CALE such as

$$\tilde{\mathcal{A}}_i^T X_{i+1}\mathcal{E} + \mathcal{E}^T X_{i+1}\tilde{\mathcal{A}}_i = -\mathcal{W}_i\mathcal{W}_i^T. \tag{4.28}$$

Algorithm 11: G-KN.

Input : $\mathcal{E}, \mathcal{A}, \mathcal{B}, \mathcal{C}$ and X_0 (initial assumption).
Output: Approximate solution X and feedback matrix K.

```
while i ≤ i_max do
    Compute Ã_i = A − BB^T X_i E and W_i = [C^T  E^T X_i B].
    For X_{i+1}, solve Ã_i^T X_{i+1} E + E^T X_{i+1} Ã_i = −W_i W_i^T.
    Compute K_{i+1} = B^T X_{i+1} E.
end while
```

The generalized CALE (4.28) can be solved for X_{i+1} by any conventional method and the corresponding feedback matrix $K_{i+1} = \mathcal{B}^T X_{i+1} \mathcal{E}$ can be estimated. The whole mechanism for solving generalized CARE is called the Kleinman-Newton method. The summary of the method is given in Algorithm-11.

4.2.2 LRCF-ADI Based Kleinman-Newton Method (KN-LRCF-ADI) for Solving Converted Generalized CALE

In each iteration of Algorithm-11, the generalized CALE needs to be solved for once and there are several techniques available to do it. In his Ph.D. thesis [95], **Patrick** discussed LRCF-ADI approaches (Algorithm-3.2 Chapter-3 and Algorithm-6.2 in Chapter-6) for solving generalized CALE derived from generalized CARE in the iterative loops of the Kleinman-Newton algorithm. Now, the above mechanisms need to be implemented for the index-1 descriptor system. For the adjustment, some modifications are required as given below.

For the unstable index-1 descriptor systems, the initial feedback matrix K_0 needs to be introduced and instead of $(\tilde{\mathcal{A}}^{(i)}, \mathcal{E})$, corresponding shift parameters can be computed from the eigen-pair $(\tilde{\mathcal{A}}^{(i)} - \mathcal{B}K_0, \mathcal{E})$. The sparse form of the desired eigen-pair can be structured as

$$\begin{aligned}(\tilde{\mathcal{A}}^{(i)} - \mathcal{B}K_0, \mathcal{E}) &= ((\mathcal{A} - \mathcal{B}K_0) - \mathcal{B}\mathcal{B}^T(Z^{(i)}(Z^{(i)})^T)\mathcal{E}, \mathcal{E}), \\ &= \left(\begin{bmatrix} (J_1 - B_1K_0) - B_1B_1^T(Z^{(i)}(Z^{(i)})^T)E_1 & J_2 \\ (J_3 - B_2K_0) - B_2B_1^T(Z^{(i)}(Z^{(i)})^T)E_1 & J_4 \end{bmatrix}, \begin{bmatrix} E_1 & O \\ O & O \end{bmatrix} \right).\end{aligned} \tag{4.29}$$

To find $V_j^{(i)}$, in each LRCF-ADI (inner) iteration a shifted linear system is needs to be solved as

$$\begin{aligned} ((\tilde{\mathcal{A}}^{(i)} - \mathcal{B}K_0) + \mu_j^{(i)}\mathcal{E})^T V_j^{(i)} &= \mathcal{W}_{j-1}^{(i)}, \\ or, ((\mathcal{A} - \mathcal{B}K_0) - \mathcal{B}\mathcal{B}^T(Z_{j-1}^{(i)}(Z_{j-1}^{(i)})^T)\mathcal{E} + \mu_j^{(i)}\mathcal{E})^T V_j^{(i)} &= \mathcal{W}_{j-1}^{(i)}. \end{aligned} \tag{4.30}$$

Thus, $V_j^{(i)}$ can be obtained from the following sparse form of the shifted linear system

$$\begin{aligned} &\begin{bmatrix} (J_1 - B_1K_0) - B_1B_1^T(Z_{j-1}^{(i)}(Z_{j-1}^{(i)})^T)E_1 + \mu_j E_1 & J_2 \\ (J_3 - B_2K_0) - B_2B_1^T(Z_{j-1}^{(i)}(Z_{j-1}^{(i)})^T)E_1 & J_4 \end{bmatrix}^T \begin{bmatrix} V_j^{(i)} \\ * \end{bmatrix} \\ &\qquad = \begin{bmatrix} C_1^T & E_1^T(Z_{j-1}^{(i)}(Z_{j-1}^{(i)})^T)B_1 \\ C_2^T & O \end{bmatrix}. \end{aligned} \tag{4.31}$$

The feedback matrix $K^{(i)} = \mathcal{B}^T X^{(i)}\mathcal{E} = \mathcal{B}^T(Z^{(i)}(Z^{(i)})^T)\mathcal{E}$ needs to be computed in each LRCF-ADI (inner) iteration and the optimal feedback matrix $K^o = K^{(i_{max})}$ needs to be stored after the final Newton (outer) iteration. The summary of the desired LRCF-ADI based Kleinman-Newton method is given in the Algorithm-(12).

Finally, considering $\mathcal{A}_s = \mathcal{A} - \mathcal{B}K^o$ the target system (3.10) can be optimally stabilized as the system (3.30).

Algorithm 12: G-KN-LRCF-ADI.

Input : $E_1, J_1, J_2, J_3, J_4, B_1, B_2, C_1, C_2, K_0$ (initial feedback matrix), and τ (tolerance).

Output: Low-rank Cholesky-factor Z such that $X \approx ZZ^T$ and optimal feedback matrix K^o.

for $i \leftarrow 1$ *to* i_{max} **do**
- Choose $Z_0^{(i)} = [\,]$, $K_0^{(i)} = K_0$ and $j = 0$.
- Assume $\mathcal{W}_0^{(i)} = \begin{bmatrix} C_1^T & (K^{(i-1)})^T \\ C_2^T & O \end{bmatrix}$.
- Compute adaptive shifts $\left\{\mu_1^{(i)}, ..., \mu_J^{(i)}\right\}$ from the eigenpair defined in (4.29).
- **while** $\left(\|\mathcal{W}_j^{(i)}\|^2 > \tau \|\mathcal{W}_0^{(i)}\|^2\right)$ **do**
  - $j = j + 1$
  - Solve the linear system (4.31) for $V_j^{(i)}$.
  - **if** $Im(\mu_j^{(i)}) = 0$ **then**
    - Update $Z_j^{(i)} = \left[Z_{j-1}^{(i)} \quad \sqrt{-2\mu_i} V_j^{(i)}\right]$,
    - Compute $\mathcal{W}_j^{(i)} = \mathcal{W}_{j-1}^{(i)} - 2\mu_j^{(i)} E_1^T V_j^{(i)}$,
    - Compute $K_j^{(i)} = K_{j-1}^{(i)} - 2\mu_j^{(i)} (B_1 - J_2 {J_4}^{-1} B_2)^T V_j^{(i)} (V_j^{(i)})^T E_1$.
  - **else**
    - Assume $\gamma_j^{(i)} = \sqrt{-2\mathrm{Re}(\mu_j^{(i)})}$, $\quad \beta_j^{(i)} = \frac{\mathrm{Re}(\mu_j^{(i)})}{\mathrm{Im}(\mu_j^{(i)})}$, $\delta_j^{(i)} = \mathrm{Re}(V_j^{(i)}) + \beta_j^{(i)} \mathrm{Im}(V_j^{(i)})$,
    - Compute $Z_d = \left[\gamma_j^{(i)} \delta_j^{(i)} \quad \gamma_j^{(i)} \sqrt{(\beta_j^{(i)})^2 + 1}\ \mathrm{Im}(\mu_j^{(i)})\right]$,
    - Update $Z_{j+1}^{(i)} = \left[Z_{j-1}^{(i)} \quad Z_d\right]$,
    - Compute $\mathcal{W}_{i+1}^{(i)} = \mathcal{W}_{i-1}^{(i)} - 4\mathrm{Re}(\mu_j^{(i)}) E_1^T \delta_j^{(i)}$,
    - Compute $K_{j+1}^{(i)} = K_{j-1}^{(i)} + (B_1 - J_2 {J_4}^{-1} B_2)^T Z_d {Z_d}^T E_1$,
    - $j = j + 1$.
  - **end if**
- **end while**
- Update $Z^{(i)} = Z_j^{(i)}$ and $K^{(i)} = K_j^{(i)}$.

end for

Chapter 5

Numerical Result

In this chapter we analyze the practical benefits of the RKSM and LRCF-ADI based Kleinaman-Newton method, which are discussed in Chapter-3 and Chapter-4 respectively. We illustrate the numerical results to asses the applicability and compatibility of the proposed techniques. The vital aspects of the numerical computation are memory allocation and the rate of convergence. Also, the robustness of the simulation process is under the focus.

To justify the efficiency and accuracy of the proposed methods, we implement the proposed methods to the real-world power models.

5.1 Brazilian Inter-connected Power System (BIPS) Models

Power system models are an essential part of engineering fields that consists of simulations based on power systems and grid networks. The computation required in order to analyze electrical power systems by means of mathematical models utilizing real-time data. There are a number of applications of the power system model, i.e., electric power generation, utility transmission and distribution, railway power systems and industrial power generation [96]. The power system models can be represented by the DAEs with appropriate constraints.

The Brazilian Inter-connected Power System (BIPS) models are one of the most convenient examples of the power system models [97]. The models in BIPS systems can be

represented as the block matrix representation of DAEs as follows

$$\begin{aligned}\underbrace{\begin{bmatrix} E_1 & O \\ O & O \end{bmatrix}}_{E}\underbrace{\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix}}_{\dot{x}(t)} &= \underbrace{\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}}_{A}\underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}}_{B} u(t), \\ y(t) &= \underbrace{\begin{bmatrix} C_1 & C_2 \end{bmatrix}}_{C}\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + D_a u(t).\end{aligned} \tag{5.1}$$

where $E_1 \in \mathbb{R}^{n_1}$, $J_4 \in \mathbb{R}^{n_2}$ with $n_1 + n_2 = n$ are non-singular sub-matrices and other sub-matrices are sparse in appropriate dimensions. As $n_1 < n$, a number of finite eigenvalues in the spectrum $\Lambda(A, E)$ lie in the $\mathbb{C}^-$. The above special structure of the power model (5.1) provides an index-1 descriptor system.

Since the inverse of E does not exist, the formulation of the matrix equations is not as conventional as before. Eliminating the algebraic variables in (5.1), the system can be reduced to the generalized state-space system as

$$\begin{aligned}\mathcal{E}\dot{x}(t) &= \mathcal{A}x(t) + \mathcal{B}u(t), \\ y(t) &= \mathbb{C}x(t) + \mathcal{D}u(t).\end{aligned} \tag{5.2}$$

Here, the matrices formed in terms of `Schur` compliments as

$$\begin{aligned}&x = x_1,\ \mathcal{E} = E_1,\ \mathcal{A} = J_1 - J_2{J_4}^{-1}J_3, \\ &\mathcal{B} = B_1 - J_2{J_4}^{-1}B_2,\ \mathbb{C} = C_1 - C_2{J_4}^{-1}J_3,\ \mathcal{D} = D_a - C_2{J_4}^{-1}B_2.\end{aligned}$$

Here, our aim is to apply the proposed methods to some of the BIPS model derived from the test systems BIPS98 and BIPS07 [98, 99]. The models considered for the numerical computations are all of the index-1 descriptor systems. The following Table-5.1 provides the details about the models [6].

The models $mod-606$, $mod-1998$, $mod-2476$ and $mod-3078$ have the unstable eigenvalues, whereas the models $mod-1142$, $mod-1450$ and $mod-1963$ have stable eigenvalues [100]. Here name of the models are considered according to their number of states. In the numerical computation we will target the unstable models only.

The stability of the target models is investigated and the unstable models are stabilized through the Riccati based feedback stabilization process. The proposed methods are employed to find the solution of Riccati equation arising from the BIPS models and

[6] https://sites.google.com/site/rommes/software

Table. 5.1. Structure of the Models derived from BIPS test systems

Test systems	BIPS98				BIPS07		
Dimensions	7135	9735	11265	13545	15066	16861	21128
States	606	1142	1450	1963	1998	2476	3078
Algebraic variables	6529	8593	9815	11582	13068	14385	18050
Inputs	4	4	1	4	4	4	4
Outputs	4	4	1	4	4	4	4

corresponding feedback matrices are generated for system stabilization. Also, initial Bernoulli feedback stabilization is implemented for the models having unstable eigenvalues.

All the results have been achieved using the MATLAB 8.5.0 (R2015a) on a Windows machine having Intel-Xeon Silver 4114 CPU 2.20 GHz clock speed, 2 cores each and 64 GB of total RAM.

