



HAL
open science

Rational Lanczos-type methods for model order reduction

Houda Barkouki

► **To cite this version:**

Houda Barkouki. Rational Lanczos-type methods for model order reduction. General Mathematics [math.GM]. Université du Littoral Côte d'Opale; Université Cadi Ayyad (Marrakech, Maroc). Faculté des sciences et techniques Guéliz, 2016. English. NNT : 2016DUNK0440 . tel-01556601

HAL Id: tel-01556601

<https://theses.hal.science/tel-01556601>

Submitted on 5 Jul 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITY CADI AYYAD
FACULTY OF SCIENCES
AND TECHNOLOGIES
MARRAKECH

UNIVERSITY OF LITTORAL
CÔTE D'OPALE, LMPA
- CALAIS -
FRANCE.

A dissertation submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy in Mathematics

Prepared as part of a joint supervision between
University Cadi Ayyad and University of Littoral Côte d'Opale

Rational Lanczos-type methods for model order reduction.

Presented on 22 December 2016 by
Houda BARKOUKI

The members of jury:

Hassane SADOK	University of Littoral Côte d'Opale, France	President
Abdeslem Hafid BENTBIB	University Cadi Ayyad, FST Marrakech	Supervisor
Khalide JBILOU	University of Littoral Côte d'Opale, France	Supervisor
Jilali ABOUIR	University Hassan II, FST Mohammedia	Reporter
Mohamed EL ALAOUI TALIBI	University Cadi Ayyad, FS Semlalia, Marrakech	Reporter
Mohammed GUEDDA	University Picardie Jules-Verne, France	Reporter

Cette thèse a été préparée aux:



Laboratoire de Mathématiques Appliquées et Informatique

Université Cadi Ayyad

Faculté des Sciences et Techniques.

B.P. 549, Av. Abdelkarim Elkhattabi

Guéliz Marrakech - Maroc

Télé: 05 24 43 34 04 / 05 24 43 31 63

Fax: 05 24 43 31 70

Site: www.fstg-marrakech.ac.ma



Laboratoire de Mathématiques Pures et Appliquées

Centre Universitaire de la Mi-Voix

Maison de la Recherche Blaise Pascal

50, Rue Ferdinand Buisson

CS 80699

Calais Cedex, France

Télé: 03 21 46 55 90

fax: 03 21 46 55 86

Site: www-lmpa.univ-littoral.fr

Abstract

Numerical solution of dynamical systems have been a successful means for studying complex physical phenomena. However, in large-scale setting, the system dimension makes the computations infeasible due to memory and time limitations, and ill-conditioning. The remedy of this problem is model reduction. This dissertation focuses on projection methods to efficiently construct reduced order models for large linear dynamical systems. Especially, we are interested by projection onto unions of standard Krylov subspaces which lead to a class of reduced order models known as rational interpolation. Based on this theoretical framework that relate Krylov projection to rational interpolation, four rational Lanczos-type algorithms for model reduction are proposed.

At first, an adaptive rational block Lanczos-type method for reducing the order of large scale dynamical systems is introduced, based on a rational block Lanczos algorithm and an adaptive approach for choosing the interpolation points. A generalization of the first algorithm is also given where different multiplicities are considered for each interpolation point. Next, we proposed another extension of the standard Krylov subspace method for Multiple-Input Multiple-Output (MIMO) systems, which is the global Krylov subspace, and we obtained also some equations that describe this process. Finally, an extended block Lanczos method is introduced and new algebraic properties for this algorithm are also given.

The accuracy and the efficiency of all proposed algorithms when applied to model order reduction problem are tested by means of different numerical exper-

iments that use a collection of well known benchmark examples.

Keywords: Lanczos algorithm, Model reduction, Moment matching, Rational Krylov subspace, Transfer function.

Résumé

La solution numérique des systèmes dynamiques est un moyen efficace pour étudier des phénomènes physiques complexes. Cependant, dans un cadre à grande échelle, la dimension du système rend les calculs infaisable en raison des limites de mémoire et de temps, ainsi que le mauvais conditionnement. La solution de ce problème est la réduction de modèles. Cette thèse porte sur les méthodes de projection pour construire efficacement des modèles d'ordre inférieur à partir des systèmes linéaires dynamiques de grande taille. En particulier, nous nous intéressons à la projection sur la réunion de plusieurs sous-espaces de Krylov standard qui conduit à une classe de modèles d'ordre réduit. Cette méthode est connue par l'interpolation rationnelle. En se basant sur ce cadre théorique qui relie la projection de Krylov à l'interpolation rationnelle, quatre algorithmes de type Lanczos rationnel pour la réduction de modèles sont proposés.

Dans un premier temps, nous avons introduit une méthode adaptative de type Lanczos rationnel par block pour réduire l'ordre des systèmes linéaires dynamiques de grande taille, cette méthode est basée sur l'algorithme de Lanczos rationnel par block et une méthode adaptative pour choisir les points d'interpolation. Une généralisation de ce premier algorithme est également donnée, où différentes multiplicités sont considérées pour chaque point d'interpolation. Ensuite, nous avons proposé une autre extension de la méthode du sous-espace de Krylov standard pour les systèmes à plusieurs-entrées plusieurs-sorties, qui est le sous-espace de Krylov global. Nous avons obtenu des équations qui décrivent cette procédure. Finalement, nous avons proposé une méthode de Lanczos étendu par block et nous

avons établi de nouvelles propriétés algébriques pour cet algorithme.

L'efficacité et la précision de tous les algorithmes proposés, appliqués sur des problèmes de réduction de modèles, sont testées dans plusieurs exemples numériques.

Mots clés: Algorithme de Lanczos, Fonction de transfert, Moment correspondant, Réduction de modèle, Sous-espace de Krylov rationnel.

To my parents

*That no dedication can express what I owe them for their support, affection, and
everything*

To my husband

He shared with me difficult moments

To my Family and Friends

For the moral support and encouragement

Finally to all my teachers and specially my dear supervisors Professor

Abdeslem Hafid Bentbib and Professor Khalide Jbilou

This thesis would be impossible without you

Acknowledgements

I measure now how much the quality of supervision and environmental wealth I received was exceptional. This is why it is important for me to extend my sincere appreciation to all those who helped and supported me throughout this thesis, that this support's nature professional or friendly, whether regular or occasional. I have many people to thank who contributed significantly to my doctoral studies and made this dissertation possible.

Foremost on this list are my dear supervisors Professor Abdeslem Hafid Bentbib and Professor Khalide Jbilou. Beyond doubt this thesis would be impossible without them. Many thanks to them for all the guidance, encouragement and support. Their knowledge and their excellence in doing research have influenced me at each and every step of my graduate study and will continue to do so in the future as well. With their careful and detailed readings and corrections of every material I have written, I can't also forget that they taught me how to write my first academic paper and how to put the ideas into words. Honestly, this small paragraph can't expressed how I am very grateful to my dear supervisors, and I will never forget what they did for me.

Besides my supervisors, I would like to thank Professor Nouredine Alaa for his insightful comments and encouragement, and also for his sympathy and kindness to me.

My sincere thanks also goes to Professor Jilali Abouir, Professor Mohamed El Alaoui Talibi and Professor Mohammed Guedda, who accepted the hard work as a reporters of this thesis and have to devote time and energy to a depth reading that gave rise to comments whose richness and relevance have effectively guided the finalization of the manuscript. I also wish to thank Professor Hassane Sadok for agreeing to examine this thesis. He devoted time and brought their expert judgement to my work.

I also want to express my gratitude to my Professors of the Mathematics Department at the Faculty of Sciences and Technology Marrakech, without whom my choice to move towards mathematics research would certainly not have been more apparent.

I'm taking this opportunity to thank all members of LAMAI laboratory and all those with whom I shared pleasant moments in different laboratories or out. I also thank all those who by their competence, their friendship, their support, encouragement or just their presence contributed throughout these few years have to make it enjoyable life in the faculty. I thank also my fellow lab-mates for the stimulating discussions, and for all the fun we have had in the last three years.

And because the years of thesis are also years of life, I can't fail to address a thank to all of who shared my joys and difficult times and who have managed to be very understanding, which have remained in the shadows but be always there to offer me friendship,

I thank my family for their kindness, thoughtfulness, dedication and love with which they have always given me support and comfort. Without them I could never go through with my plans. I can not find words to describe recognition.

The list would be incomplete if I left out all my friends I did not mentioned above, but they know how much they matter to me.

A heartfelt thanks to my husband for hem support and help specially in the last period of my thesis. Her company also and love are one of the main reasons that make me strong and enthusiasm in this period. Thanks for being such a wonderful person, a perfect friend and for always being here whenever I need.

Last but definitely not the least, I would like to thank my parents for supporting me in all my choices with absolute confidence and for making me the young woman I am today. None of this would have been possible if they were not with me. I thank especially my father who helped me financially and morally, how always guide me since my primary school and who always encourage me to continue my studies through. Thank you so much dear parents.

Houda Barkouki

Glossary

Mathematical symbols

\mathbb{R}	set of real numbers
\mathbb{C}	set of complex numbers
$\mathbb{R}^{m \times n}$	set of real $n \times m$ matrices
$\Lambda(A)$	spectrum of matrix A
$\sigma_{max}(A)$	largest singular value of matrix A
0	zero matrix
I_n	identity matrix of dimension $n \times n$
$\text{Re}(s)$	real part of s
\mathbb{C}^-	left half plane, $\{s \in \mathbb{C}^- : \text{Re}(s) < 0\}$
\mathbb{C}^+	right half plane, $\{s \in \mathbb{C}^+ : \text{Re}(s) > 0\}$
A^{-1}	inverse of matrix A
A^T	transpose of matrix A
e_i	i -th unit vector
\langle, \rangle	inner product
\langle, \rangle_F	Frobenius product
$X \perp_F Y$	$\langle X, Y \rangle_F = 0$
$\delta_{i,j}$	Kronecker symbol
j	$\sqrt{-1}$
$\ \cdot\ $	arbitrary norm of a matrix
$\ \cdot\ _2$	spectral norm of a matrix
$\ \cdot\ _F$	Frobenius norm of a matrix
$\ \cdot\ _{\mathcal{H}_2}$	\mathcal{H}_2 norm of a dynamical system
$\ \cdot\ _{\mathcal{H}_\infty}$	\mathcal{H}_∞ norm of a dynamical system

General Notation

A, B, C	state space matrices of the original MIMO state space system
$u(t)$	the input vector of the state space system
$y(t)$	the output vector of the state space system
$x(t)$	the state variable
n	order of the original state space model
Σ	n^{th} order original dynamical system
$F(s)$	transfer function of the original system
A_r, B_r, C_r	state space matrices of the reduced MIMO state space system
r	order of the reduced state space model
Σ_r	r^{th} order reduced system
$F_r(s)$	transfer function of the reduced system
Σ_m	the set of interpolation points
$\sigma_i(\Sigma)$	i^{th} Hankel singular value of Σ
$\epsilon(s)$	the error system
$f_\sigma^{(j)}$	j^{th} moment of Σ at σ
$\hat{f}_\sigma^{(j)}$	j^{th} moment of Σ_r at σ
$f_\infty^{(j)}$	j^{th} Markov parameter of Σ
$\hat{f}_\infty^{(j)}$	j^{th} Markov parameter of Σ_r
$\mathbb{K}_m(\cdot, \cdot)$	the block Krylov subspace
$\mathcal{K}_m(\cdot, \cdot)$	the matrix Krylov subspace
$\mathbb{K}_m^e(\cdot, \cdot)$	the extended block Krylov subspace
$\mathbb{K}_m(\cdot, \cdot, \cdot)$	the rational block Krylov subspace
$\mathcal{K}_m(\cdot, \cdot, \cdot)$	the matrix rational Krylov subspace
$\mathbb{V}_m, \mathbb{W}_m$	projection matrices computed using the block Lanczos algorithm
$\mathcal{V}_m, \mathcal{W}_m$	projection matrices computed using the global Lanczos algorithm
$\mathbb{V}_{2m}^e, \mathbb{W}_{2m}^e$	projection matrices computed using the extended block Lanczos algorithm

Abbreviations

LTI	Linear Time Invariant
SISO	Single input Single Output
MIMO	Multiple input Multiple Output
IRKA	Iterative Rational Krylov Method
AORBL	Adaptive Order Rational Block Lanczos
AMRBL	Adaptive Modified Rational Block Lanczos
AMRGL	Adaptive Modified Rational Global Lanczos
TRKS	Tangential Rational Krylov Subspace
EBLA	Extended Block Lanczos Algorithm
SVD	Singular value decomposition

Contents

Abstract	2
Résumé	4
Acknowledgements	7
Glossary	11
1 Introduction	19
1.1 Motivation de la thèse	19
1.2 Formulation du problème	20
1.3 Exemples motivants	23
1.3.1 Réacteurs chimiques: Contrôle de la température des réactifs	23
1.3.2 Système vibrationnel/système acoustique	25
1.3.3 Traquer une tempête dans l’océan pacifique	25
1.4 Contributions de la thèse	26

2	LTI System Theory and Model Reduction	29
2.1	Problem formulation and overview	31
2.1.1	The transfer function	32
2.1.2	Stability, controllability and observability of a dynamical system	34
2.1.2.1	Stability-Passivity	34
2.1.2.2	Controllability and observability	35
2.1.3	Controllability and Observability Gramians	36
2.2	Different dynamical system norms	38
2.2.1	The \mathcal{H}_2 norm	38
2.2.2	The Hankel norm	39
2.2.3	The \mathcal{H}_∞ -norm	40
2.3	Model reduction techniques	40
2.3.1	Lyapunov balanced truncation	41
2.3.1.1	The concept of balancing	41
2.3.1.2	Model reduction by balanced truncation	42
2.3.2	Model reduction via Krylov methods	44
2.3.2.1	Moment matching techniques	45
2.3.2.2	Approximation by moment matching	46
2.3.2.3	The non-symmetric block Lanczos-based method	47
2.3.2.4	The global Lanczos-based method	50

2.4	Contributions of the Dissertation	53
2.5	Outline of the Dissertation	55
3	An adaptive rational block Lanczos-type algorithm for model reduction of large scale dynamical systems	57
3.1	Multipoint rational interpolation	59
3.2	The rational block Lanczos method	61
3.2.1	The rational block Lanczos algorithm	61
3.2.2	Analysis of the rational block Lanczos algorithm	63
3.3	model reduction error	66
3.3.1	Residual error	66
3.3.2	An error expression for the transfer functions	67
3.3.3	Residual error expressions for the rational Lanczos algorithm	68
3.4	An adaptive-order rational block Lanczos-type algorithm	70
3.4.1	Interpolation point selection	70
3.4.2	An adaptive choice of interpolation points	71
3.5	The rational block Lanczos algorithm for the computation of the matrix exponential $e^{tA}B$	73
3.5.1	Polynomial approximation	74
3.5.2	Rational approximation	75
3.6	Numerical experiments	77
3.7	Conclusion	82

4	A modified nonsymmetric rational block Lanczos method for model reduction in large scale LTI dynamical systems	83
4.1	The modified rational block Lanczos algorithm	84
4.2	Lanczos-like equations for the rational block case	86
4.3	Residual error expressions	89
4.3.1	Simplified Lanczos residual errors in the rational block Lanczos	89
4.3.2	Error approximations	90
4.4	An adaptive modified rational block Lanczos algorithm	90
4.5	Numerical results	93
4.6	Conclusion	97
5	A matrix rational Lanczos method for model reduction in large scale first and second order dynamical systems	98
5.1	The modified rational global Lanczos method	99
5.1.1	The rational global Lanczos algorithm	100
5.1.2	The rational global Lanczos equations	101
5.1.3	Lanczos-like equations in the rational Global case	103
5.2	An adaptive modified rational global Lanczos algorithm	106
5.2.1	Adaptive choice of interpolation points	106
5.2.1.1	First approach	106
5.2.1.2	Second approach	108

5.2.2	The adaptive modified rational global Lanczos algorithm	109
5.3	Model Reduction of Second order Systems	110
5.3.1	Model order reduction techniques for second order systems	111
5.3.1.1	Second Order Structure-Preserving Krylov Techniques	113
5.3.1.2	Proportionally Damped Systems	114
5.4	Numerical experiments	117
5.5	Conclusion	126
6	An extended block Lanczos algorithm for model reduction of large scale dynamical MIMO systems	128
6.1	The extended block Lanczos algorithm	130
6.1.1	Description of the process	130
6.1.2	Theoretical results	136
6.2	Application to model reduction problem	143
6.3	Numerical experiments	146
6.4	Conclusion	150
7	Conclusions and Future Directions	152
7.1	Summary of results	152
7.2	Future directions	154
	Bibliography	155

1

Introduction

1.1 Motivation de la thèse

Les systèmes linéaires dynamiques ont été étudiés depuis longtemps et dans plusieurs domaines: la physique, la chimie, les mathématiques, l'ingénierie, et ainsi de suite. Dans le contexte d'ingénierie, les systèmes linéaires ont été largement étudiés depuis les années 1930 [3, 104, 107, 109, 115]. De nos jours, ces systèmes trouvent leurs applications dans d'autres domaines, comme le domaine de la bio-chimie et le domaine d'économie.

En général, les systèmes dynamiques proviennent directement par modélisation de phénomènes physiques. Ils proviennent également de la discrétisation spatiale des équations différentielles partielles (EDP) pour la simulation de systèmes de contrôle.

Dans la littérature, de nombreux travaux ont été publiés sur les systèmes linéaires, en particulier en ce qui concerne les parties théoriques fondamentales. Cependant, il y a une partie de la littérature relativement limitée pour les systèmes linéaires de grande taille qui se posent naturellement lorsqu'on modélise des structures complexes. En général, les systèmes linéaires de grande taille sont creuses et souvent structurés. Les méthodes itératives sont plus appropriées que les méthodes directes. Le problème c'est que ces modèles dynamiques sont souvent compliqués à étudier et contiennent un grand nombre de variables d'état. Ceci nous conduit à la notion de **réduction de modèles**.

La réduction de modèles est considérée comme un processus de description et simulation simplifiées de la dynamique d'un problème physique. Il y a une correspondance biunivoque entre la précision et le coût de calcul. Donc pour bien choisir la technique de réduction de modèles, il faut faire attention à cette correspondance et à la précision souhaitée. Une réduction de modèles permet de réduire le temps de calcul et réduire la mémoire utilisée et ceci passent par la réduction de nombre de variables d'état nécessaire.

Donc la motivation de cette thèse est la suivante: à partir des systèmes linéaires dynamiques de grande taille, on va développer de nouveaux algorithmes qui seront appliqués pour réduire la dimension de telle façon que le système réduit préserve les mêmes caractéristiques du système d'origine. C'est ainsi que le modèle simple obtenu sera utilisé à la place du modèle de départ.

1.2 Formulation du problème

Dans cette thèse, on va s'intéresser aux systèmes linéaires dynamiques de grande taille, invariants dans le temps et qui sont décrits par la forme suivante

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t); & x(t_0) = x_0 \\ y(t) = Cx(t) \end{cases} \quad (1.1)$$

où $x(t) \in \mathbb{R}^n$ est le vecteur d'état, $u(t), y(t) \in \mathbb{R}^p$ sont le vecteur d'entrée et le vecteur de sortie du système (1.1), respectivement. La matrice $A \in \mathbb{R}^{n \times n}$ est supposée être de grande taille et creuse et $B, C^T \in \mathbb{R}^{n \times p}$ ($p \ll n$). Le système (1.1) peut être aussi défini par

$$\Sigma = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right]. \quad (1.2)$$

La fonction de transfert $F(s)$ du système Σ est donnée par

$$F(s) = C(sI_n - A)^{-1}B.$$

Pour le cas particulier $p = 1$, le système dynamique est dit un système avec seule-entrée seule-sortie, et il est dit un système avec plusieurs-entrées plusieurs-sorties dans l'autre cas. Dans plusieurs applications, la dimension n du système d'origine est très grande ce qui rend les calculs très difficiles au niveaux de temps et de mémoire, et c'est la motivation de base du problème la réduction de modèles. Donc, l'objectif de la réduction de modèles est de remplacer le système (1.1) par un système dynamique d'ordre inférieur ayant la forme suivante

$$\begin{cases} \dot{x}_r(t) = A_r x_r(t) + B_r u(t) \\ y_r(t) = C_r x_r(t), \end{cases} \quad (1.3)$$

où $A_r \in \mathbb{R}^{r \times r}$, $B_r, C_r^T \in \mathbb{R}^{r \times p}$ et $r \ll n$, tel que le modèle d'ordre réduit préserve les mêmes caractéristiques du système d'origine comme la stabilité et la passivité. En plus, la sortie y_r du système réduit doit être proche à celle du modèle d'origine. On peut aussi définir le système (1.3) par

$$\Sigma_r = \left[\begin{array}{c|c} A_r & B_r \\ \hline C_r & 0 \end{array} \right], \quad (1.4)$$

sa fonction de transfert est

$$F_r = C_r(sI_r - A_r)^{-1}B_r.$$

En général, il existe deux catégories de méthodes pour la réduction de modèles: les méthodes basées sur la SVD et celles de Krylov. Une approche très connue pour la première catégorie est la méthode dite troncature équilibrée introduite par Mullis et Robert [109] et utilisée ultérieurement par Moore [107] pour les systèmes

et le contrôle. L'application de la méthode de troncature équilibrée sur un système stable préserve les propriétés théoriques importantes du système d'origine comme la stabilité et fournit une majoration de l'erreur. Cependant, cette méthode n'est pas adaptée pour les systèmes de grande taille.