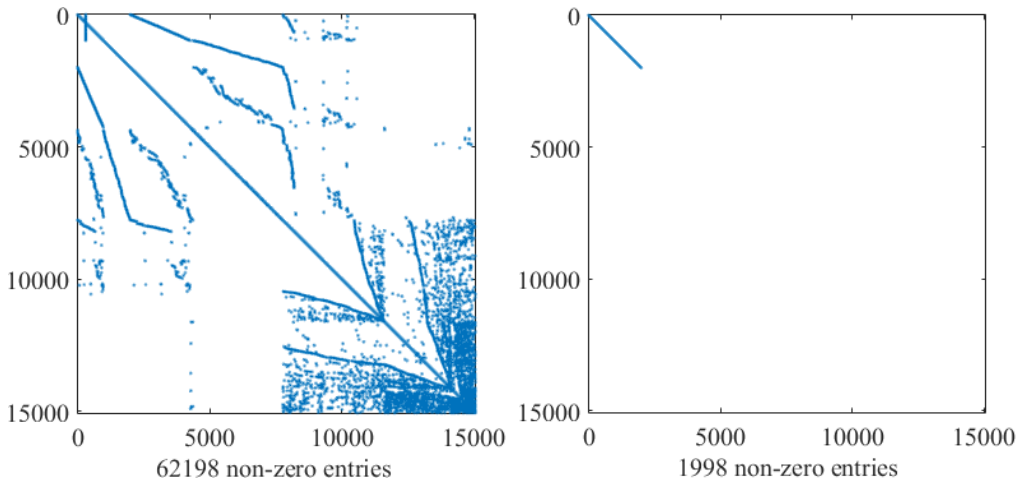
5.2 Graphical Comparison of the Properties of Original and Structured System

Since the target models are derived from the index-1 descriptor systems, they can not be simulated by the conventional approaches. The target models need to be structured according to the techniques discussed in the Section-(3.2). The analytical validation of the structured systems has been done in the mentioned section. In this section, the graphical validation will be discussed.

Since all the models in the BIPS test systems have the same pattern, to reduce the time and volume of the work we will show the properties of the $mod-1998$ only for the graphical validation.

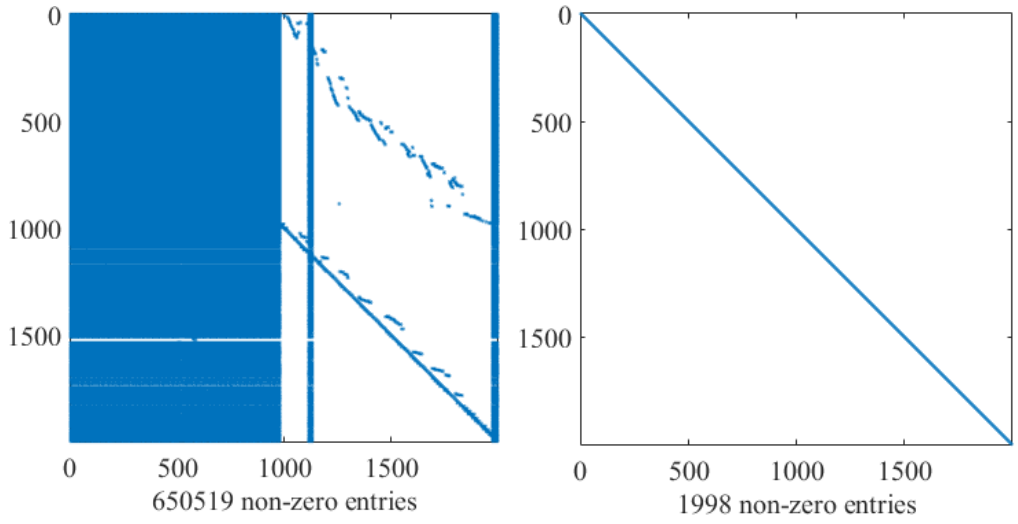
Figure-(5.1) depicts the pattern of the matrices A and E in the original system with dimension 13068. Figure-(5.2) depicts the pattern of the matrices A and E in the structured system with dimension 1998. The matrices A and E in the original system was sparse, whereas the matrix A in the structured system is dense. As mentioned in the Sub-section (3.3.3), the dense form of the matrix A in the structured system will not be computed explicitly.

The pattern preservation of the structured model for $mod-1998$ illustrates in the Figure-(5.3). In the Figure-(5.3a), the similarity of the transfer functions of the original system and structured system are displayed, whereas the Figure-(5.3b) and Figure-(5.3c) are



(a) Matrix A (b) Matrix E

Figure. 5.1. Sparse form of the matrices A and E

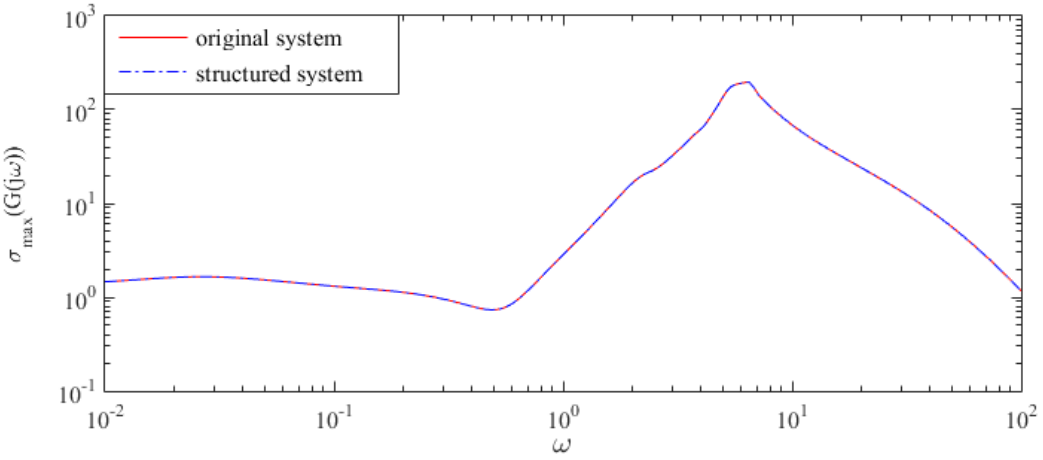


(a) Matrix A (b) Matrix E

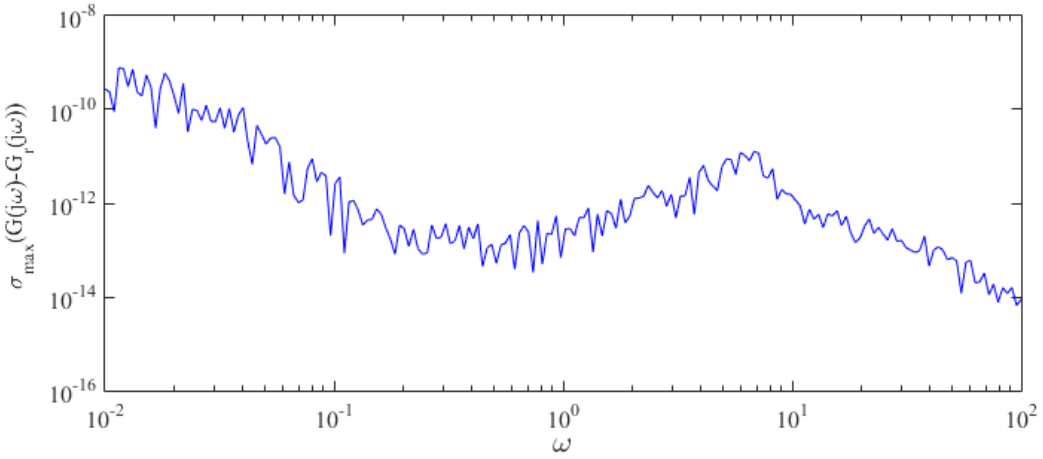
Figure. 5.2. Dense form of the matrices A and E

represented the absolute error and the relative error between the original system and structured system for $mod - 1998$.

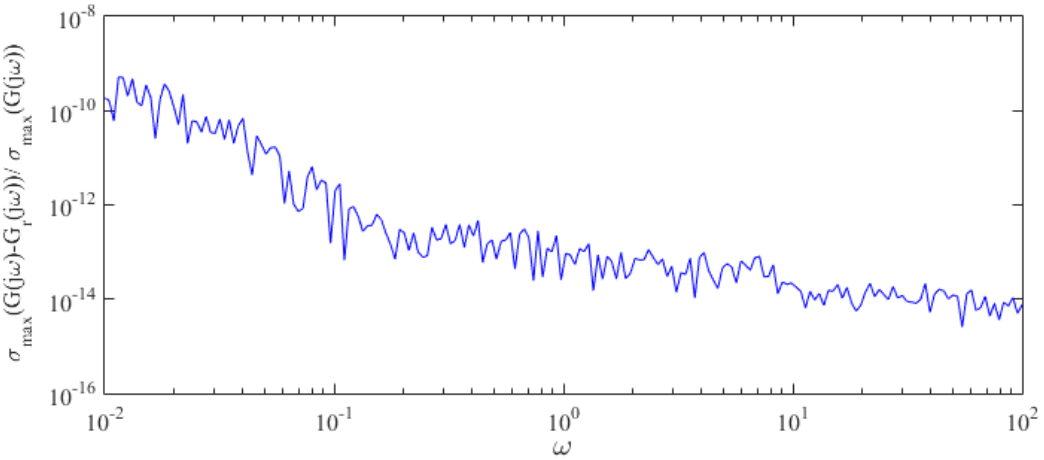
From the figurative evidence, we can conclude that the structured system is the proper representative of the original system and can be used for further manipulations.



(a) Transfer function (sigma plots)



(b) Absolute error



(c) Relative error

Figure. 5.3. Comparisons of the original system and the structured system

Table. 5.2. Results of RKSM applied BIPS models

Model	No of iterations	Tolerance	Space dimension	Numerical rank	CPU time (second)
$mod-606$	100	10^{-10}	400	195	1.8079×10^2
$mod-1998$	200	10^{-10}	800	266	1.4121×10^3
$mod-2476$	248	10^{-10}	992	265	3.0553×10^3
$mod-3078$	257	10^{-5}	1028	295	3.0065×10^3

5.3 Comparison of the RKSM and KN-LRCF-ADI Methods

In this section, we discuss the comparison of the numerical results achieved by applying RKSM and KN-LRCF-ADI methods for the unstable BIPS models.

The CAREs arising from the models $mod-606$, $mod-1998$ and $mod-2476$ are efficiently solved and stabilized the corresponding models by both RKSM and KN-LRCF-ADI techniques. As the model $mod-3078$ is semi-stable, the computation of CARE derived from this model is not possible by LRCF-ADI techniques but by the RKSM approach the model $mod-3078$ successfully stabilized and the numerical result for the model $mod-3078$ is investigated for RKSM only.

The Table-5.2 depicts the numerical results of the stabilization process via RKSM for the unstable BIPS models and various properties of the stabilized systems are illustrated, whereas the Table-5.3 displays the several modes of ADI techniques in KN-LRCF-ADI method for stabilizing the unstable BIPS models including characteristics of the stabilized models.

In both of the tables Table-5.2 and Table-5.3 we analyze the same features of the stabilized BIPS models so that we can easily compare the efficiency and robustness of the proposed methods.

From the tables Table-5.2 and Table-5.3, it can be said that the proposed RKSM approach has quick convergence ability and occupy very small solution spaces to provide efficient solutions of the CAREs. In contrast LRCF-ADI based Kleinman-Newton has several approaches for finding the solutions of the CAREs, where almost all of the approaches required higher computation time.

Also, there are deviations of the numerical ranks of the factored solution of CAREs in the Kleinman-Newton approaches and in all of the cases RKSM provides significantly better results.

Table. 5.3. Results of KN-LRCF-ADI applied BIPS models

Model	No of Newton iterations	Tolerance	Total iterations	Numerical rank	CPU time (second)
$mod-606$	5	10^{-5}	311	481	2.7408×10^2
		10^{-10}	544	953	7.2103×10^2
	10	10^{-5}	508	473	4.1036×10^2
		10^{-10}	853	969	7.9149×10^2
$mod-1998$	5	10^{-5}	277	663	2.2981×10^3
		10^{-10}	485	1201	5.8261×10^3
	10	10^{-5}	514	497	3.2732×10^3
		10^{-10}	1003	1417	1.1709×10^4
$mod-2476$	5	10^{-5}	254	473	3.1296×10^3
		10^{-10}	363	937	5.7464×10^3
	10	10^{-5}	366	441	3.0172×10^3
		10^{-10}	698	849	9.5013×10^4

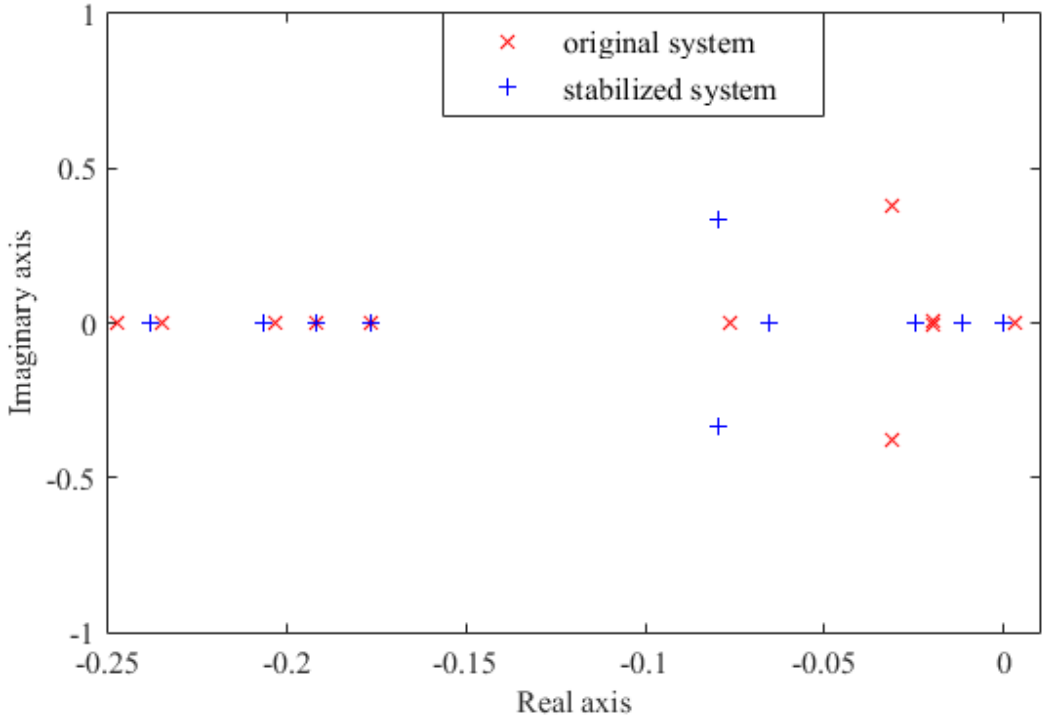
5.4 Graphical Comparisons of Stabilization of the Unstable Systems

In this section, we show the stabilization of the unstable BIPS models in terms of eigenvalues and step-responses. The comparative discussion for the RKSM and KN-LRCF-ADI are illustrated as well.