Les méthodes de Krylov pour la réduction de modèles sont basées sur l'interpolation rationnelle de la fonction de transfert du système d'origine autour de quelques fréquences. Ces méthodes utilisent directement les bases biorthogonales d'un certain sous-espace de Krylov pour construire le modèle d'ordre réduit. Parmi ces méthodes, celles basées sur Arnoldi et Lanczos [3, 78, 81]. Malheureusement, la version standard de ces algorithmes a tendance de créer des modèles d'ordre réduit qui donne de mauvaise approximation. Pour traiter ce problème, des algorithmes d'Arnoldi et de Lanczos rationnels ont été proposés [1, 13, 43, 50, 71]. Ces procédures permettent d'obtenir un modèle d'ordre réduit tel que sa fonction de transfert approxime la fonction de transfert du système d'origine autour de plusieurs points d'interpolation. L'avantage des méthodes de sous-espace de Krylov est qu'elles traitent les problèmes de grande taille et qu'en plus elles s'implémentent d'une manière itérative et efficace. Dans la littérature, plusieurs travaux ont montré que la méthode de Lanczos est liée à l'approximation de Padé [24] qui est aussi une méthode utile pour réduire la dimension des systèmes de grande taille [25, 26, 49]. L'inconvénient de cette approche c'est qu'elle ne préserve pas la stabilité du système d'origine. Pour résoudre ce problème, un approximant de type Padé partiel a été introduit pour préserver les caractéristiques principales du système d'origine comme la stabilité et la passivité [10].

Dans cette thèse, on va utiliser les méthodes de Krylov rationnel pour réduire la dimension des systèmes linéaires dynamiques de grande taille. Plus précisément, on va proposer des algorithmes de type-Lanczos rationnel pour construire deux matrices $V, W \in \mathbb{R}^{n \times r}$ avec $W^T V = I_r$ et telle que les matrices du système réduit sont données par

$$A_r = W^T A V, \quad B_r = W^T B, \quad C_r = C V.$$

Le problème majeur des méthodes de Krylov rationnel est la construction d'un ensemble de points d'interpolation que l'on doit utiliser pour construire les espaces de Krylov rationnel. Cette procédure n'est pas automatique et les points

d'interpolation doivent être convenablement choisis pour avoir de bonnes approximations et garantir une bonne convergence de la procédure. Plusieurs techniques pour bien choisir ces paramètres vont être aussi proposées dans ce travail.

1.3 Exemples motivants

Dans cette section, on va donner quelques exemples d'applications de systèmes dynamiques de grande dimension. Ces types de systèmes sont utilisés pour la simulation et le contrôle. Pour plus d'exemples voir [3, 124]

1.3.1 Réacteurs chimiques: Contrôle de la température des réactifs

Le premier exemple est un système qui apparaît lors de l'optimisation de la température (chauffage/refroidissement) d'un écoulement fluide dans un tube. L'application potentielle serait la régulation de la température de certaines entrées de réactif dans un réacteur chimique. Les équations du modèle sont:

$$\begin{aligned} \frac{\partial x}{\partial t} - \kappa \Delta x + v \cdot \nabla x &= 0 && \text{sur } \Omega \\ x &= x_0 && \text{sur } \Gamma_{in} \\ \frac{\partial x}{\partial n} &= \sigma(u - x) && \text{sur } \Gamma_{heat_1} \cup \Gamma_{heat_2} \\ \frac{\partial x}{\partial n} &= 0 && \text{sur } \Gamma_{out} \end{aligned}$$

Ici Ω désigne le domaine rectangulaire représenté sur la figure 1.1. Le flux entrant Γ_{in} est du côté gauche du domaine, tandis que le flux sortant Γ_{out} est à la frontière droite. Nous pouvons considérer ce domaine juste en dimension 2, en supposant une symétrie rotationnelle ce qui est équivalent à supposer un écoulement non-turbulent. Les matrices tests ont été créés en utilisant le logiciel COMSOL4 multi-physique, leurs dimensions est 1090. Le système est doté d'une seule entrée

appliquée aux limites supérieures et inférieures vu sa symétrie rotationnelle. Les trois données de sortie correspondent à la température de l'écoulement du flux à la sortie. Notons que dans ce cas, nous avons un domaine convexe qui nous permet d'utiliser les points d'évaluation comme des sorties.

En utilisant une discrétisation spatiale par éléments fini, le modèle semi-discret s'écrit sous la forme suivante:

$$\begin{cases} M\dot{x} = \tilde{A}x + \tilde{B}u(t) \\ y(t) = \tilde{C}x \end{cases} \quad (1.5)$$

En supposant que la matrice M est inversible, ce système peut être représenté sous la forme standard (1.1).

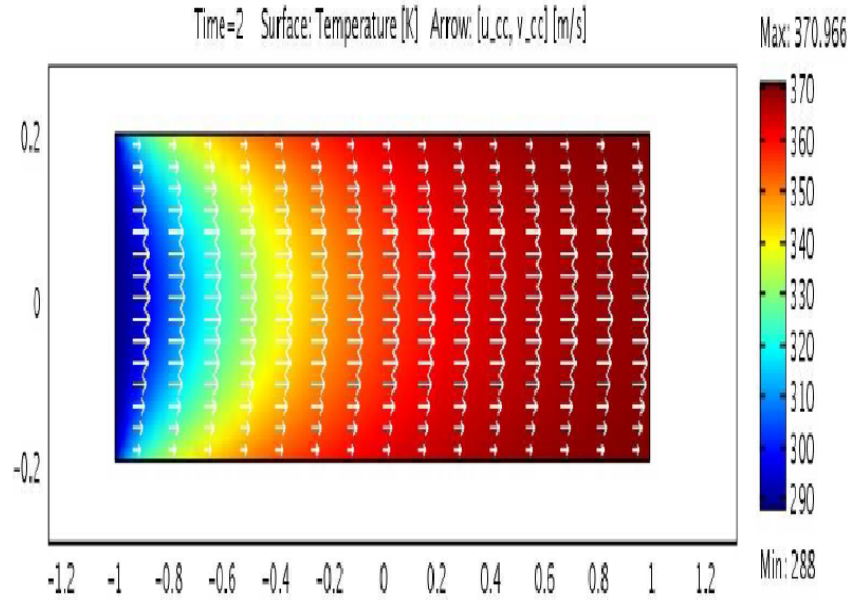


Figure 1.1: Domain Ω for the Inflow Example: A 2d cross-section of the liquid flow in a round tube.

1.3.2 Système vibrationnel/système acoustique

Considérons un pare-brise d'une voiture soumise à une accélération. Le problème consiste à calculer le bruit généré en des points en dehors de la fenêtre de la voiture. Le premier pas dans la résolution de ce problème est l'EDP décrivant la déformation du pare-brise constitué d'un matériau donné. La discrétisation par éléments finis donne 7564 noeuds (3 couches de 60×30 éléments), pour un matériau constitué de verre avec un module de Young égale à $7.107N/m^2$, une densité de $2490kg/m^3$ et un facteur de poisson de 0.23. Ces paramètres aident à déterminer expérimentalement les coefficients du modèle résultant par les éléments finis. Enfin le pare-brise subit une force en un point donné, et l'objectif est donc de calculer le déplacement de ce point.

La discrétisation par éléments finis de cet exemple mène à l'équation du second ordre suivante

$$M \frac{d^2}{dt^2} x(t) + C \frac{d}{dt} x(t) + Kx(t) = f(t),$$

où la dimension du problème discrétisé est $n = 22692$, x représente la position, et $\frac{d}{dt}x$ est la vitesse du pare-brise au point choisi. Les matrices M, C et K sont respectivement les matrices de masse, d'amortissement et de raideur. Comme il s'agit d'un système de second ordre, sa complexité est deux fois plus élevée (45 384 états).

1.3.3 Traquer une tempête dans l'océan pacifique

Le problème consiste à étudier la sensibilité de l'équilibre de l'atmosphère face aux perturbations. En particulier, nous souhaitons déterminer la perturbation initiale qui génère la plus grande perturbation dans un intervalle de temps spécifié. Ces perturbations sont gouvernées par les équations de ORR-Sommerfield. En supposant des perturbations harmoniques de la vitesse du vent de la forme $\Phi(x, y, t) = \phi(y, t)e^{ikx}$, on a

$$\frac{\partial \phi(y, t)}{\partial t} = A\phi(y, t) = -iky \frac{\partial^2 \phi(y, t)}{\partial y^2} + \frac{1}{Re} \left(\frac{\partial^2 \phi(y, t)}{\partial y^2} - k^2 \phi(y, t) \right)^2,$$

où Re désigne le nombre de Reynolds. La discrétisation en variable y mène à l'ensemble des ODEs suivant :

$$\frac{d\hat{\phi}(t)}{dt} = \hat{A}\phi(t), \quad \hat{A} \in \mathbb{R}^{n \times n}.$$

On suppose que ce système est influencé par les perturbations, en particulier on suppose que

1. Les entrées aléatoires affectent toutes les variables $\hat{\phi}_i$,
2. Toutes ces variables sont observables.

Le système discrétisé est donc un système linéaire ayant le même nombre d'entrée m , des vecteurs d'état n , et des sorties p . i.e,

$$\Sigma \equiv \left[\begin{array}{c|c} \hat{A} & I_n \\ \hline I_n & 0 \end{array} \right] \Rightarrow \mathbf{m} = \mathbf{p} = \mathbf{n}.$$

De tels modèles sont utilisés pour traquer les tempêtes dans les moyennes altitudes de l'océan pacifique [47].

1.4 Contributions de la thèse

L'objectif de ce travail est maintenant clair: on va proposer des algorithmes basés sur la méthode de Lanczos rationnel, puis on va les appliquer pour réduire la dimension des systèmes linéaires dynamiques de grande taille. La thèse est présentée selon le plan suivant:

Dans le deuxième chapitre, on va présenter tous les outils nécessaires pour les systèmes linéaires, ainsi on va expliquer les idées principaux pour la réduction de modèles. On va rappeler deux approches pour traiter le problème de la réduction

de modèles, qui sont la méthode de la troncature équilibrée et la méthode basée sur les sous-espaces de Krylov.