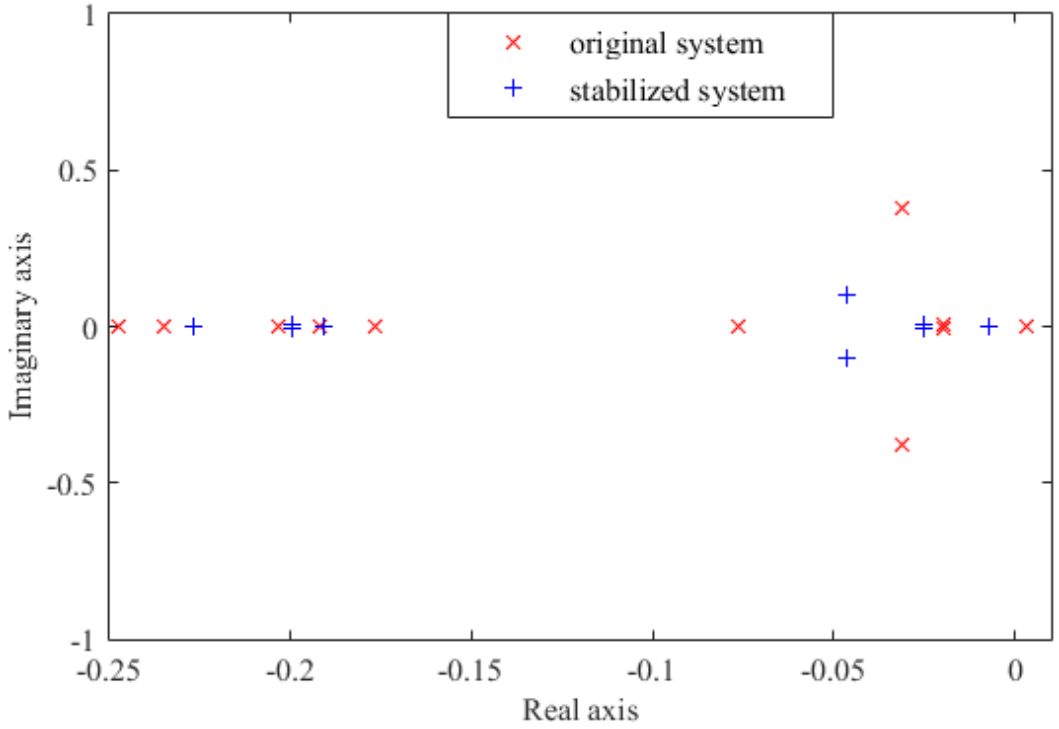
From the sub-figures in Figure-5.4, Figure-5.5 and Figure-5.6, it can be concluded that both the RKSM and LRCF-ADI based Kleinman-Newton techniques have adequate efficiency to stabilize the unstable index-1 descriptor systems by closed-loop structures via Riccati based feedback stabilization. But the Figure-5.7 illustrates the applicability of the RKSM for the semi-stable index-1 descriptor system, whereas LRCF-ADI based techniques are ineffective in this case.

Here, for the simulation tool capacity and visual convenience, the magnified view of the eigenspaces are considered.

The investigation of the figures from Figure-5.8 to Figure-5.13 consists of the step-responses of the unstable index-1 descriptor systems for several input-output relations to compare the RKSM and LRCF-ADI based Kleinman-Newton approaches via of the system stabilization. From those figures, it is evident that the Riccati based feedback stabilization by RKSM is suitably robust. Contrariwise, though sometimes the Kleinman-Newton approaches provide very good accuracy it has some scattered behaviors.