Dans le troisième chapitre, vu l'objectif de la réduction de modèles et puisque les méthodes de Krylov rationnel sont plus efficaces que les méthodes standards, on va commencer par proposer un algorithme de type Lanczos rationnel par bloc pour construire les deux bases V et W de l'espace de Krylov rationnel. Après, on va établir quelques équations rationnelles qui décrivent la relation entre ces bases et la matrice A du système d'origine. En suite, on va utiliser ces équations pour établir une expression de l'erreur entre la fonction de transfert du système d'origine et celle du système réduit. Une approche adaptative pour choisir les points d'interpolation va être aussi introduite. Finalement, on va montrer l'efficacité des méthodes proposées en donnant des résultats numériques et des comparaisons entre notre méthode et la célèbre approche IRKA ("Iterative Rational Krylov Algorithm").

Dans le quatrième chapitre, on va proposer un algorithme de Lanczos rationnel par bloc modifié. Cette procédure peut être considérée comme une généralisation de l'algorithme proposé dans Chapitre 3 où plusieurs multiplicités sont considérées pour chaque point d'interpolation. L'avantage de l'algorithme modifié est que les équations simples de Lanczos standard restent vraies aussi dans le cas rationnel. Après, on va utiliser ces équations pour établir des expressions simples de l'erreur résiduelle. Comme les méthodes de Krylov rationnel sont toujours liées à l'ensemble des points d'interpolation, on va proposer à nouveau une technique adaptative pour choisir ces paramètres.

Chapitre 5 représente une autre extension des méthodes de sous-espace de Krylov pour les systèmes avec plusieurs-entrées plusieurs-sorties, qui est la méthode de sous-espace de Krylov global. On va commencer par introduire l'algorithme général de Lanczos rationnel global, puis établir les équations rationnelles qui décrivent cette procédure. Ensuite, on va modifier cette algorithme de telle façon que les équations simples de Lanczos global restent vraies aussi dans le cas rationnel. Quelques techniques pour choisir les points d'interpolations vont être aussi proposées. Dans la deuxième partie de ce chapitre, on va traiter les systèmes

linéaires dynamiques de deuxième ordre et on va appliquer l'algorithme de Lanczos rationnel global modifié sur les modèles du premier ordre correspondant. Finalement, quelques résultats numériques vont être donnés.

Toujours dans le contexte des méthodes de type Lanczos rationnel pour la réduction de modèles, on va proposer aussi dans le sixième chapitre un algorithme de Lanczos étendu par bloc pour construire deux bases biorthogonales pour le sous-espace de Krylov étendu. L'avantage de cette méthode est qu'on n'a pas à construire les points d'interpolation comme les autres méthodes de Krylov rationnel. Après avoir décrit la procédure de cette méthode, on va montrer un ensemble de propriétés algébriques, puis on va appliquer l'algorithme de Lanczos étendu par block sur le problème de la réduction de modèles. La dernière section de ce chapitre est consacrée à quelques résultats numériques et comparaison avec l'approche IRKA.

Dans le dernier chapitre (Chapitre 7), on va donner un résumé de tous les résultats proposés, ainsi que des idées à explorer dans les futurs travaux.

2

LTI System Theory and Model Reduction

Nowadays, abundant natural laws and phenomena can be described (or approximated) by linear systems. For example, in chemical engineering there are heat transfer or reaction, convection, diffusion models; in mechanical engineering there are wave propagation or vibration models; in electrical engineering, circuit simulation and design, amplifier or filter design, digital signal processing . . . All require linear system theories. These mathematical models can be used to simulate the behavior of the processes in question. Sometimes, they are also used to modify or control the behavior of the processes. The weather, on the one hand, and very large scale integration (VLSI) circuits, on the other, constitute examples of such processes, the former physical and the latter artificial. Furthermore, these are dynamical systems, as their future behavior depends on their past evolution. Yet for all mentioned applications, the accuracy and reliability of the model plays an important role. The better the model describes reality, the better the expectable

results from simulation, and the more likely predictions apply, . . . Increasing demands on the accuracy, however, typically bring about higher complexity of the model which may complicate or even inhibit the fulfilment of the given task due to limitations of memory and/or computational capacity. This phenomena leads to concept of `model reduction`.

Then, the motivation of this thesis is: starting from large scale linear dynamical systems, we develop new algorithms that will be applied to reducing the original system to a lower dimensional system that has same response characteristics and capture the main features of the original complex model. This need arises from limited computational, accuracy, and storage capabilities. The simplified model is then used in place of the original complex model, for either simulation or control. Thereby, efficiency can be dramatically increased, as comparable results can be produced in far less time.

A dynamical system has input and output variables. The output variables can be measured while the input variables can influence the outputs of the system and could be controlled to give more interesting properties to the dynamical system. Controlling a system according to measurements of the output variables is called feedback. It needs the knowledge of the state variables, assumed to be known. They can be estimated by a special system called the observer. Generally, a dynamical system comes from the discretization (in the space) of partial differential equations (PDE's). Attention is often devoted to the classical approximation of complex dynamic systems, and the first type of approximation is devoted to obtaining linearized time-invariant models from non-linear, distributed, or time-variant systems. Indeed, even a linear time-invariant system derived from this type of approximation is often too complicated to be investigated due to the large number of state variables that are included.

In this thesis, we will focus on continuous Linear Time Invariant (LTI) dynamical systems.

2.1 Problem formulation and overview

A continuous linear dynamical system can be expressed via the following state-space form

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t); & x(t_0) = x_0 \\ y(t) = & Cx(t) \end{cases} \quad (2.1)$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t), y(t) \in \mathbb{R}^p$ are the input and the output vectors of the system (2.1), respectively. The system matrices $B, C^T \in \mathbb{R}^{n \times p}$ and $A \in \mathbb{R}^{n \times n}$ are assumed to be large and sparse. When $p = 1$, the dynamical system is called Single-Input Single-Output (SISO), and is called Multiple-Input Multiple-Output (MIMO) otherwise. The control problem consists of acting on the input vector $u(t)$ so that the output vector $y(t)$ has a desirable time trajectory. Modifying the input $u(t)$ according to the output $y(t)$ which is observed or to the state $x(t)$ is called feedback. The LTI dynamical system (2.1) can be also denoted as

$$\Sigma = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right]. \quad (2.2)$$

When the dimension n of the original system is very large, as stated above, it is not practical to use the full system for simulation or run-on-time control, so the concept of model reduction was introduced. The reduced order dynamical system can be stated as follows

$$\begin{cases} \dot{x}_r(t) = A_r x_r(t) + B_r u(t) \\ y_r(t) = C_r x_r(t), \end{cases} \quad (2.3)$$

where $A_r \in \mathbb{R}^{r \times r}, B_r, C_r^T \in \mathbb{R}^{r \times p}$ and $r \ll n$. The system (2.3) can be also expressed as

$$\Sigma_r = \left[\begin{array}{c|c} A_r & B_r \\ \hline C_r & 0 \end{array} \right]. \quad (2.4)$$

The reduced order dynamical system (2.3) should be constructed such that

- The output $y_r(t)$ of the reduced system approaches the output $y(t)$ of the

original system.

- Some properties of the original system are preserved, such as passivity and stability.
- The computation methods are steady and efficient.

2.1.1 The transfer function

The state space representation is usually referred as an internal representation of a dynamical system because it involves the state variables x which are internal variables of the system. The input/output representation, also called external representation, is obtained by eliminating the state vector, between the state equation and the output equation with zero initial conditions. To get the frequency domain description we apply the Laplace transform

$$L(f)(s) := \int_0^{\infty} e^{-st} f(t) dt$$

to (2.1), we obtain

$$\begin{cases} sX(s) &= AX(s) + BU(s) \\ Y(s) &= CX(s), \end{cases}$$

where $X(s), Y(s)$ and $U(s)$ are the Laplace transforms of $x(t), y(t)$ and $u(t)$, respectively. If we eliminate $X(s)$ in the previous two equations we obtain $Y(s) = F(s)U(s)$, where $F(s)$ is called the transfer function of the system (2.1) and defined as

$$F(s) = C(sI_n - A)^{-1}B. \quad (2.5)$$

We will see later that most of the model reduction techniques are based on this simple looking transfer function (especially the moment matching approach), since what one concerns most is the output of a system under different inputs.

Actually in the frequency domain there are infinity many state-space descriptions for a given linear system with given input and output, to see this, we introduce the following definition:

Definition 2.1.1 Two LTI dynamical systems $\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right]$ and $\left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & 0 \end{array} \right]$ are called equivalent if they have the same transfer function, i.e.,

$$\tilde{F}(s) = \tilde{C}(sI_n - \tilde{A})^{-1}\tilde{B} = C(sI_n - A)^{-1}B = F(s), \quad \forall s \in \mathbb{C}.$$

It is easy to verify that for any non-singular $n \times n$ matrix T , the LTI system $\left[\begin{array}{c|c} T^{-1}AT & T^{-1}B \\ \hline CT & 0 \end{array} \right]$ is equivalent to the LTI system $\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right]$. Hence, if the main concern is the outputs under some specific inputs, we have many choices of the state-space description. The choice of the matrix T is very important and the states are connected by the relation $x(t) = T\tilde{x}(t)$.

If we solve the state equation (2.1) with initial condition $x_0 = x(t_0)$, we get

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau. \quad (2.6)$$

We notice that this solution is a sum of two terms: the first term $e^{A(t-t_0)}x(t_0)$ represents the state evolution of the autonomous system ($u = 0$) while the second term $\int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau$ corresponds to the state evolution for the zero initial condition. This last term written as a convolution product of $e^{At}B$ with $u(t)$ is called the input-to-state impulse matrix. From (2.1) and (2.6), the output response $y(t)$ is given by

$$y(t) = Ce^{A(t-t_0)}x(t_0) + \int_{t_0}^t Ce^{A(t-\tau)}Bu(\tau)d\tau. \quad (2.7)$$

The state-space approach has many advantages: it can be obtained directly from a real problem as in PDE control problems where the matrix A comes from the spatial discretization of the PDE, the control gives the input $u(t)$, the boundary conditions provide the matrix B while C is obtained from measurements of the output. The transfer function $F(s)$ relates the Laplace transform of the output vector to that of the input vector. Each entry $F_{ij}(s)$ is a rational function representing the transfer function between the i -th input and the j -th output, all other inputs being set equal to zero.

2.1.2 Stability, controllability and observability of a dynamical system

2.1.2.1 Stability-Passivity

we start by given the definition of a stable dynamical system.