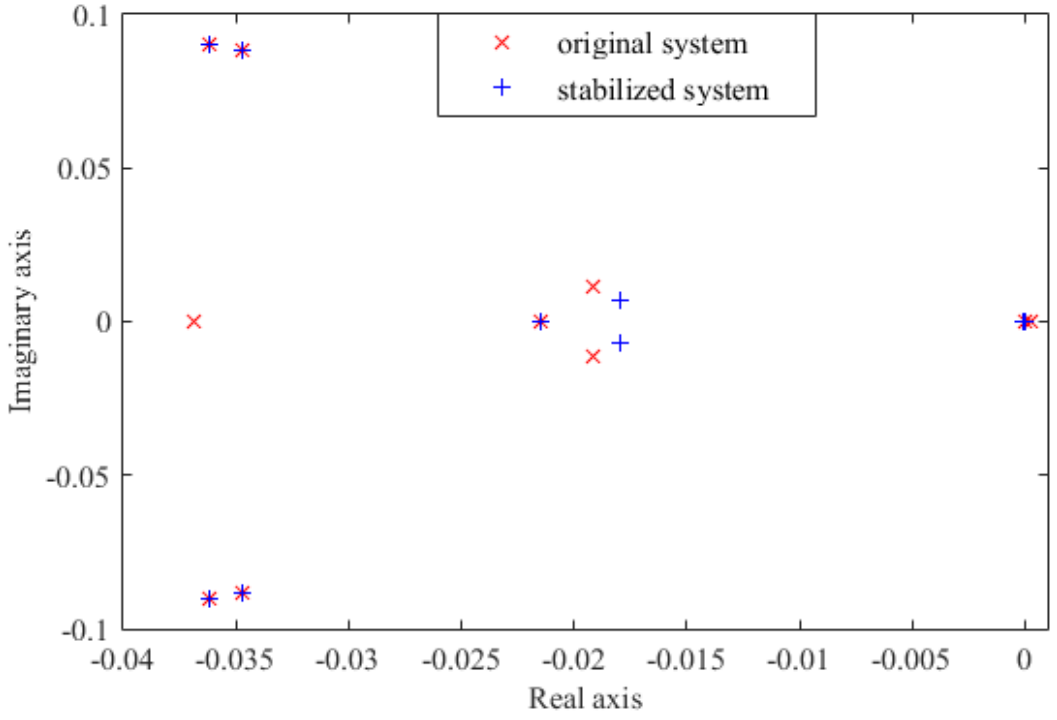


(a) Stabilized by RKSM

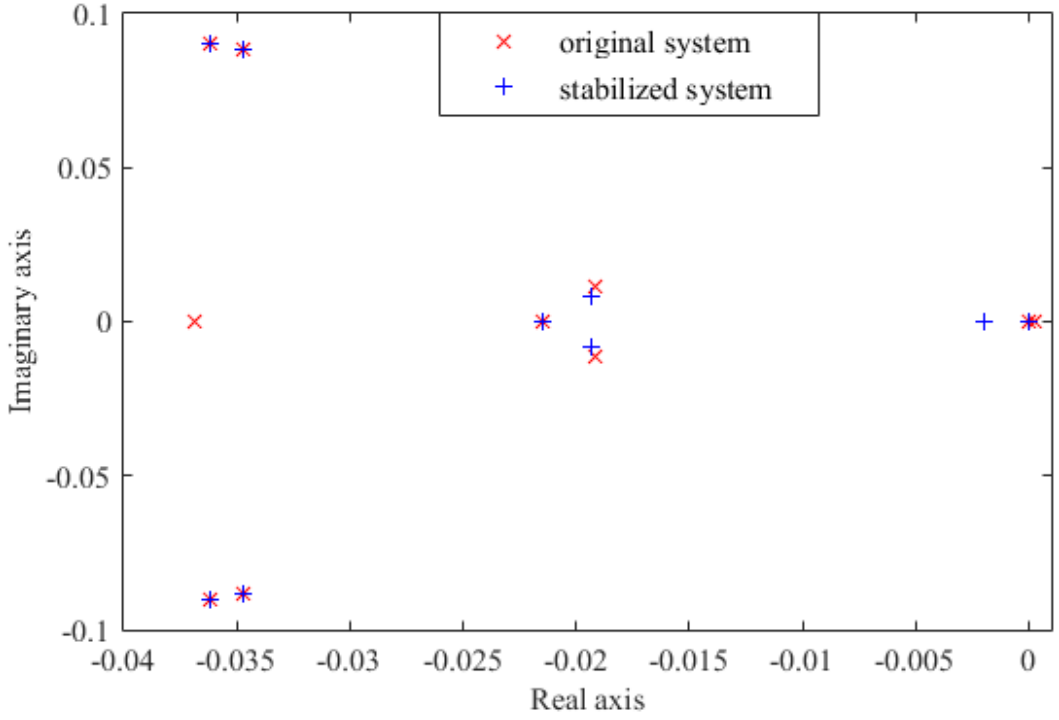


(b) Stabilized by KN-LRCF-ADI

Figure. 5.4. Comparisons of the eigenvalues for the model $mod - 606$

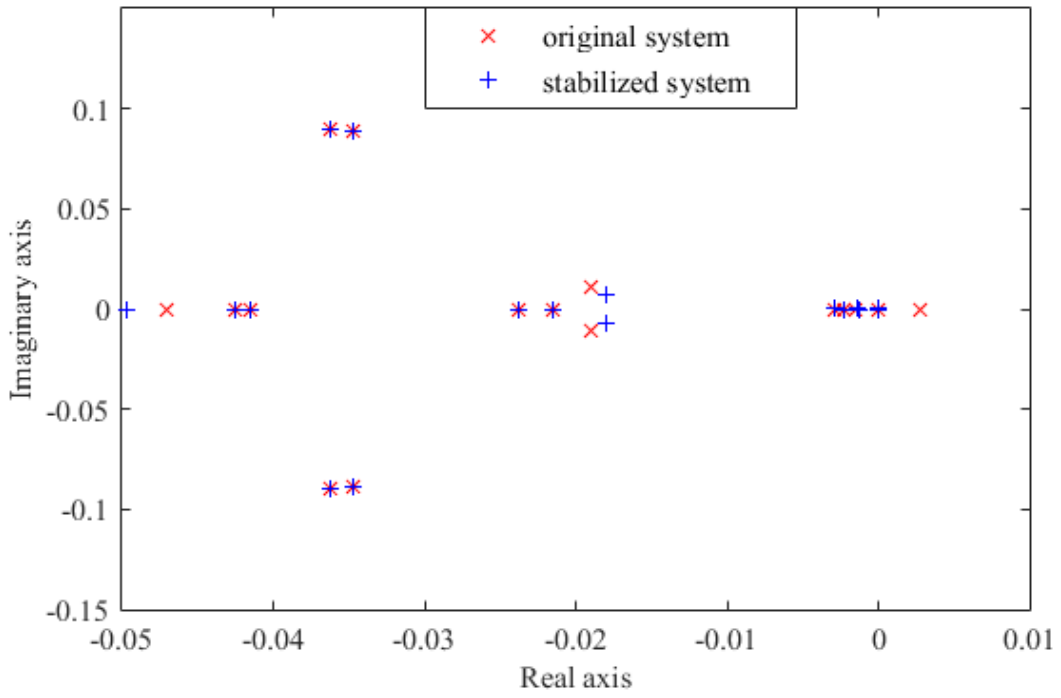


(a) Stabilized by RKSM

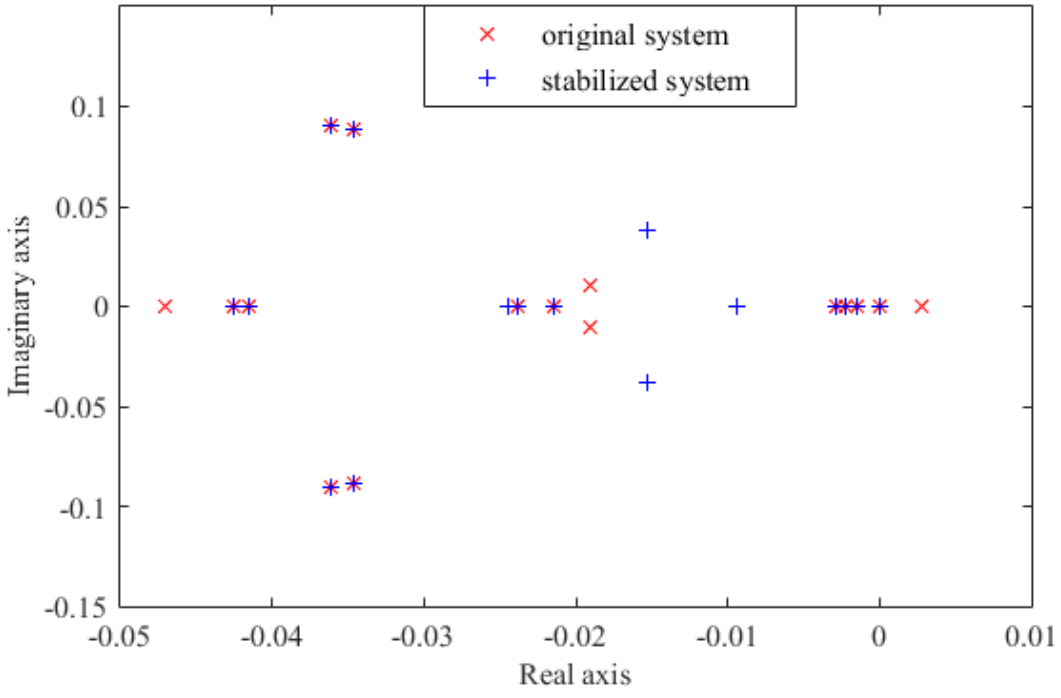


(b) Stabilized by KN-LRCF-ADI

Figure. 5.5. Comparisons of the eigenvalues for the model $mod - 1998$



(a) Stabilized by RKSM



(b) Stabilized by KN-LRCF-ADI

Figure. 5.6. Comparisons of the eigenvalues for the model $mod - 2476$

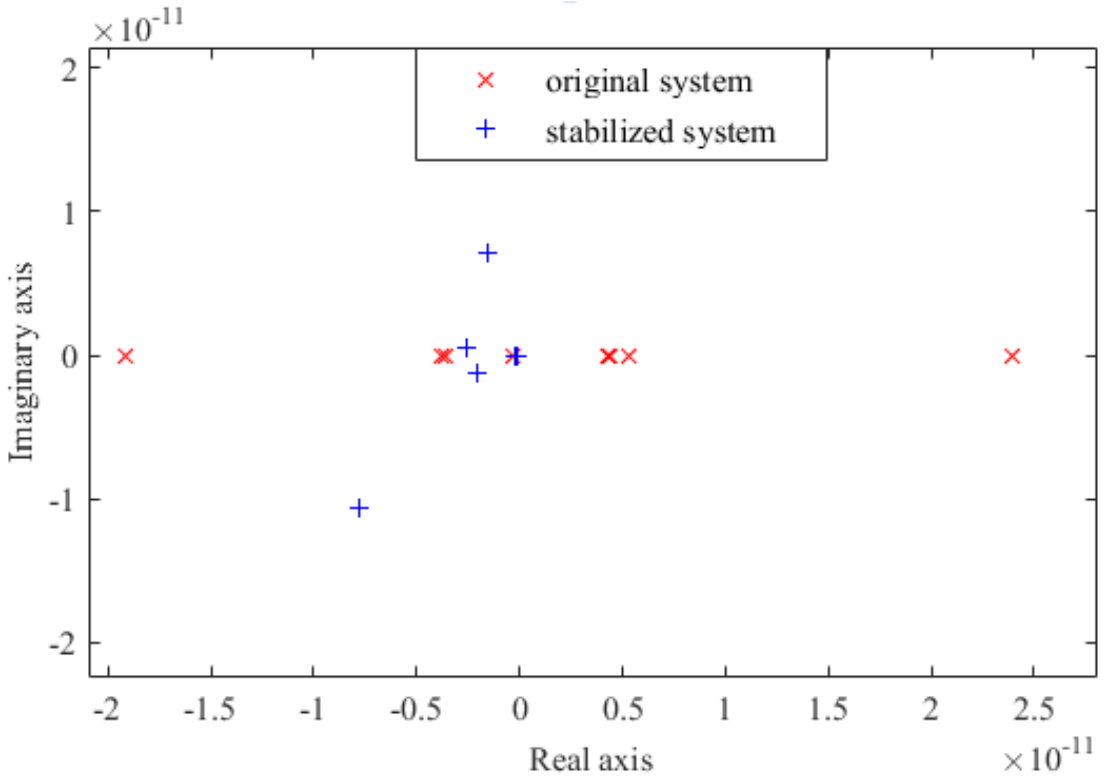
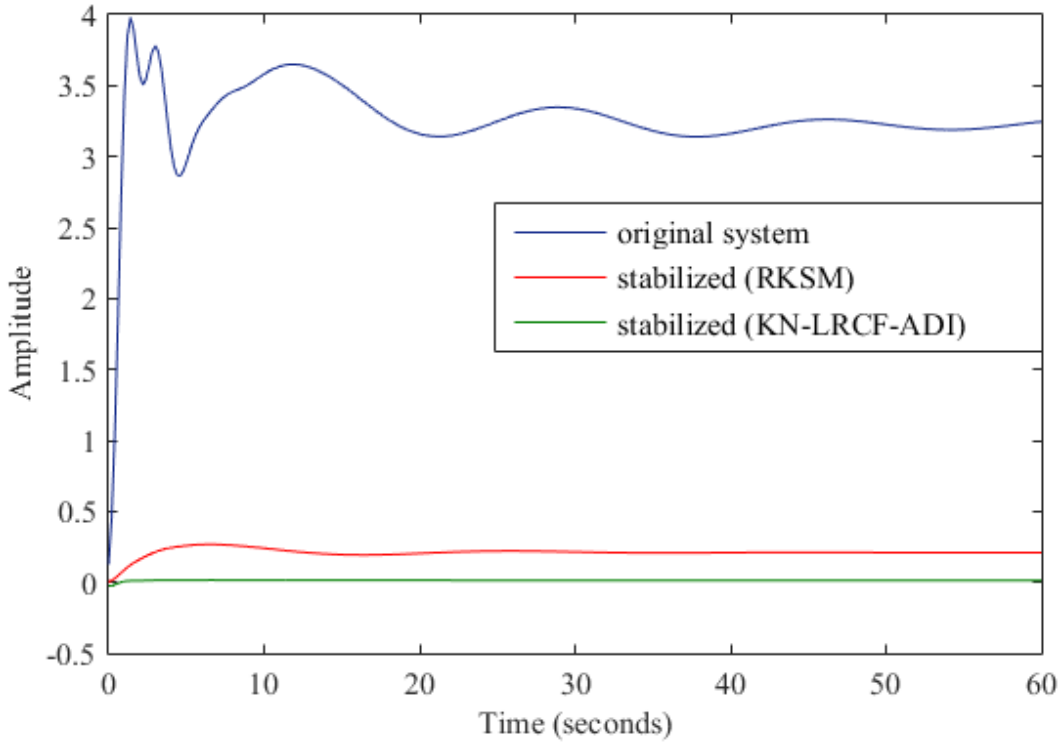


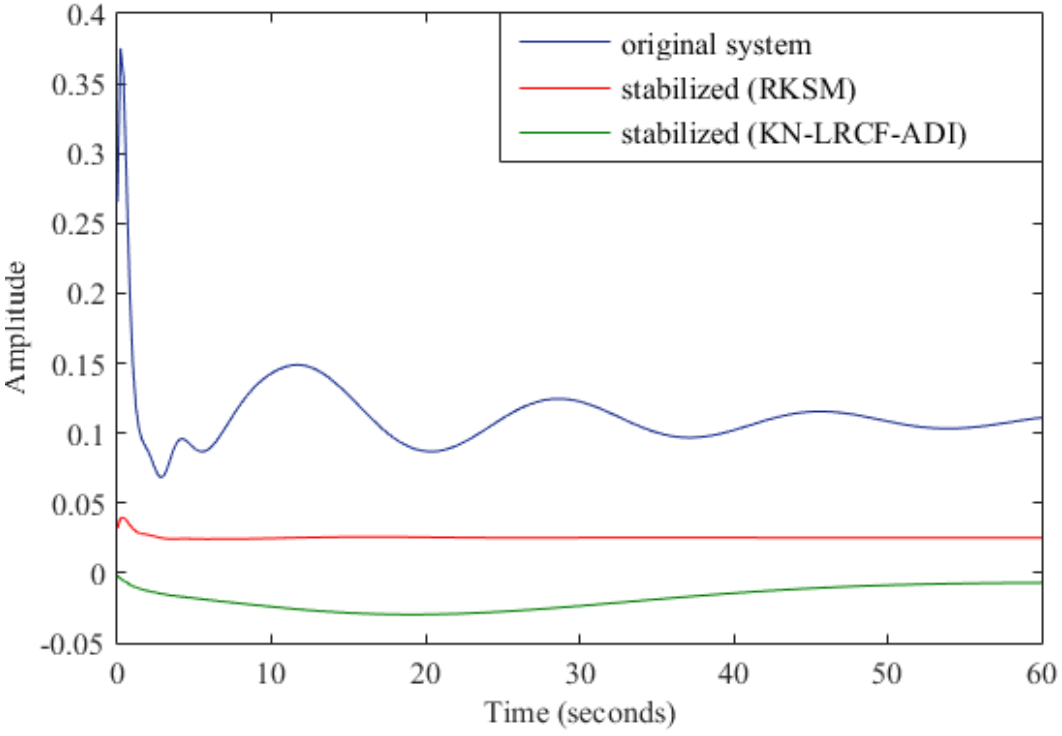
Figure. 5.7. Comparisons of the eigenvalues (stabilized by RKSM) for the model $mod - 3078$

Moreover, the Figure-5.14 and Figure-5.15 show the applicability of the RKSM technique for the Riccati based feedback stabilization for the semi-stable index-1 descriptor systems.

It to be noted that for the effective comparison step-responses of the only significant input-output relations are investigated.

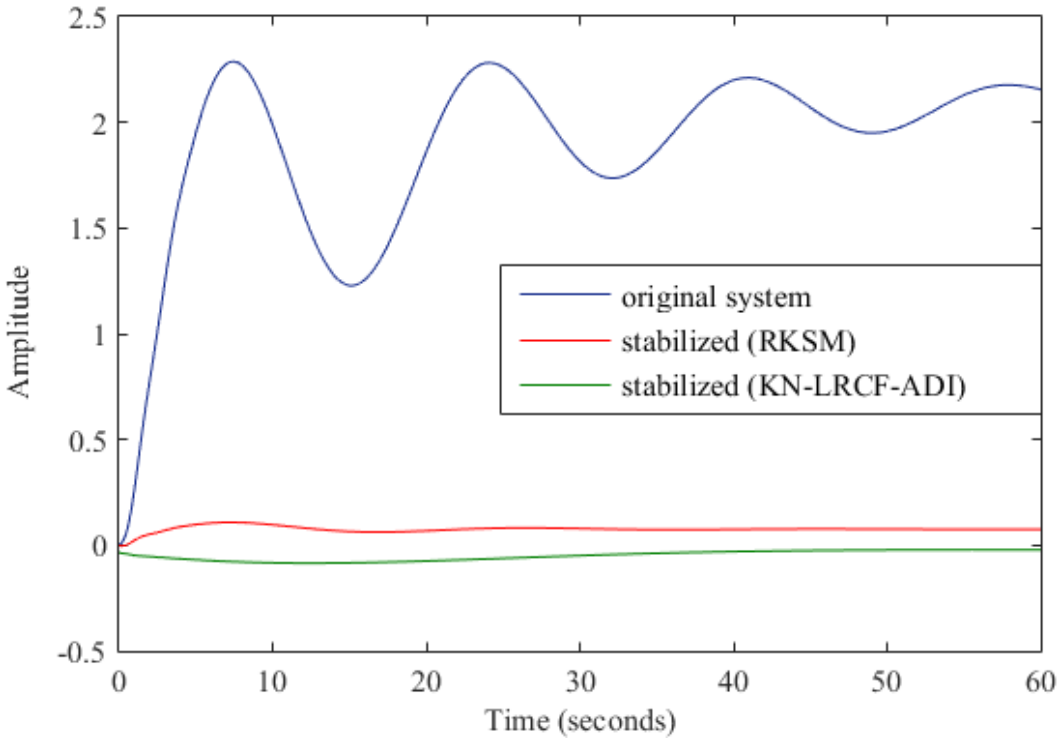


(a) First-input/Third-output

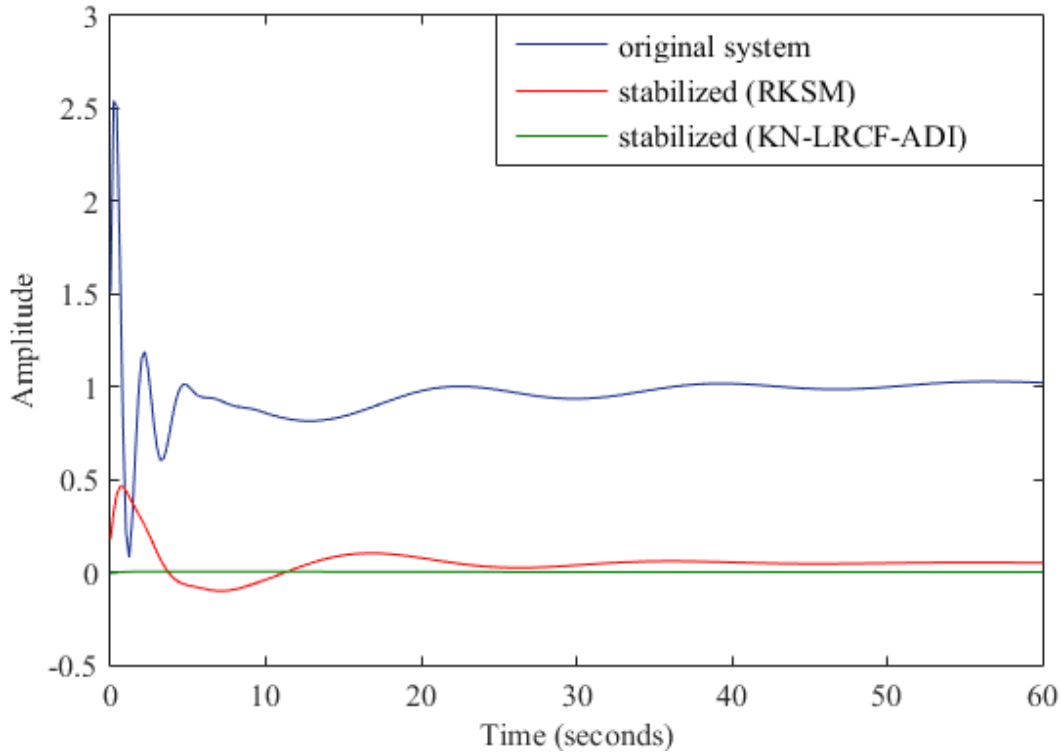


(b) Second-input/First-output

Figure. 5.8. Comparisons of step-responses for the model $mod - 606$ (1st part)