Definition 2.1.2 *The LTI dynamical system (2.1) is*

- *Asymptotically stable, if and only if A is stable ($\Lambda(A) \subset \mathbb{C}^{-1}$).*
- *Stable, if and only if all eigenvalues of A have non-positive real parts, and in addition, all pure imaginary eigenvalues have multiplicity one.*

An important property of a dynamical system is its stability which means the ability of the autonomous system ($u = 0$) to recover its equilibrium point after being disturbed from it. Roughly speaking, stability means that for bounded inputs $u(t)$, the state-variable vector $x(t)$ will remain bounded for all times t . A stable system tends to return to its equilibrium state when perturbed from it. Conversely, perturbations are increased by an unstable system. Formally, the dynamical system (2.1) is asymptotically stable if for any initial condition $x(t_0)$, we have

$$\lim_{t \rightarrow \infty} x(t) = 0. \quad (2.8)$$

Using the expression (2.6) with $u = 0$, the limit condition (2.8) holds if and only if the matrix A has all its eigenvalues in the open left-half plane \mathbb{C}^{-1} . In this case the matrix A is called stable or Hurwitz. There are several properties associated with stability. Clearly, if A is stable, then also A^{-1} and A^T are stable. Moreover, if the product of matrices AB is stable, then also BA can be shown to be stable. It is also clear that, due to the relation between eigenvalues of A and poles of the transfer function, stability can also be formulated in terms of the poles of the transfer function $F(s)$. A stable structure can become unstable if non-linear components are connected to it. Another property called passivity is more

stronger than stability. An LTI system is passive if it is incapable of generating energy. The passivity of the transfer function $F(s)$ is defined as follows.

Definition 2.1.3 (*Passivity*) *A stable LTI system is called passive if:*

- F has no pole in \mathbb{C}^+ .
- $\operatorname{Re}(F(s)) \geq 0 \quad \forall s \in \mathbb{C}^+$,

in this case F is also called positive-real.

2.1.2.2 Controllability and observability

The controllability of a dynamical system is related to the ability of the system to attain a given state under the action of an appropriate control signal. If a state is not controllable, then it is not possible to move this state to another one by acting on the control input. If the matrix representing the dynamics of a non controllable state is stable, then the state is said to be stabilizable. The observability is related to the possibility of evaluating the state of a system through output measurements. The notions of controllability and observability are due to Kalman [89, 90].

Definition 2.1.4 (*Controllability*) *An LTI dynamical system is called controllable if starting from zero initial state, any state can be reached via a suitable control within finite time, i.e., given any state $z \in \mathbb{R}^n$, starting from $x(t_0)$, there exists $u(t)$ such that at time t , $x(t) = z$.*

Proposition 2.1.1 *The LTI dynamical system (2.1) is controllable if and only if the controllability matrix*

$$\mathcal{C} = [B, AB, A^2B, \dots, A^{n-1}B]$$

is of full rank, i.e; $\operatorname{rank}(\mathcal{C}) = n$. In this case the pair (A, B) is said to be controllable.

If $\text{rank}(\mathcal{C}) = k < n$, then $n - k$ is the number of the uncontrollable modes (the eigenvalues of the matrix A satisfying $\text{rank}([\lambda I - AB]) < n$). If all uncontrollable modes are stable then the system is said to be stabilizable.

Definition 2.1.5 (*Observability*) Given the dynamical system, the system is called observable if without control, different initial states lead to different outputs, i.e., when $u(t) = 0$, $y(t)$ is uniquely determined by $x(t_0)$.

Proposition 2.1.2 The LTI dynamical system (2.1) is observable if and only if the observability matrix

$$\mathcal{O} = [C^T, A^T C^T, (A^2)^T C^T, \dots, (A^{n-1})^T C^T]^T$$

is of full rank, i.e; $\text{rank}(\mathcal{O}) = n$. In this case the pair (A, C) is said to be observable. If $\text{rank}(\mathcal{O}) = l < n$, then $n - l$ is the number of the unobservable modes (the eigenvalues of the matrix A satisfying $\text{rank}([\lambda I - A^T C^T]^T) < n$).

The observability is linked to the possibility of evaluating the state of a system through output measurements. If a state is not observable there is no way to determine its evolution. If the dynamics of a non observable state is stable, then the state is said to be detectable.

Proposition 2.1.3 A stable LTI system (2.1) is controllable if and only if the controllability Gramian given by (2.9) is positive definite and it is observable if and only if the observability Gramian (2.10) is positive definite.

2.1.3 Controllability and Observability Gramians

We assume that the LTI dynamical system is stable.

Definition 2.1.6 The controllability Gramian associated to the LTI system (2.1) is defined as

$$P = \int_0^{\infty} e^{tA} B B^T e^{tA^T} dt, \quad (2.9)$$

and the observability Gramian is defined by

$$Q = \int_0^{\infty} e^{tA^T} C^T C e^{tA} dt. \quad (2.10)$$

By applying Parseval's relation (or Plancherel Theorem) [147], we get another integral for P and Q (these formulas are also proved in [19]):

$$P = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (j\omega I - A)^{-1} B B^T (-j\omega I - A^T)^{-1} d\omega, \quad (2.11)$$

$$Q = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (-j\omega I - A^T)^{-1} C^T C (j\omega I - A)^{-1} d\omega. \quad (2.12)$$

The two Gramians are the unique solutions of the following coupled Lyapunov matrix equations

$$AP + PA^T + BB^T = 0, \quad (2.13)$$

$$A^T Q + QA + C^T C = 0. \quad (2.14)$$

As can be seen from the expressions (2.9) and (2.10), the Gramians P and Q are at least positive semi-definite.

The following theorem strengthens semi-definiteness by relating definiteness to controllability and observability.

Theorem 2.1.1 *The LTI system (2.1) is controllable if and only if the solution P of (2.13) is positive definite; it is observable if and only if the solution Q of (2.14) is positive definite.*

We will see later that the product PQ plays an important role in model reduction.

Consider the new equivalent LTI dynamical system

$$\tilde{\Sigma} = \left[\begin{array}{c|c} T^{-1}AT & T^{-1}B \\ \hline CT & 0 \end{array} \right],$$

where T is a non-singular matrix. Then the associated controllability and observability Gramians \tilde{P} and \tilde{Q} are expressed as

$$\tilde{P} = \int_0^{\infty} e^{t\tilde{A}} \tilde{B} \tilde{B}^T e^{t\tilde{A}^T} dt,$$

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

where $\tilde{A} = T^{-1}AT$, $\tilde{B} = T^{-1}B$ and $\tilde{C} = CT$. Hence, we obtain

$$\tilde{P} = T^{-1}PT^{-T} \quad \text{and} \quad \tilde{Q} = T^TQT, \quad (2.15)$$

and these last relations show that the Gramians of two equivalent LTI systems are not similar. However, the similarity is preserved for the product of the controllability and observability Gramians and we have

$$\tilde{P}\tilde{Q} = T^{-1}PQT \quad \text{and} \quad \tilde{Q}\tilde{P} = T^TQPT^{-T}.$$

Hence the eigenvalues of PQ are invariant under state-space transform, these eigenvalues turn out to be the very important **Hankel singular values** (to be defined latter).

2.2 Different dynamical system norms

System norms play an important role in the analysis of LTI systems, as they quantify certain properties of the model and also they are used to measure the accuracy of the reduced order model. In this thesis, we will concentrate on the \mathcal{H}_∞ norm, other norms like the \mathcal{H}_2 norm and the Hankel norm will be also introduced in this section; see [3] for more details.

2.2.1 The \mathcal{H}_2 norm

We start by recall the definition of the \mathcal{H}_2 -norm of the transfer function $F(s)$ associated to the dynamical system Σ .

Definition 2.2.1 *The \mathcal{H}_2 -norm of $F(s)$ is defined as*

$$\|F(\cdot)\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \text{trace}[F(j\omega)^T F(j\omega)] d\omega, \quad (2.16)$$

where j is the complex number $j^2 = -1$.

Consider the impulse response $g(t) = \mathcal{L}^{-1}[F(s)] = Ce^{tA}B$ where \mathcal{L} is the Laplace transform

$$\mathcal{L}(g)(s) = \int_0^\infty g(t)e^{-st} dt = F(s).$$

Then using the Parseval relation

$$\int_0^{\infty} \text{trace}[g(t)^T g(t)] dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \text{trace}[F(j\omega)^T F(j\omega)] d\omega,$$

the \mathcal{H}_2 norm can be also expressed as

$$\|F(\cdot)\|_{\mathcal{H}_2}^2 = \int_0^{\infty} \text{trace}[g(t)^T g(t)] dt. \quad (2.17)$$

Therefore, the \mathcal{H}_2 norm could be calculated as follows

$$\|F(\cdot)\|_{\mathcal{H}_2}^2 = \text{trace} \left[B^T \left(\int_0^{\infty} e^{tA^T} C^T C e^{tA} \right) B \right]. \quad (2.18)$$

Setting

$$Q = \int_0^{\infty} e^{tA^T} C^T C e^{tA} dt, \quad (2.19)$$

we get

$$\|F(\cdot)\|_{\mathcal{H}_2}^2 = \text{trace}(B^T Q B). \quad (2.20)$$

Assuming that A is a stable matrix, the observability Gramian Q can be computed by solving the Lyapunov matrix equation (2.13). We notice that in a similar way, the \mathcal{H}_2 norm can be computed by using the controllability Gramian defined by (2.9). Therefore, \mathcal{H}_2 norm can be expressed as

$$\|F(\cdot)\|^2 = \text{trace}(C P C^T).$$

2.2.2 The Hankel norm

The Hankel singular values of a stable LTI system are the square roots of the product of the controllability and observability Gramians:

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

where P and Q are the Gramians associated to the LTI dynamical system (2.1).

Definition 2.2.2 *The Hankel norm of a stable LTI dynamical system is the largest system Hankel singular value of the associated Hankel operator of this system, i.e.,*

$$\|F\|_{\mathcal{H}} = \max_{i=1,2,\dots,n} \sigma_i(F).$$

2.2.3 The \mathcal{H}_∞ -norm

An other important norm in linear system theory is the well known \mathcal{H}_∞ -norm, which is related to the Hardy space \mathcal{H}_∞ .