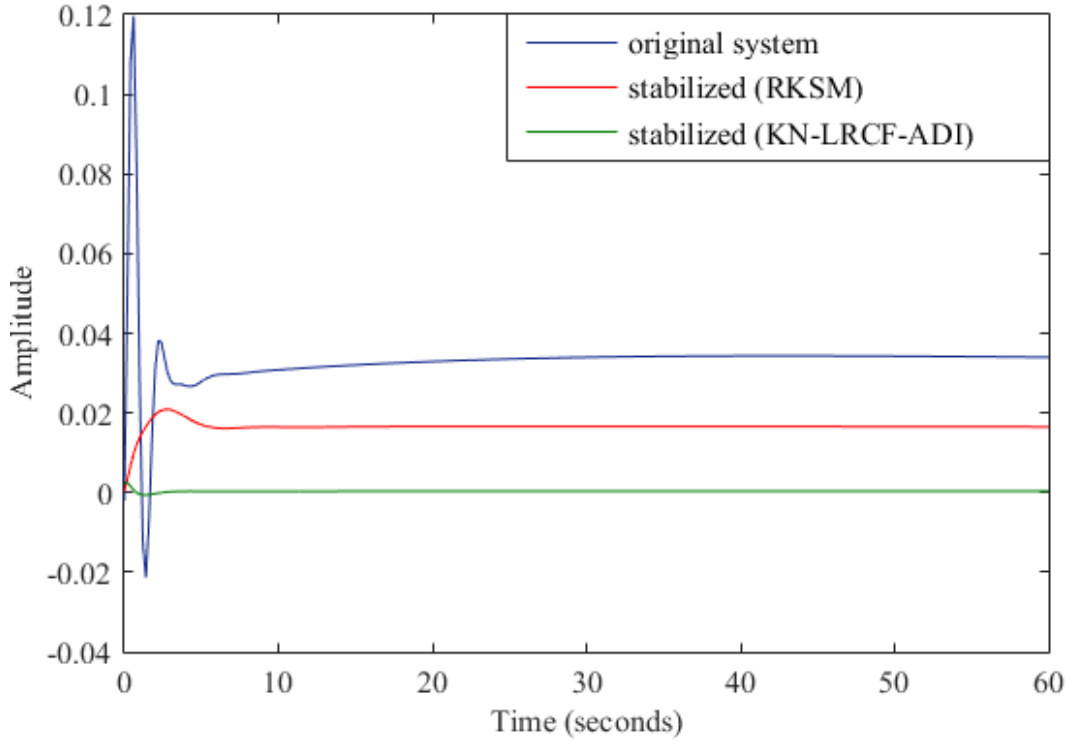


(a) Third-input/First-output

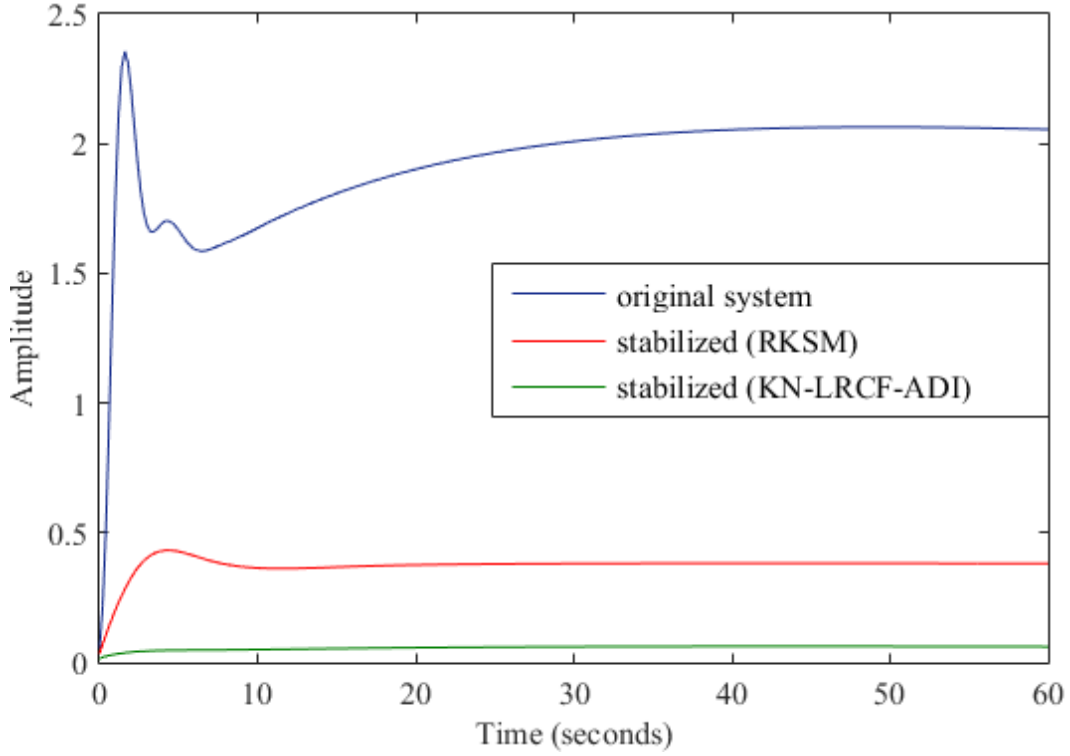


(b) Fourth-input/Third-output

Figure. 5.9. Comparisons of step-responses for the model $mod-606$ (2nd part)

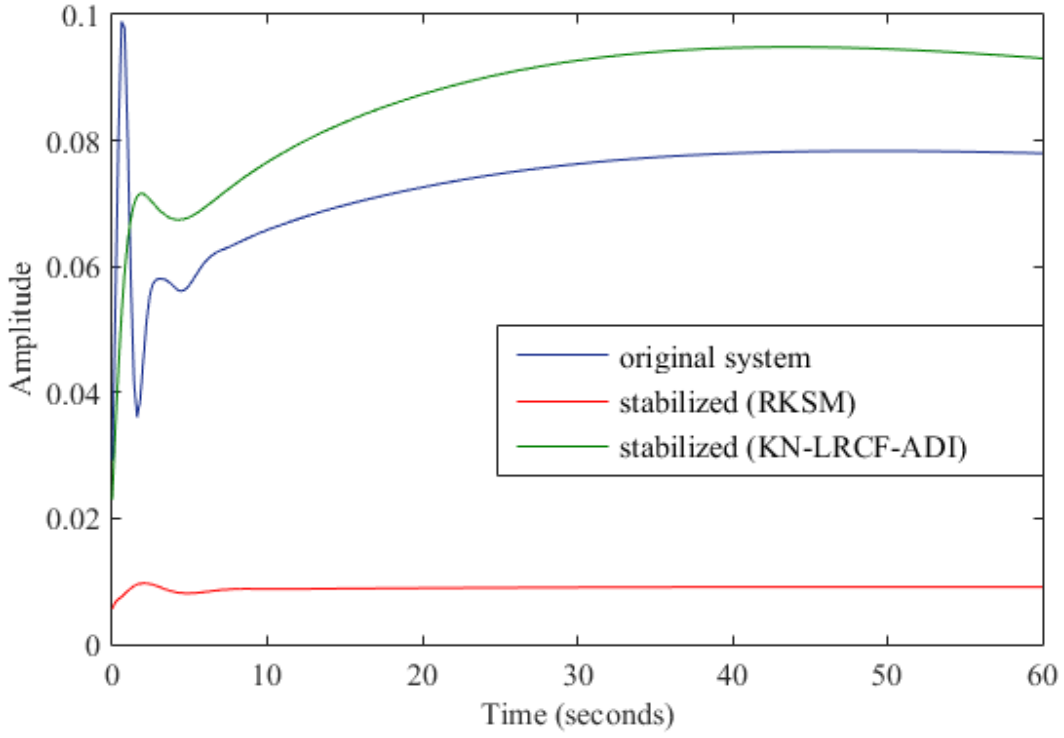


(a) First-input/Third-output

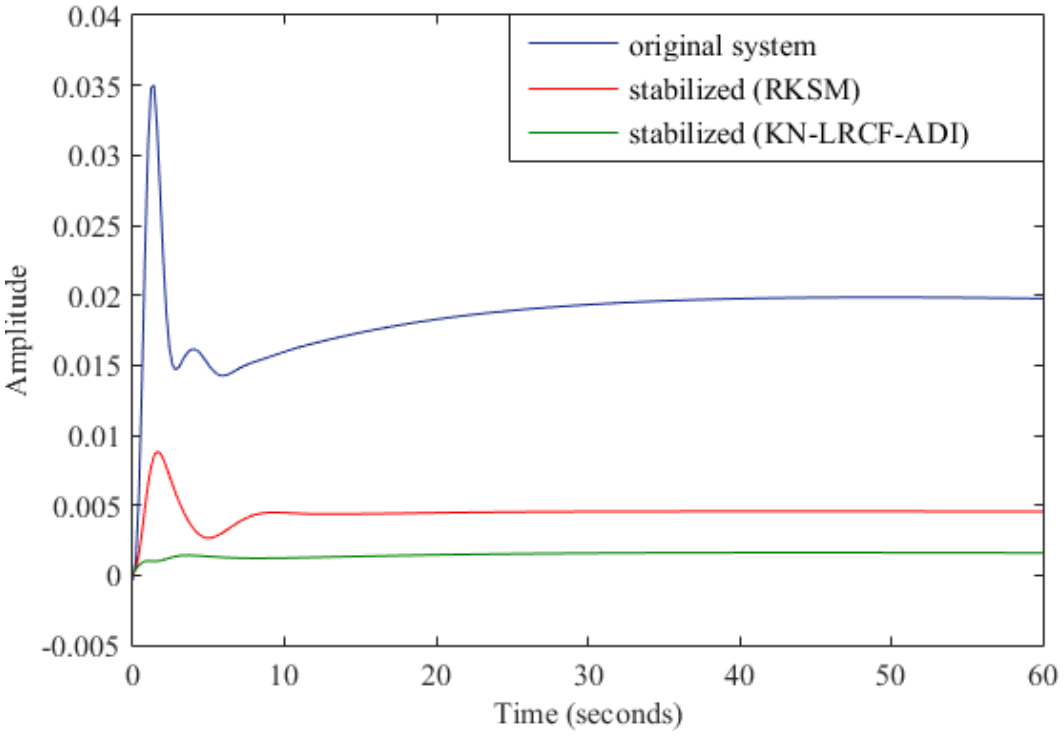


(b) Second-input/First-output

Figure. 5.10. Comparisons of step-responses for the model $mod-1998$ (1st part)