Definition 2.2.3 *The \mathcal{H}_∞ norm of the transfer function $F(\cdot)$ is defined as*

$$\|F(\cdot)\|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} \sigma_{max}(F(j\omega)), \quad (2.21)$$

where σ_{max} denotes the largest singular value. To approximate the \mathcal{H}_∞ -norm, we choose a set of frequencies $\Omega_n = \omega_1, \omega_2, \dots, \omega_n$ and search for

$$\sup_{1 \leq k \leq n} \sigma_{max}(F(j\omega_k)) \approx \|F(\cdot)\|_{\mathcal{H}_\infty}.$$

2.3 Model reduction techniques

There are two well known model reduction methods for MIMO systems which are currently in use, SVD based methods and Krylov (moment matching) based methods. The key steps in the first category are the computation of the so-called Hankel singular values and balancing of the system. In general terms, balancing consists of simultaneously diagonalization of two appropriate chosen positive definite matrices [3], according to solutions of Lyapunov equations or Riccati equations. One of the most common approach of the SVD-based methods is the so-called Balanced truncation model reduction [96, 126, 107]. This method have nice system theoretical properties, such as preservation of stability and computation of an error bound. However, they are not suited for large scale systems [3]. This drawback stems from the fact that they require dense matrix factorizations, such as solving two Lyapunov equations, and therefore the computational cost on the order $\mathcal{O}(n^3)$ and storage of order $\mathcal{O}(n^2)$ becomes impractical for systems of order $n \ll 1000$. However, Krylov-subspace methods [8, 13, 70, 71, 85] based on moment matching have the advantage for large problems. This is due to the fact that they need only matrix-vector operations and no decomposition of large matrices is required. They require $\mathcal{O}(nr^2)$ operations for sparse problems and the requirement of memory is about $\mathcal{O}(rn)$ where n is the size of the original problem

and r is the size of the reduced system. Unfortunately, these methods lack good theoretical properties; see [3].

2.3.1 Lyapunov balanced truncation

One of the most commonly used model reduction scheme is the so-called Balanced Truncation model reduction, which was first introduced by Mullis and Robert [109] and later in the systems and control literature by Moore [107]. The approximation theory underlying this approach was developed by Glover [66, 67]. Several researchers have recognized the importance of balanced truncation for model reduction because of its theoretical properties. Computational schemes for small to medium scale problems already exist. However, the development of computational methods for large scale setting is still an active area research.

2.3.1.1 The concept of balancing

We assume here that the LTI system is stable, controllable and observable (in this case we call it also stable and minimal). Then the controllability and observability Gramians are unique positive definite. The concept of balanced truncation is to transform the original LTI system to an equivalent one in which the states that are difficult to reach are also difficult to observe. This reduces to finding a non-singular matrix T such that the new Gramians \tilde{P} and \tilde{Q} given by (2.15) are equal. We consider the controllability and observability Gramians P and Q of the original system (2.1). The square roots of the eigenvalues of the product PQ are the Hankel singular values of the LTI system Σ :

$$\sigma_i = \sqrt{\lambda_i(PQ)}.$$

Definition 2.3.1 *The reachable, observable and stable system $F(s)$ is called Lyapunov-balanced if*

$$P = Q = \text{diag}(\sigma_1, \dots, \sigma_n), \quad (2.22)$$

where σ_i is the i -th Hankel singular value of the LTI system.

The following construction of the balanced transform T by the Cholesky factors of P and Q was first discussed in [96]. Suppose we have the Cholesky factorisation of the controllability and observability Gramians P and Q such that

$$P = L_c L_c^T, \quad Q = L_o L_o^T, \quad (2.23)$$

where L_o, L_c are lower triangular matrices. Compute the singular value decomposition of $L_c^T L_o$ as

$$L_c^T L_o = Z \Sigma Y^T, \quad (2.24)$$

where Z and Y are unitary $n \times n$ matrices and Σ is a tridiagonal matrix containing the singular values. Let T be defined as

$$T = L_c Z \Sigma^{-1/2}, \quad T^{-1} = \Sigma^{-1/2} Y^T L_o^T, \quad (2.25)$$

then it can be verified by direct calculation that the Gramians \tilde{P} and \tilde{Q} are diagonal and equal, i.e.

$$\tilde{P} = \tilde{Q} = \Sigma,$$

where Σ is a diagonal matrix whose elements are the Hankel singular values $\sqrt{\lambda_i(PQ)}$ since PQ is similar to $\tilde{P}\tilde{Q}$. There are other possible way for the construction of the matrix T . It was remarked by Glover [66] that the balanced transformation is not unique but unique up to a non-singular transformation.

2.3.1.2 Model reduction by balanced truncation

As the concept of balancing truncation method has the property that the states which are difficult to reach are simultaneously difficult to observe. Then, a reduced model is obtained by truncating the states which have this property, i.e., those which correspond to small Hankel singular values σ_i . We have the following theorem.

Theorem 2.3.1 *Assume that the LTI dynamical system (2.1) is stable, minimal*

and having the following balanced realization

$$\tilde{\Sigma} = \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & 0 \end{array} \right],$$

with $P = Q = \text{diag}(\Sigma_m, \tilde{\Sigma}_m)$, where $\Sigma_m = \text{diag}(\sigma_1, \dots, \sigma_m)$ and $\tilde{\Sigma}_m = \text{diag}(\sigma_{m+1}, \dots, \sigma_n)$.

Then, the reduced order model represented by

$$\tilde{\Sigma}_m = \left[\begin{array}{c|c} A_{11} & B_1 \\ \hline C_1 & 0 \end{array} \right]$$

is asymptotically stable, minimal and satisfies

$$\|F(\cdot) - F_m(\cdot)\|_{\mathcal{H}_\infty} \leq 2(\sigma_{m+1} + \dots + \sigma_n). \quad (2.26)$$

The equality holds if $\tilde{\Sigma}_m$ contains only σ_n .

The preceding theorem shows that if the neglected singular values $\sigma_{m+1}, \dots, \sigma_n$ are small, then the reduced order LTI system is close to the original one. Note that the inequality (2.26) is an a priori error bound. Then given a error tolerance, one can decide how many states to truncate without forming the reduced model.

Balanced truncation technique can be applied to any Σ which is asymptotically stable and minimal. For application of balancing to unstable and non-minimal systems, see [34, 91, 136, 149] and the references therein.

Now, let us see how to construct the low order model Σ_m . We define the matrices

$$W_m = L_o Y_m \Sigma_m^{-1/2} \quad \text{and} \quad V_m = L_c Z_m \Sigma_m^{-1/2}, \quad (2.27)$$

where $\Sigma_m = \text{diag}(\sigma_1, \dots, \sigma_m)$ and Z_m, Y_m correspond to the leading m columns of the matrices Z, Y given by the singular value decomposition (2.24), respectively. We can easily verified that $W_m^T V_m = I_m$ and then also that $V_m W_m^T$ is an oblique projector. The matrices of the reduced LTI system

$$\tilde{\Sigma}_m = \left[\begin{array}{c|c} A_m & B_m \\ \hline C_m & 0 \end{array} \right],$$

are given by

$$A_m = W_m^T A V_m, \quad B_m = W_m^T B \quad \text{and} \quad C_m = C V_m. \quad (2.28)$$

The use of Cholesky factors in the Gramians P and Q is not applicable for large-scale problems. Instead, and as we will see later, one can compute low rank approximations of P and Q and use them to construct an approximate balanced truncation model.

Let \tilde{A} , \tilde{B} and \tilde{C} be the following matrices

$$\tilde{A} = \left(\begin{array}{c|c} A & 0 \\ \hline 0 & A_m \end{array} \right), \quad \tilde{B} = \begin{pmatrix} B \\ B_m \end{pmatrix}, \quad \tilde{C} = \begin{pmatrix} C & C_m \end{pmatrix}.$$

Then, the Gramians corresponding to the error dynamical system

$$\tilde{\Sigma} = \left(\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & 0 \end{array} \right),$$

are the solutions of the following Lyapunov matrix equations

$$\tilde{A}\tilde{P} + \tilde{P}\tilde{A}^T + \tilde{B}\tilde{B}^T = 0,$$

and

$$\tilde{A}^T\tilde{Q} + \tilde{Q}\tilde{A} + \tilde{C}^T\tilde{C} = 0.$$

Therefore, the Hankel norm of the error can be expressed as

$$\|F(s) - F_m(s)\|_{\mathcal{H}} = \sqrt{\lambda_{\max}(\tilde{P}\tilde{Q})}.$$

2.3.2 Model reduction via Krylov methods

An important class of numerical methods for model reduction is the Krylov-based model reduction. Unlike the SVD based methods, stability of the reduced model constructed by Krylov methods is not guaranteed and no a priori error bound exists. However, the methods are numerically reliable and can be implemented iteratively; see, for example, [7, 71, 59, 60, 85, 95] for efficient implementations of the Krylov based methods. The main advantage of this approach is that it requires a low computational effort and small memory storage especially when compared to other reduction approaches. They reduce the computational cost to $\mathcal{O}(n^2r)$ (to $\mathcal{O}(nr)$ if the matrix A is sparse) and the storage requirements to $\mathcal{O}(nr)$. Moreover, Krylov methods are not based on minimization, as with the SVD-based model reduction. Instead they are based on moment matching, where one attempts to match some of the first coefficients of Taylor (or Neumann) series expansion of the original and reduced transfer functions. These methods find the

reduced order model in a relatively short time with a good numerical accuracy via a projection using bases of particular Krylov subspaces.

2.3.2.1 Moment matching techniques

Let $F(s) = C(sI_n - A)^{-1}B$ be the transfer function associated with the linear dynamical system (2.1). Then, if we expand $F(s)$ as the Taylor series expansion around a given finite point $\sigma \in \mathbb{R}$, we get

$$\begin{aligned} F(s) &= C(\sigma I_n - A - (\sigma - s)I_n)^{-1}B \\ &= C(I_n - (\sigma I_n - A)^{-1}(\sigma - s))^{-1}(\sigma I_n - A)^{-1}B \\ &= f_\sigma^{(0)} + f_\sigma^{(1)}(\sigma - s) + f_\sigma^{(2)}(\sigma - s)^2 + f_\sigma^{(3)}(\sigma - s)^3 + \dots \\ &= \sum_{j=0}^{\infty} f_\sigma^{(j)}(\sigma - s)^j. \end{aligned}$$

The coefficients $f_\sigma^{(j)}$ for $j \geq 0$ are called the j^{th} moments of the original system around σ , and they are given by

$$f_\sigma^{(j)} = C(\sigma I_n - A)^{-(j+1)}B.$$

It can be shown that these moments are the values of the transfer function and its derivatives evaluated at the point σ ; see [3]. Then, model-order reduction using a moment matching method consists in finding a lower order transfer function $F_r(s)$ having a power series expansion at σ as follows

$$F_r(s) = \hat{f}_\sigma^{(0)} + \hat{f}_\sigma^{(1)}(\sigma - s) + \hat{f}_\sigma^{(2)}(\sigma - s)^2 + \hat{f}_\sigma^{(3)}(\sigma - s)^3 + \dots$$

such that k moments are matched, i.e.,

$$f_\sigma^{(j)} = \hat{f}_\sigma^{(j)}, \quad j = 0, \dots, k - 1 \quad (2.29)$$

for an appropriate $k \ll n$. The reduced-order model resulting is known as a rational interpolation. Expanding $F(s)$ around $\sigma = \infty$, the Taylor series is given by

$$F(s) = \sum_{j=1}^{\infty} f_\infty^{(j)} s^{-j},$$

where $f_\infty^{(i)}$ are called the Markov parameters of $F(s)$ and defined by

$$f_\infty^{(j)} = CA^{j-1}B, \quad j \geq 1,$$

and the corresponding problem is known as a partial realization [69]. A power series expansion can also be performed about $\sigma = 0$, in this case the moments satisfy $f_0^{(j)} = -CA^{-(j+1)}B$ for $j \geq 0$, and the computed approximation is a Padé approximation [3, 140].

Importantly, these problems can be solved in a recursive and numerically reliable way, by using the Lanczos and Arnoldi procedures. In general for an arbitrary $\sigma \in \mathbb{R}$, the problem is a rational interpolation problem, see, for example, [4]. In this case rational Lanczos and rational Arnoldi methods give a numerically efficient solution. We can also use multiple interpolation points σ_i and this becomes the multi-point rational interpolation problem, see [36, 71, 146]. In this case, we use the rational Krylov method of Ruhe [120] to produce the reduced order models. The concept of these methods is to match the moments of the transfer function at selected frequencies and therefore we obtain a better approximation of the transfer function over a broad frequency range.

2.3.2.2 Approximation by moment matching

A straightforward approach to produce the reduced-order models can be obtained by explicitly computing $2m$ moments of the original system, where m is the order of the reduced model. Then, the frequency response of the reduced-order system is forced to correspond to the selected moments. This can be viewed as a selection of the coefficients for the numerator and denominator of the reduced-order transfer function through the solution of a linear system involving Hankel matrices. Unfortunately, numerical drawbacks of the explicit moment-matching can occur, such as ill-conditioned Hankel matrices, sensitivity of the partial realization, moment scaling, and the stability of the approximation [61].

Numerically reliable and efficient algorithms have been proposed in the literature for moment matching method without using an explicit moment computation; see [71]. When the matrix A is non-symmetric, the main Krylov subspace methods for non-symmetric problems rely on the Arnoldi algorithm and the non-symmetric Lanczos algorithm. As this thesis focus on the non-symmetric Lanczos procedure and the dynamical system in question is MIMO, we need to use the

non-symmetric block Lanczos and the non-symmetric global Lanczos algorithms to construct the projecting matrices. We recall in this paragraph the standard versions of these processes and we give some of them algebraic properties.

Now, we consider the dynamical system represented by the internal description

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

then one should to find two matrices V_m and W_m and construct the low order model

$$\Sigma_m = \left[\begin{array}{c|c} A_m & B_m \\ \hline C_m & 0 \end{array} \right],$$

by applying the oblique projector $P_m = V_m W_m^T$ to the original system Σ such that

$$A_m = W_m^T A V_m, \quad B_m = W_m^T B \quad \text{and} \quad C_m = C V_m.$$

A careful selection of V_m and W_m as the bases of certain Krylov subspaces results in moment matching. Then, these bases could be chosen using either the block Lanczos process [8, 68, 78] (in this case the projecting matrices will be noted \mathbb{V}_m and \mathbb{W}_m) or the global Lanczos process [86, 87](the projecting matrices will be noted \mathcal{V}_m and \mathcal{W}_m).

2.3.2.3 The non-symmetric block Lanczos-based method

The non-symmetric Lanczos algorithm was originally proposed by Lanczos in 1950 [95] as an oblique projection method for the computation of eigenvalues of non-symmetric matrices. The idea was to reduce the general matrix to a tridiagonal form from which the eigenvalues could be easily computed [9].

The first mathematical connection between the Lanczos algorithm and model reduction was shown in [69]. It was shown that partial realizations could be generated through the Lanczos process. Villemagne and Skelton [36] have shown that adaptations to the Krylov subspaces could be performed in order to generate Padé approximations. Applications of the moment matching results were utilized in the structural dynamics field as a model reduction technique for flexible

structures [135] and MIMO systems [92, 93].

Next, we first recall the definition of the block Krylov subspace and then we introduce the non-symmetric block Lanczos procedure.

Definition 2.3.2 *Given a matrix $A \in \mathbb{R}^{n \times n}$ and a block vectors $B \in \mathbb{R}^{n \times p}$, the i -th block Krylov subspace of A and B , denoted by $\mathbb{K}_i(A, B)$, is defined as*

$$\mathbb{K}_i(A, B) = \text{Range}(B, AB, \dots, A^{i-1}B). \quad (2.30)$$

Let V and W be two initial blocks of $\mathbb{R}^{n \times p}$, and consider the block Krylov subspaces $\mathbb{K}_m(A, V)$ and $\mathbb{K}_m(A^T, W)$. The non-symmetric block Lanczos algorithm applied to the pairs (A, V) and (A^T, W) generates two sequences of bi-orthonormal $n \times p$ matrices $\{V_i\}$ and $\{W_j\}$ such that

$$\mathbb{K}_m(A, V) = \text{Range}(V_1, V_2, \dots, V_m),$$

and

$$\mathbb{K}_m(A^T, W) = \text{Range}(W_1, W_2, \dots, W_m).$$

The matrices V_i and W_j that are generated by the block Lanczos algorithm satisfy the bi-orthogonality conditions, i.e.

$$\begin{cases} W_j^T V_i = 0_p, & \text{if } i \neq j, \\ W_j^T V_i = I_p, & \text{if } i = j. \end{cases} \quad (2.31)$$

Next, we give a stable version of the non-symmetric block Lanczos algorithm that was defined in [8]. The algorithm is summarized as follows.

Algorithm 1 The non-symmetric block Lanczos algorithm (BLA)

• **Inputs:** $A \in \mathbb{R}^{n \times n}$, $V, W \in \mathbb{R}^{n \times p}$ and m an integer.

1. **Compute the QR decomposition of $W^T V$, i.e.,** $W^T V = \delta \beta$;
 $V_1 = V \beta^{-1}$; $W_1 = W \delta$; $\tilde{V}_2 = A V_1$; $\tilde{W}_2 = A^T W_1$;
 2. **For $j = 1, \dots, m$**
 $\alpha_j = W_j^T \tilde{V}_{j+1}$; $\tilde{V}_{j+1} = \tilde{V}_{j+1} - V_j \alpha_j$; $\tilde{W}_{j+1} = \tilde{W}_{j+1} - W_j \alpha_j^T$;
Compute the QR decomposition of \tilde{V}_{j+1} and \tilde{W}_{j+1} , i.e.,
 $\tilde{V}_{j+1} = V_{j+1} \beta_{j+1}$; $\tilde{W}_{j+1} = W_{j+1} \delta_{j+1}^T$;
Compute the singular value decomposition of $W_{j+1}^T V_{j+1}$, i.e.,
 $W_{j+1}^T V_{j+1} = U_j \Sigma_j Z_j^T$;
 $\delta_{j+1} = \delta_{j+1} U_j \Sigma_j^{1/2}$; $\beta_{j+1} = \Sigma_j^{1/2} Z_j^T \beta_{j+1}$;
 $V_{j+1} = V_{j+1} Z_j \Sigma_j^{-1/2}$; $W_{j+1} = W_{j+1} U_j \Sigma_j^{-1/2}$;
 $\tilde{V}_{j+2} = A V_{j+1} - V_j \delta_{j+1}$; $\tilde{W}_{j+2} = A^T W_{j+1} - W_j \beta_{j+1}^T$;
 3. **end For.**
-

Setting $\mathbb{V}_m = [V_1, V_2, \dots, V_m]$ and $\mathbb{W}_m = [W_1, W_2, \dots, W_m]$, we have the following block Lanczos relations

$$A \mathbb{V}_m = \mathbb{V}_m \mathbb{T}_m + V_{m+1} \beta_{m+1} E_m^T,$$

and

$$A^T \mathbb{W}_m = \mathbb{W}_m \mathbb{T}_m^T + W_{m+1} \delta_{m+1}^T E_m^T,$$

where E_m is last $mp \times p$ block of the identity matrix I_{mp} and \mathbb{T}_m is the $mp \times mp$ block tridiagonal matrix defined by

$$\mathbb{T}_m = \begin{pmatrix} \alpha_1 & \delta_2 & & & \\ \beta_2 & \alpha_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \delta_m \\ & & & & \beta_m & \alpha_m \end{pmatrix}.$$

Let $\mathbb{V}_m, \mathbb{W}_m \in \mathbb{R}^{n \times mp}$ be the bi-orthonormal matrices computed by Algorithm 1, the application of the oblique projector $\Pi_m = \mathbb{V}_m \mathbb{W}_m^T$ on the original system (2.1) yields a reduced order system such that

$$A_m = \mathbb{W}_m^T A \mathbb{V}_m, \quad B_m = \mathbb{W}_m^T B \quad \text{and} \quad C_m = C \mathbb{V}_m. \quad (2.32)$$

2.3.2.4 The global Lanczos-based method

The global Krylov method was first proposed in [86, 87] for solving linear equations with multiple right hand sides and Lyapunov equations. Application to model-order reduction of first order systems is also studied in [29, 30, 31, 32]. It was also used for solving large Lyapunov matrix equations [88]. Basically, the global Krylov method is similar to the standard Krylov method except that the standard inner product is replaced by the Frobenius inner product defined above.

Next, we review some notations and definitions that will be used for the global Lanczos method. For two matrices X and Y in $\mathbb{R}^{n \times p}$, we define the Frobenius inner product $\langle X, Y \rangle_F = \text{Tr}(X^T Y)$ where $\text{Tr}(X^T Y)$ denotes the trace of the square matrix $X^T Y$. The associated Frobenius norm is given by $\|Y\|_F = \text{Tr}(Y^T Y)^{\frac{1}{2}}$. A system $\{V_1, V_2, \dots, V_m\}$ of elements of $\mathbb{R}^{n \times p}$ is said to be F -orthonormal if it is orthonormal with respect to the inner product $\langle \cdot, \cdot \rangle_F$, i.e., $\langle V_i, V_j \rangle_F = \delta_{i,j}$. For $Y \in \mathbb{R}^{n \times p}$, we denote by $\text{vec}(Y)$ the vector of \mathbb{R}^{np} obtained by stacking the columns of Y . For two matrices A and B , $A \otimes B = [a_{i,j} B]$ denotes the Kronecker product of the matrices A and B .

In the sequel, we give some properties of the Kronecker product.