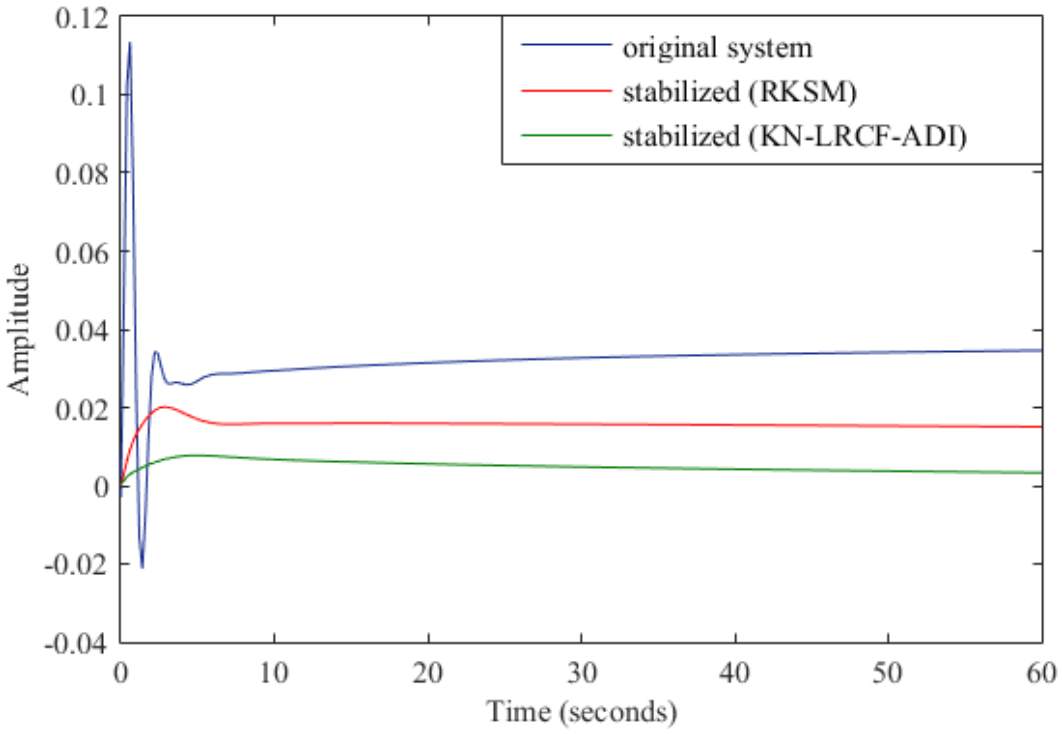


(a) Third-input/First-output

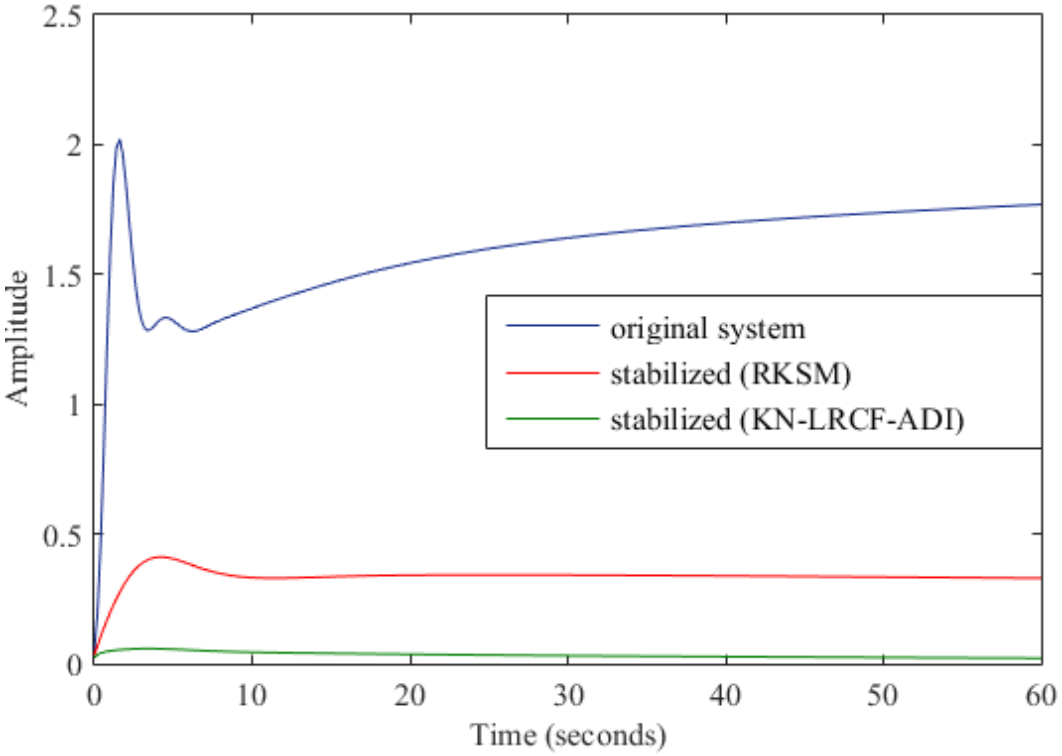


(b) Fourth-input/Third-output

Figure. 5.11. Comparisons of step-responses for the model $mod - 1998$ (2nd part)

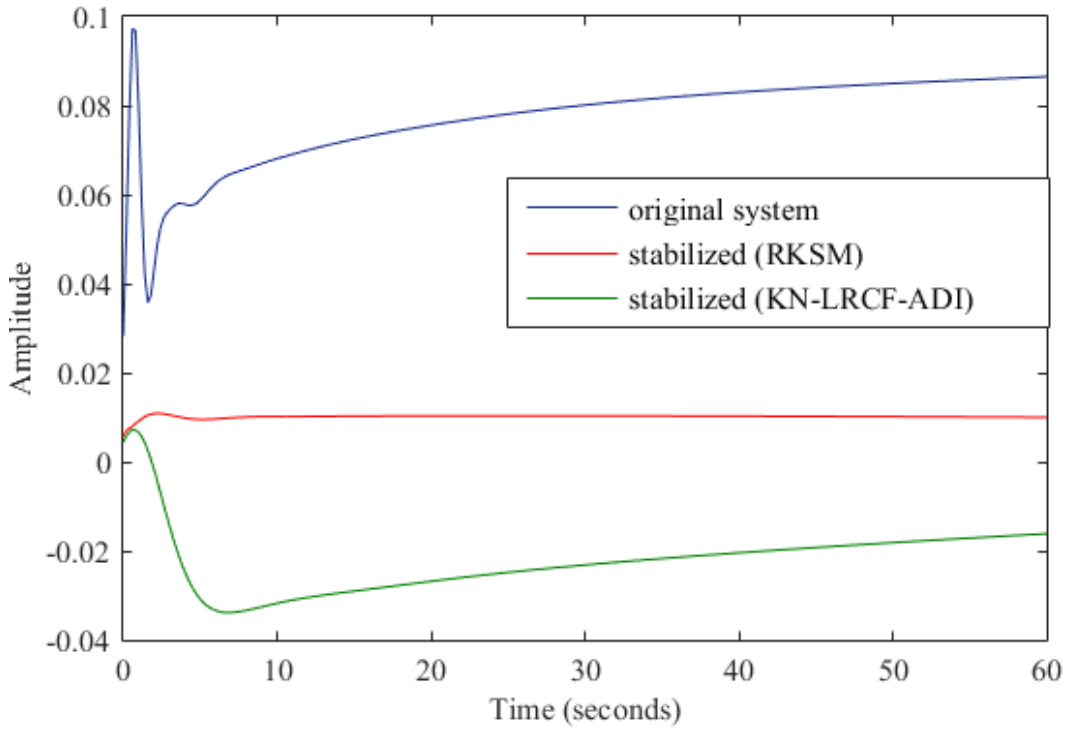


(a) First-input/Third-output

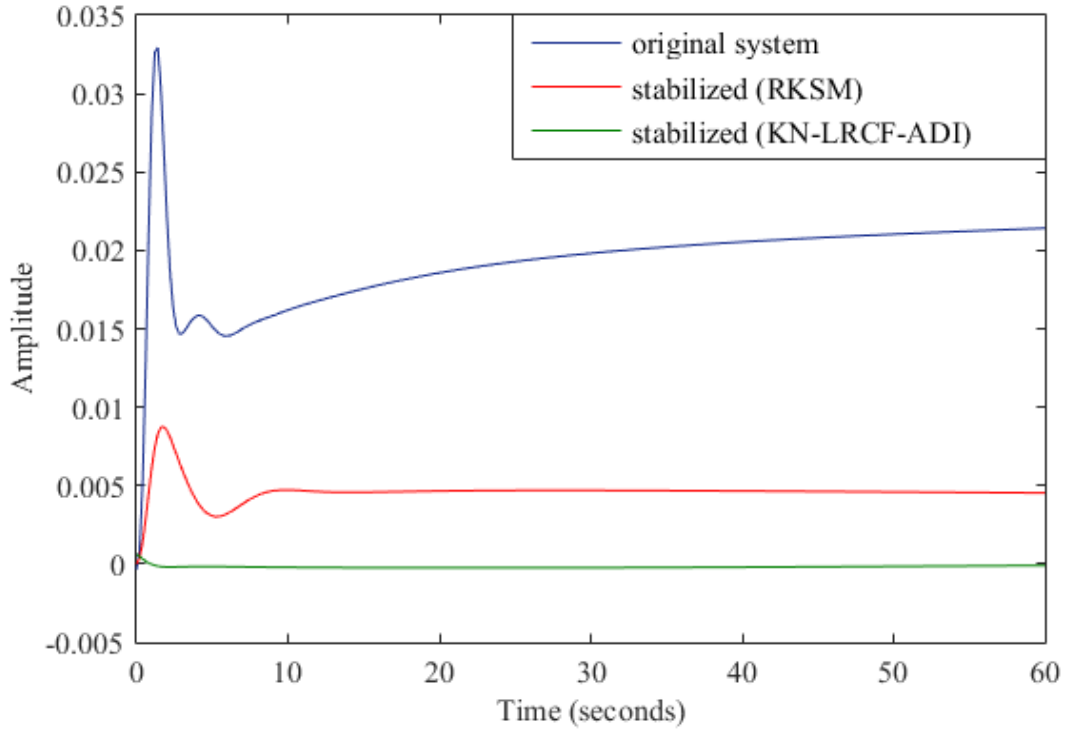


(b) Second-input/First-output

Figure. 5.12. Comparisons of step-responses for the model $mod-2476$ (1st part)

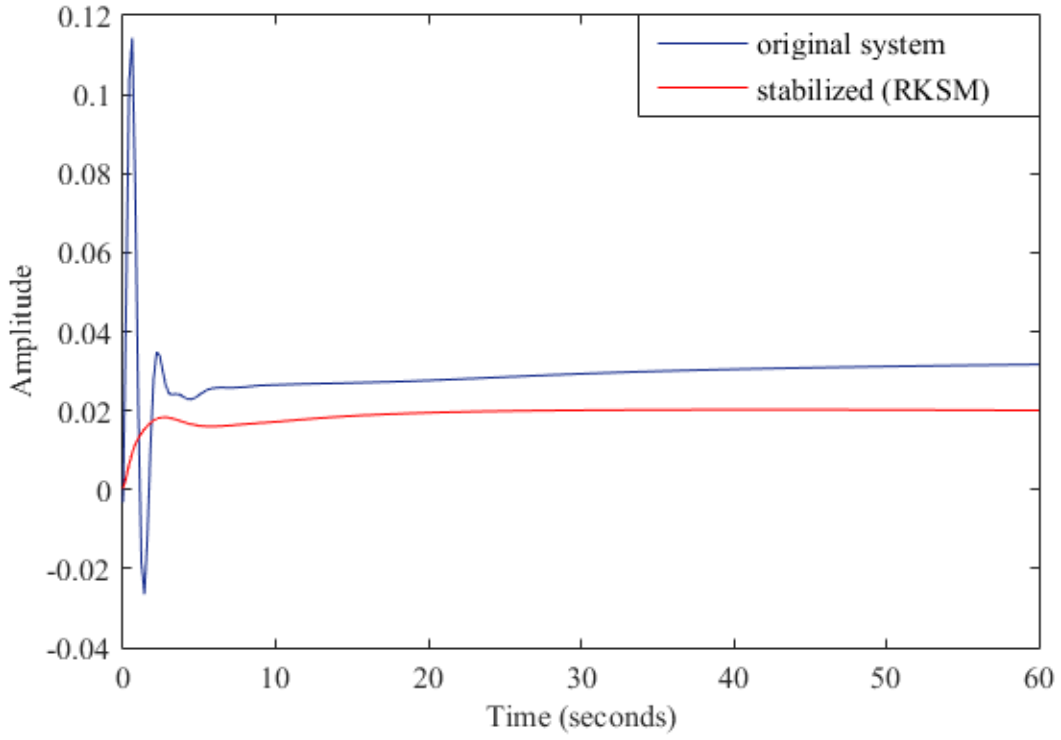


(a) Third-input/First-output

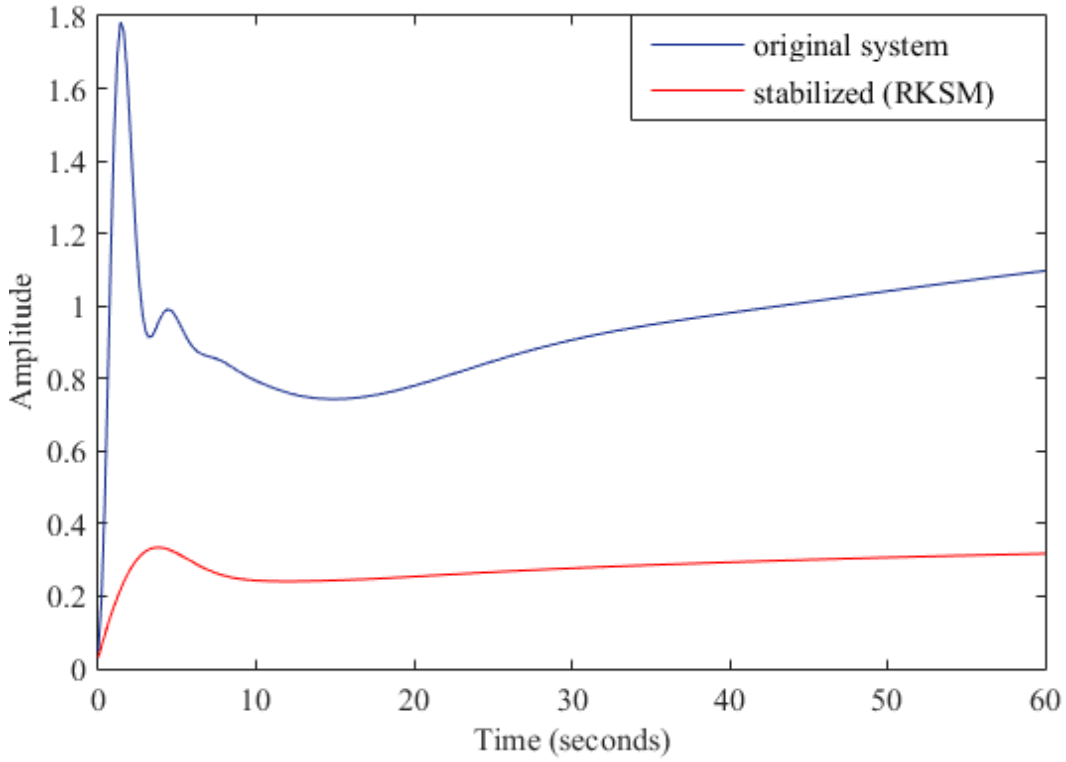


(b) Fourth-input/Third-output

Figure. 5.13. Comparisons of step-responses for the model $mod - 2476$ (2nd part)

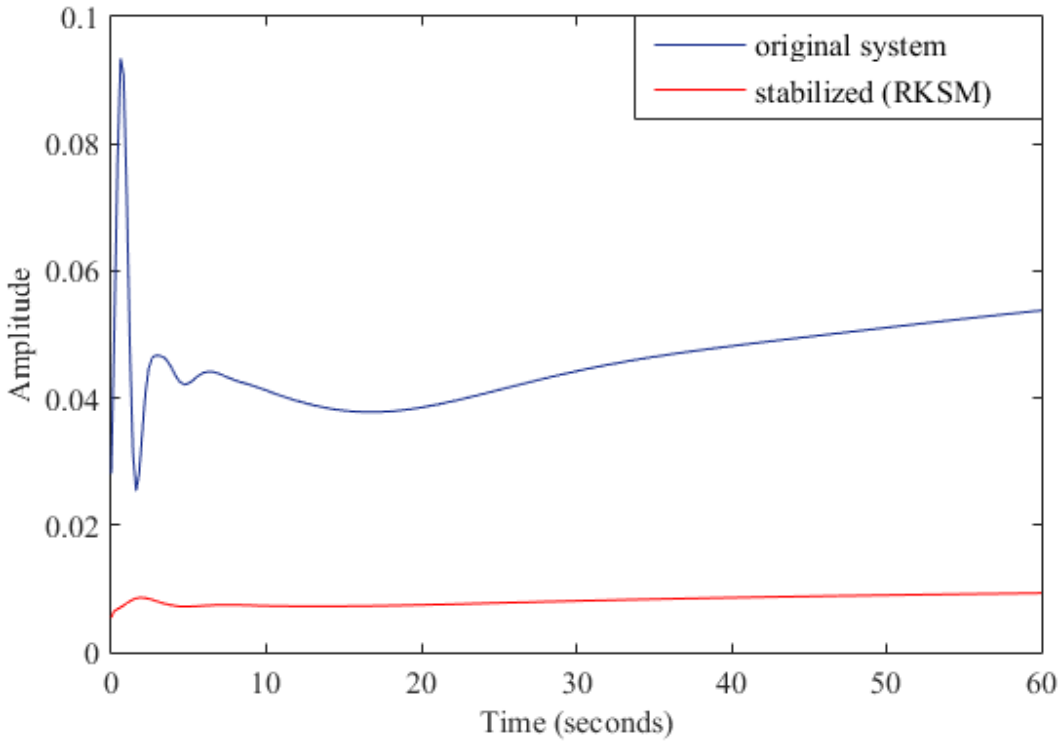


(a) First-input/Third-output

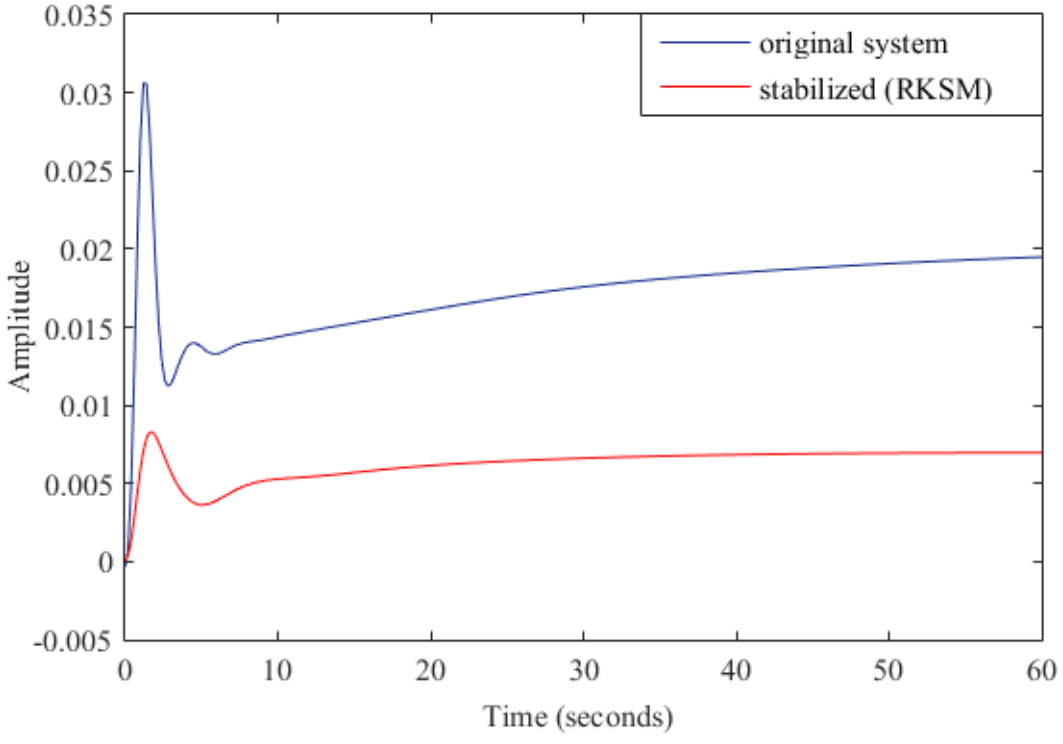


(b) Second-input/First-output

Figure. 5.14. Comparisons of step-responses for the model $mod-3078$ (1st part)



(a) Third-input/First-output



(b) Fourth-input/Third-output

Figure. 5.15. Comparisons of step-responses for the model $mod - 3078$ (2nd part)

Chapter 6

Conclusion and Future Research

In this chapter, we discuss the findings and limitations of the research work and draw the concluding remarks. Also, we highlight some future research work in brief.

6.1 Conclusion

The thesis is mainly divided into two parts. Rational Krylov Subspace Method (RKSM) is discussed for the solution of Continuous Algebraic Riccati Equation (CARE) and LRCF integrated Alternative Direction Implicit (LRCF-ADI) technique based Kleinman-Newton method is discussed as the comparative method.

A Krylov subspace-based projection method for solving Continuous Algebraic Riccati Equations (CARE) derived from the large-scale sparse index-1 descriptor system and hence apply Riccati based feedback stabilization to the unstable systems. Implementation of the adaptive shift parameters enhanced the feasible construction of the Krylov subspaces in the sense of computation time and rapid convergence. Application of the *Sherman-Morrison-Woodbury* formula made the iterative steps smother. A normalized residual technique for the stopping condition of the proposed method is derived as well. Sparsity preserving LRCF-ADI techniques is introduced to solve the Continuous Algebraic Lyapunov Equation (CALE). The conversion of CARE into CALE is discussed for the index-1 descriptor systems and hence Kleinman-Newton method is applied to solve the converted CALE, where the inner iterations are executed via the structured LRCF-ADI technique. Also, a factorized residual technique is introduced as the convergence criteria.

Applicability, efficiency, and robustness of the proposed methods are validated by applying them to the unstable index-1 descriptor systems derived from Brazilian Interconnected Power Systems (BIPS). Stabilization of the unstable index-1 descriptor systems is investigated through the eigenvalue comparisons and stabilization of the step-responses of several dominant input-output relations.

From the analytical and graphical comparisons of the results of numerical computations, the findings are as follows:

- By both RKSM and KN-LRCF-ADI techniques CAREs arising from the unstable index-1 descriptor systems are efficiently solved and the corresponding models are stabilized.

- The semi-stable index-1 descriptor system successfully stabilized through Riccati based feedback stabilization by RKSM, whereas KN-LRCF-ADI is still not suitable for it.

- There are deviations of the numerical ranks of the factored solutions of CAREs in the Kleinman-Newton approaches and RKSM provides significantly better results for all the cases.

- RKSM approach has quick convergence ability and occupies very small solution spaces to provide the efficient solutions of the CAREs. In contrast LRCF-ADI based Kleinman-Newton has several approaches for finding the solutions of CAREs, where almost all of the approaches required higher computation time.

- Riccati based feedback stabilization for the index-1 descriptor systems by the RKSM approach is very effective and robust. Contrariwise, LRCF-ADI based Kleinman-Newton method is slightly scattered in case of the stabilization of step-responses.

Thus, it can be concluded that the RKSM is suitably applicable to unstable index-1 descriptor systems for Riccati based feedback stabilization and this method is more preferable than the Kleinman-Newton method in the sense of computation time and memory allocation.

6.2 Future Research

The research of the thesis can be extended for the future. The projection-based iterative techniques through Krylov subspace will be a vital part of the reduced-order model generation and system stabilization. In control theory, applications of the RKSM will be very effective for the cheap computation time and minimized storage requirement. The techniques proposed in this thesis can be applied for the descriptor systems of the higher indices. The proposed techniques can be applied to the converted first-order form of the second-order systems as well.

In this thesis the MATLAB library command `care` is used in RKSM to find the solution of the Riccati equation governed from the reduced-order model. In the future, we will try to find the self-sufficient RKSM algorithm for solving Riccati equations. Also, the LRCF-ADI techniques are not applicable to the semi-stable systems, we have a goal to work on it as well.

In future research, we will try to apply the Iterative Rational Krylov Algorithm (IRKA) to find the solution of the Riccati equation governed from large-scale sparse descriptor systems of different indices and higher orders. Moreover, the development of a machine-independent iterative solver will be tried for the matrix equations (i.e., Riccati, Lyapunov) arising from large-scale sparse descriptor systems.

References

[1] Zhou, Y., *Numerical methods for large scale matrix equations with applications in LTI system model reduction*," Ph.D. dissertation, Rice University, Houston, Texas, May 2002.

[2] Craig, R. and Kurdila, A. J., *Fundamentals of Structural Dynamics*. Wiley, 2006.

[3] Gugercin, S., Antoulas, A. C. and Beattie, C., "H_2 model reduction for large-scale linear dynamical systems," *SIAM journal on matrix analysis and applications*, vol. 30, no. 2, pp. 609–638, 2008.

[4] Benner, P. and Saak, J., "Efficient balancing-based MOR for large-scale second-order systems," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 17, no. 2, pp. 123–143, 2011.

[5] Knizhnerman, L. and Simoncini, V., "A new investigation of the extended Krylov subspace method for matrix function evaluations," *Numerical Linear Algebra with Applications*, vol. 17, no. 4, pp. 615–638, 2010.

[6] Antoulas, A. C., Beattie, C. A. and Gugercin, S., *Interpolatory model reduction of large-scale dynamical systems*. Springer, 2010.

[7] Stykel, T. and Simoncini, V., "Krylov subspace methods for projected Lyapunov equations," *Applied Numerical Mathematics*, vol. 62, no. 1, pp. 35–50, Jan. 2012.

[8] Sinani, K., *Iterative rational Krylov algorithm for unstable dynamical systems and generalized coprime factorizations*, Ph.D. dissertation, Virginia Tech, 2016.

[9] Flagg, G. M., Beattie, C. A. and Gugercin, S., "Convergence of the iterative rational Krylov algorithm," *Systems & Control Letters*, vol. 61, pp. 688–691, 2012.

[10] Freitas, F. D., Rommes, J. and Martins, N., "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.

[11] Freitas, F. D., Martins, N., Varricchio, S. L., Rommes, J. and Véliz, F. C., "Reduced-order transfer matrices from RLC network descriptor models of electric power grids," *IEEE transactions on power systems*, vol. 26, no. 4, pp. 1905–1916, 2011.

[12] Li, J. R., *Model reduction of large linear systems via low rank system gramians*, Ph.D. dissertation, Massachusetts Institute of Technology, 2000.

[13] Jbilou, K., "Block Krylov subspace methods for large algebraic Riccati equations," *Numerical Algorithms*, vol. 34, no. 2-4, pp. 339–353, 2003.

[14] Heyouni, M. and Jbilou, K., "An extended block Arnoldi algorithm for large-scale solutions of the continuous-time algebraic Riccati equation," *Electronic Transactions on Numerical Analysis*, vol. 33, pp. 53–62, 2009.

[15] Simoncini, V., "A new iterative method for solving large-scale Lyapunov matrix equations," *SIAM Journal on Scientific Computing*, vol. 29, no. 3, pp. 1268–1288, 2007.

[16] Druskin, V. and Simoncini, V., "Adaptive rational Krylov subspaces for large-scale dynamical systems," *Systems & Control Letters*, vol. 60, no. 8, pp. 546–560, 2011.

[17] Simoncini, V., Szyld, D. B. and Monsalve, M., "On two numerical methods for the solution of large-scale algebraic Riccati equations," *IMA Journal of Numerical Analysis*, vol. 34, no. 3, pp. 904–920, 2013.

[18] Simoncini, V., "Analysis of the rational Krylov subspace projection method for large-scale algebraic Riccati equations," *SIAM Journal on Matrix Analysis and Applications*, vol. 37, no. 4, pp. 1655–1674, 2016.

[19] Benner, P., Li,J. R. and Penzl, T., "Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems," *Numerical Linear Algebra with Applications*, vol. 15, pp. 1–23, 2008.

[20] Hylla, T., *Extension of inexact kleinman-newton methods to a general monotonicity preserving convergence theory*, Ph.D. dissertation, Universität Trier, Trier, Germany, 2011.

[21] Gugercin, S., Stykel, T. and Wyatt, S., "Model reduction of descriptor systems by interpolatory projection methods," *SIAM Journal on Scientific Computing*, vol. 35, no. 5, pp. B1010–B1033, 2013.

[22] Huang, C. C., Tsai, J. S. H., Guo, S. M., Sun, Y. J. and Shieh, L. S., "Solving algebraic Riccati equation for singular system based on matrix sign function," *International journal of innovative computing, information & control*, vol. 9, pp. 2771–2788, 2013.

[23] Uddin, M. M., *Computational methods for model reduction of large-scale sparse structured descriptor systems*, Ph.D. dissertation, Otto-von-Guericke Universität Magdeburg, 2015.

[24] Uddin, M. M., Hossain, M. S. and Uddin, M. F., "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.

[25] Hasan, S. and Uddin, M. M., "Model reduction of structured dynamical systems by projecting onto the dominant eigenspace of the gramians," *Journal of Modeling and Optimization*, vol. 10, no. 2, pp. 94–94, 2018.

[26] Hasan, S., Fony, A. M., and Uddin, M. M., "Reduced model based feedback stabilization of large-scale sparse power system model," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. IEEE, 2019, pp. 1–6.

[27] Benner, P., Kürschner, P. and Saak, J., "Self-generating and efficient shift parameters in ADI methods for large Lyapunov and Sylvester equations," *Electronic Transactions on Numerical Analysis (ETNA)*, vol. 43, pp. 142–162, 2014.

[28] Bansch, E., Benner, P., Saak, J. and Weichelt, H. K., "Riccati-based boundary feedback stabilization of incompressible Navier-Stokes flow," *SIAM Journal on Scientific Computing*, vol. 37, no. 2, pp. A832–A858, 2015.

[29] Benner, P., Bujanović, Z., Kürschner, P. and Saak, J., "A numerical comparison of solvers for large-scale, continuous-time algebraic Riccati equations," *arXiv preprint arXiv:1811.00850*, 2018.

[30] Palitta, D., "The projected Newton-Kleinman method for the algebraic riccati equation," *arXiv preprint arXiv:1901.10199*, 2019.

[31] Antsaklis, P. J. and Michel, A. N. , *Linear systems*. Springer Science & Business Media, 2006.

[32] Ogata, K., *Modern control engineering*. Prentice Hall Upper Saddle River, NJ, 2009.

[33] Saad, Y., *Iterative Methods for Sparse Linear Systems*. Philadelphia, PA, USA: SIAM, 2003.

[34] Kose, I., *Introduction to state-space control theory*, Department of Mechanical Engineering, Bogazici University, 2003.

[35] Benner, P. and Stykel, T., "Model order reduction for differential-algebraic equations: A survey," *Surveys in Differential-Algebraic Equations IV*, p. 107, 2017.

[36] Kunkel, P. and Mehrmann, V., *Differential-algebraic equations: analysis and numerical solution*. European Mathematical Society, 2006, vol. 2.

[37] Schilders, W. H., Vorst, H. A. V, and Rommes, J., *Model order reduction: theory, research aspects and applications*. Springer, 2008, vol. 13.

[38] Uddin, M. M., *Model reduction for piezo-mechanical systems using balanced truncation*, Master's thesis, Stockholm, Sweden, 2011. http://www.qucosa.de/fileadmin/data/qucosa/documents/7822/Master_Thesis_Uddin.pdf

[39] Benner, P., Gugercin, S., and Willcox, K., "A survey of projection-based model reduction methods for parametric dynamical systems," *SIAM review*, vol. 57, no. 4, pp. 483–531, 2015.

[40] Benner, P. and Schneider, A., *Reduced Representation of Power Grid Models*, P. Benner, Ed. Cham: Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-07236-4_3

[41] Gajic, Z. and Qureshi, M. T. J., *Lyapunov matrix equation in system stability and control*. Courier Corporation, 2008.

[42] Chu, E. K. W., "Solving large-scale algebraic Riccati equations by doubling," in *Talk presented at the Seventeenth Conference of the International Linear Algebra Society, Braunschweig, Germany*, vol. 22, 2011.

[43] Barrachina, S., Benner, P., and Quintana-Ortí, E. S., "Efficient algorithms for generalized algebraic Bernoulli equations based on the matrix sign function," *Numerical Algorithms*, vol. 46, no. 4, pp. 351–368, 2007.

[44] Amodei, L. and Buchot, J., "A stabilization algorithm based on algebraic Bernoulli equation," *Numerical Linear Algebra with Applications*, 2009.

[45] Chen, W. and Qiu, L., "Linear quadratic optimal control of continuous-time LTI systems with random input gains," *IEEE Transactions on Automatic Control*, vol. 61, no. 7, pp. 2008–2013, 2016.

[46] Benner, P., Byers, R., Mehrmann, V. and Xu, H., "A robust numerical method for the γ-iteration in H_∞ control," *Linear Algebra and its Applications*, vol. 425, no. 2-3, pp. 548–570, 2007.

[47] Benner, P., Saak, J. and Uddin, M. M., "Structure preserving model order reduction of large sparse second-order index-1 systems and application to a mechatronics model," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 22, no. 6, pp. 509–523, 2016.

[48] Benner, P., Kürschner, P. and Saak, J., "An improved numerical method for balanced truncation for symmetric second-order systems," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 19, no. 6, pp. 593–615, 2013.

[49] Datta, B., *Numerical Methods for Linear Control Systems*. Elsevier, 2004.

[50] Cosentino, C. and Bates, D., *Feedback control in systems biology*. Crc Press, 2011.

[51] Hein, S., *MPC-LQG based optimal control of parabolic PDEs*, Ph.D. Dissertation, TU Chemnitz, February 2009, http://archiv.tu-chemnitz.de/pub/2010/0013.

[52] Benner, P. and Heiland, J., "Convergence of approximations to Riccati-based boundary-feedback stabilization of laminar flows," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12 296–12 300, 2017.

[53] Uddin, M., Khan, M. H. and Uddin, M. M., "Riccati based optimal control for linear quadratic regulator problems," in *2019 5th International Conference on Advances in Electrical Engineering (ICAEE)*. IEEE, 2019, pp. 290–295.

[54] Uddin, M., Khan, M. H. and Uddin, M. M., "Efficient computation of Riccati-based optimal control for power system models," in *2019 22nd International Conference of Computer and Information Technology (ICCIT)*. IEEE, 2019, pp. 260–265.

[55] Fassbender, H. and Kressner, D., "Structured eigenvalue problems," *GAMM-Mitteilungen*, vol. 29, no. 2, pp. 297–318, 2006.

[56] Saad, Y., *Numerical methods for large eigenvalue problems: revised edition*. Siam, 2011, vol. 66.

[57] Roger, H. and Charles, R. J., *Topics in matrix analysis*. Cambridge University Press, 1994.

[58] Van den Bos, A., *Parameter estimation for scientists and engineers*. John Wiley & Sons, 2007.

[59] Duff, I. S., Erisman, A. M. and Reid, J. K., *Direct methods for sparse matrices*. Oxford University Press, 2017.

[60] Trefethen, L. N. and Bau III, D., *Numerical linear algebra*. Siam, 1997, vol. 50.

[61] Benner, P., Mehrmann, V. and Sorensen, D. C., *Dimension reduction of large-scale systems*. Springer, 2005, vol. 45.

[62] Golub, G. H. and Van Loan, C. F., *Matrix Computations*, 3rd ed. Baltimore: Johns Hopkins University Press, 1996.

[63] Schott, J. R., *Matrix analysis for statistics*. John Wiley & Sons, 2016.

[64] Antoulas, A. C., *Approximation of large-scale dynamical systems*. Siam, 2005, vol. 6.

[65] Mehrmann, V. and Stykel, T., *Balanced truncation model reduction for large-scale systems in descriptor form*. Springer, 2005.

[66] Datta, B. N., *Numerical linear algebra and applications*. Siam, 2010, vol. 116.

[67] Kressner, D., "Memory-efficient Krylov subspace techniques for solving large-scale Lyapunov equations," in *2008 IEEE International Conference on Computer-Aided Control Systems*. IEEE, 2008, pp. 613–618.

[68] Demmel, J. W., *Applied numerical linear algebra*. Siam, 1997, vol. 56.

[69] Treves, F., *Topological Vector Spaces, Distributions and Kernels: Pure and Applied Mathematics*. Elsevier, 2016, vol. 25.

[70] Jeffrey, A. and Zwillinger, D., *Table of integrals, series, and products*. Elsevier, 2007.

[71] Bini, D. A., Iannazzo, B. and Meini, B., *Numerical solution of algebraic Riccati equations*. Siam, 2012, vol. 9.

[72] Benner, P. and Bujanović, Z., "On the solution of large-scale algebraic Riccati equations by using low-dimensional invariant subspaces," *Linear Algebra and its Applications*, vol. 488, pp. 430–459, 2016.

[73] Byers, R., "A Hamiltonian-Jacobi algorithm," *IEEE transactions on automatic control*, vol. 35, no. 5, pp. 566–570, 1990.

[74] Bai, Z. and Demmel, J., "Using the matrix sign function to compute invariant subspaces," *SIAM Journal on Matrix Analysis and Applications*, vol. 19, no. 1, pp. 205–225, 1998.

[75] Krylov, A. N., "On the numerical solution of the equation by which in technical questions frequencies of small oscillations of material systems are determined," *Izvestija AN SSSR (News of Academy of Sciences of the USSR), Otdel. mat. i estest. nauk*, vol. 7, no. 4, pp. 491–539, 1931.

[76] Lein, C., Beitelschmidt, M. and Bernstein, D., "Improvement of Krylov subspace reduced models by iterative mode-truncation," *IFAC-PapersOnLine*, vol. 48, no. 1, pp. 178–183, 2015.

[77] Gams, M. and Ozek, M., "Model order reduction for large LTI control systems," *Journal of Computer and Systems Sciences International*, vol. 49, no. 5, pp. 696–701, 2010.

[78] Morris, K. and Navasca, C., "Approximation of low rank solutions for linear quadratic control of partial differential equations," *Computational Optimization and Applications*, vol. 46, no. 1, pp. 93–111, 2010.

[79] Guo, C. H. and Laub, A. J., "On a Newton like method for solving algebraic Riccati equations," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 2, pp. 694–698, 2000.

[80] Sorensen, D. C. and Zhou, Y., "Direct methods for matrix Sylvester and Lyapunov equations," *Journal of Applied Mathematics*, vol. 2003, pp. 277–303, 2002.

[81] Wachspress, E. L., "The ADI minimax problem for complex spectra," *Applied Mathematics Letters*, vol. 1, no. 3, pp. 311–314, 1988.

[82] Penzl, T., "A cyclic low-rank Smith method for large sparse Lyapunov equations," *SIAM Journal on Scientific Computing*, vol. 21, no. 4, pp. 1401–1418, 1999.

[83] Wachspress, E. L., *The ADI model problem*. Springer, 2013.

[84] Saak, J., *Efficient numerical solution of large scale algebraic matrix equations in pde control and model order reduction*, Ph.D. dissertation, TU Chemnitz, July 2009, available from http://nbn-resolving.de/urn:nbn:de:bsz:ch1-200901642.

[85] Mena, H. and Saak, J., "On the parameter selection problem in the Newton-ADI iteration for large-scale Riccati equations," *Electronic Transactions on Numerical Analysis*, vol. 29, pp. 136–149, 2008.

[86] Benner, P., Kürschner, P. and Saak, J., "Efficient handling of complex shift parameters in the low-rank Cholesky factor ADI method," *Numerical Algorithms*, vol. 62, no. 2, pp. 225–251, 2013.

[87] Lin, Y. and Simoncini, V., "A new subspace iteration method for the algebraic Riccati equation," *Numerical Linear Algebra with Applications*, vol. 22, no. 1, pp. 26–47, 2015.

[88] Abidi, O., Hached, M. and Jbilou, K., "Adaptive rational block Arnoldi methods for model reductions in large-scale MIMO dynamical systems," *New Trends in Mathematical Sciences*, vol. 4, no. 2, pp. 227–239, 2016.

[89] Lin, Y. and Simoncini, V., "Minimal residual methods for large scale Lyapunov equations," *Applied Numerical Mathematics*, vol. 72, pp. 52–71, 2013.

[90] Ruhe, A., "Rational Krylov algorithms for non-symmetric eigenvalue problems. ii. matrix pairs," *Linear algebra and its Applications*, vol. 197, pp. 283–295, 1994.

[91] Uddin, M. M., *Computational Methods for Approximation of Large-Scale Dynamical Systems*. Chapman and Hall/CRC, 2019.

[92] Peaceman, D. W. and Rachford Jr, H. H., "The numerical solution of parabolic and elliptic differential equations," *Journal of the Society for industrial and Applied Mathematics*, vol. 3, no. 1, pp. 28–41, 1955.

[93] Beckermann, B. and Gryson, A., "Extremal rational functions on symmetric discrete sets and super-linear convergence of the ADI method," *Constructive Approximation*, vol. 32, no. 3, pp. 393–428, 2010.

[94] F. Feitzinger, Hylla, T. and Sachs, E. W., "Inexact Kleinman-Newton method for Riccati equations," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 2, pp. 272–288, 2009.

[95] Kürschner, P., *Efficient low-rank solution of large-scale matrix equations*, Ph.D. dissertation, Shaker Verlag Aachen, 2016.

[96] Martins, N., Pellanda,, P. C. and Rommes, J., "Computation of transfer function dominant zeros with applications to oscillation damping control of large power systems," *IEEE transactions on power systems*, vol. 22, no. 4, pp. 1657–1664, 2007.

[97] Freitas, F. and Costa, A. S., "Computationally efficient optimal control methods applied to power systems," *IEEE transactions on power systems*, vol. 14, no. 3, pp. 1036–1045, 1999.

[98] Rommes, J. and Martins, N., "Exploiting structure in large-scale electrical circuit and power system problems," *Linear Algebra and its Applications*, vol. 431, no. 3-4, pp. 318–333, 2009.

[99] Decker, I. C., e Silva, A. S., Agostini, M. N., Prioste, F. B., Mayer, B. T. and Dotta, D., "Experience and applications of phasor measurements to the Brazilian interconnected power system," *European Transactions on Electrical Power*, vol. 21, no. 4, pp. 1557–1573, 2011.

[100] Leandro, R., e Silva, A. S., Decker, I. and Agostini, M., "Identification of the oscillation modes of a large power system using ambient data," *Journal of Control, Automation and Electrical Systems*, vol. 26, no. 4, pp. 441–453, 2015.