1. $(A \otimes B)^T = A^T \otimes B^T$.
2. $(A \otimes B)(C \otimes D) = (AC \otimes BD)$.
3. If A and B are non-singular matrices of size $n \times n$ and $p \times p$ respectively, then the $np \times np$ matrix $A \otimes B$ is non-singular and $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$.
4. $\text{vec}(A)^T \text{vec}(B) = \text{Tr}(A^T B)$.

Definition 2.3.3 Let $A = [A_1, \dots, A_s]$ and $B = [B_1, \dots, B_l]$ be matrices of dimension $n \times sp$ and $n \times lp$, respectively, where A_i and B_j ($i = 1, \dots, s$; $j = 1, \dots, l$) are $n \times p$. Then the $s \times l$ matrix $A^T \diamond B$ is defined by:

$$A^T \diamond B = [\langle A_i, B_j \rangle_F]_{1 \leq i \leq s; 1 \leq j \leq l}.$$

Remark 2.3.1 The following relations were established in [21].

1. The matrix $A = [A_1, \dots, A_s]$ is F -orthonormal if and only if $A^\top \diamond A = I_s$.
2. For all $X \in \mathbb{R}^{n \times p}$, we have $X^\top \diamond X = \|X\|_F^2$.
3. $(DA)^\top \diamond B = A^\top \diamond (D^\top B)$.
4. $A^\top \diamond (B(L \otimes I_p)) = (A^\top \diamond B)L$.
5. $\|A^\top \diamond B\|_F \leq \|A\|_F \|B\|_F$.

Let $V \in \mathbb{R}^{n \times p}$, then the matrix Krylov subspace

$$\mathcal{K}_m(A, V) = \text{Span}\{V, AV, \dots, A^{m-1}V\}$$

is spanned by the matrices $V, AV, \dots, A^{m-1}V$. Hence $Z \in \mathcal{K}_m(A, V)$ means that

$$Z = \sum_{i=0}^{m-1} \alpha_i A^i V, \quad \alpha_i \in \mathbb{R}, \quad i = 0, \dots, m-1.$$

We recall that the previous subspace is different from the block Krylov subspace $\mathbb{K}_m(A, V)$ where $Z \in \mathbb{K}_m(A, V)$ means that

$$Z = \sum_{i=0}^{m-1} A^i V \Omega_i, \quad \Omega_i \in \mathbb{R}^{p \times p}, \quad i = 0, \dots, m-1.$$

let $W \in \mathbb{R}^{n \times p}$, the global Lanczos algorithm constructs two F -biorthogonal bases

$$\mathcal{V}_m = \{V_1, V_2, \dots, V_m\} \quad \text{and} \quad \mathcal{W}_m = \{W_1, W_2, \dots, W_m\}$$

of the matrix Krylov subspaces $\mathcal{K}_m(A, V)$ and $\mathcal{K}_m(A^T, W)$, respectively. This algorithm is summarized as follows.

Algorithm 2 The Global Lanczos Algorithm (GLA)

1. **Inputs:** $A \in \mathbb{R}^{n \times n}$, $V, W \in \mathbb{R}^{n \times p}$ and an integer m .
 2. Choose two $n \times p$ matrices V_1 and W_1 such that $\langle V_1, W_1 \rangle_F = 1$,
 3. Set $\beta_1 = \delta_1 = 0$ and $W_0 = V_0 = 0$,
 4. for $j=1, \dots, m$

$$\alpha_j = \langle W_j, AV_j \rangle_F,$$

$$\widetilde{V}_{j+1} = AV_j - \alpha_j V_j - \beta_j V_{j-1},$$

$$\widetilde{W}_{j+1} = A^T W_j - \alpha_j W_j - \delta_j W_{j-1},$$

$$\delta_{j+1} = |\text{Tr}(\widetilde{V}_{j+1}^T \widetilde{W}_{j+1})|^{1/2},$$

$$\beta_{j+1} = \text{Tr}(\widetilde{V}_{j+1}^T \widetilde{W}_{j+1}) / \delta_{j+1},$$

$$V_{j+1} = \widetilde{V}_{j+1} / \delta_{j+1},$$

$$W_{j+1} = \widetilde{W}_{j+1} / \beta_{j+1},$$
 5. Endfor
-

Let \mathcal{T}_m be the tridiagonal matrix of dimension $m \times m$ defined as

$$\mathcal{T}_m = \begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \delta_2 & \alpha_2 & & & \\ & \cdot & \cdot & \cdot & \\ & & \cdot & \cdot & \beta_m \\ & & & \delta_m & \alpha_m \end{pmatrix},$$

where α_i, β_i and δ_i are the scalars defined in the Algorithm 2. Define the matrix

$$\widetilde{\mathcal{T}}_m = \begin{pmatrix} \mathcal{T}_m \\ \delta_{m+1} e_m^T \end{pmatrix}.$$

The following result holds for the standard global Lanczos algorithm (see [87]).

Theorem 2.3.2 *Assume that the global Lanczos algorithm does not break down before m steps. Let $\mathcal{V}_{m+1} = [V_1, \dots, V_m, V_{m+1}]$ and $\mathcal{W}_{m+1} = [W_1, \dots, W_m, W_{m+1}]$ be the F -biorthogonal matrices of $\mathbb{R}^{n \times (m+1)p}$ constructed by Algorithm 2, then we*

have the following relations

$$\begin{aligned} A\mathcal{V}_m &= \mathcal{V}_m(T_m \otimes I_p) + \delta_{m+1}V_{m+1}E_m^T, \\ A\mathcal{V}_m &= \mathcal{V}_{m+1}(\tilde{T}_m \otimes I_p), \end{aligned} \tag{2.33}$$

$$\tag{2.34}$$

where $e_m = (0, \dots, 0, 1)^T \in \mathbb{R}^m$ and $E_m = (e_m \otimes I_p) = [0_p, \dots, 0_p, I_p]$.

2.4 Contributions of the Dissertation

The motivation of this thesis is now clear: There are many applications of linear systems in engineering, many of the real world applications are large scale. In the literature, many works are proposed for model order reduction problem based on standard Krylov subspace. Unfortunately, the standard versions of the Krylov subspace algorithms tend to create reduced order models that poorly approximated low frequency dynamics. To overcome this problem, some rational Krylov methods were recently defined; see [71, 120] and the references therein, and it was shown that these rational-based methods are more effective for model order reduction [1, 13, 14, 42, 50, 70], and also for solving large Lyapunov and Riccati matrix equations [40, 133].

Hence, we will focus in this thesis on Multi-input Multi-output (MIMO) continuous Linear Time Invariant (LTI) large scale dynamical systems, and we develop new efficient rational Krylov algorithms to produce reduced order systems.

first, and towards the goal of Krylov based model reduction, we propose a rational block Lanczos-type algorithm to compute two bi-orthogonal bases of the rational Krylov subspaces. After that, we show how to obtain some rational equations that describe the relation between these bases and the matrix A of the original model. Once the reduced order model is constructed, we must compute the exact transfer matrix error between the original and the reduced systems to measure the accuracy of the resulting reduced-order model. Grimme in his thesis [71] proposed the computation of the exact error in term of two residual vectors in the case of SISO systems, this result will be extended her to the MIMO case.

Moreover, we use the rational equations proposed to derive another error expression. One of the drawbacks of the rational Krylov methods is the selection of some interpolation points which must be appropriately chosen for a good convergence of the process. Some techniques to choose these parameters will be also proposed.

Next, we will propose a modified non-symmetric rational block Lanczos algorithm. This process can be considered as a generalization of the rational block Lanczos algorithm given in the last chapter where different multiplicities are considered for each interpolation point. The advantage of the modified rational block Lanczos algorithm is that the standard Lanczos equations remain valid also in the rational block case. After that, we will use these simple equations to derive simple residual error expressions. As rational Krylov methods are always related to the interpolation points, we will propose an adaptive approach for selecting some good shifts.

Another extension of the standard Krylov subspace method for MIMO systems is the global Krylov subspace. The global Lanczos process is an algorithm for computing F -biorthogonal bases of the rational matrix Krylov subspaces. We start by describing the general form of the rational global Lanczos algorithm and then we establish the rational equations which describe this process. After that, we modify the rational global Lanczos algorithm in such a way the standard Lanczos equations remain valid also in the rational global case, the resulting algorithm will be named the modified rational global Lanczos process. Next, we propose some adaptive techniques for choosing the interpolation points and then we combine one of these methods and the modified rational global Lanczos process to get an Adaptive Modified Rational Global Lanczos (AMRGL) procedure for reducing the dimension of large scale linear dynamical systems. In the second part of this Chapter, we consider the second-order dynamical systems and we applied the AMRGL algorithm to an equivalent state space model.

Always in the context of rational Krylov methods for model order reduction, we will propose also an extended block Lanczos algorithm for constructing bi-orthogonal bases of the extended Krylov subspace. This Krylov based method was first proposed by Druskin and Knizhnerman in [41] for numerically approx-

imating the action of a matrix function to a given vector. The advantage of this approach is that we are not even mentioning the numerical difficulties that might arise for selecting the poles of the rational Krylov subspace and then we don't need to construct the set of interpolation points as for the other rational Krylov algorithms. After describing the extended block Lanczos algorithm, we obtain some algebraic properties and then we apply this method to model order reduction. We show how to apply the extended block Lanczos process to MIMO dynamical systems to produce a low-order dimensional systems.

2.5 Outline of the Dissertation

We conclude this introduction with a summary of each of the remaining chapters.

In Chapter 3, we start by proposing a rational block Lanczos-type algorithm and then we show how to obtain a set of equations that describe the relations between the matrix A and the bases constructed by this procedure. After that, we use these equations to obtain an error expression between the original and the reduced-order transfer functions. Moreover, we propose adaptive techniques for selecting some interpolation points that used to construct the rational Krylov subspaces. The application of this algorithm to approximate the exponential on a block vector B will be also considered in this chapter. The last section is devoted to some numerical experiments to show the effectiveness of the proposed methods.

In Chapter 4, a modified version of the algorithm introduced in Chapter 2 is proposed, namely, the modified rational block Lanczos algorithm. The advantage of this process is that the standard Lanczos equations remain valid also in the rational block case. Moreover, these simple equations will be used to develop simple residual error expressions. Next, an adaptive method for choosing the interpolation points is propose and finally, some numerical examples will be given.

Chapter 5 presents another extension of the standard Krylov subspace method for MIMO systems, which is the global Krylov subspace. We first describe the adaptive modified rational global Lanczos (**AMRGL**) algorithm proposed and then

we establish Lanczos-like equations for the global case. Next, some adaptive techniques for choosing the interpolation points will be proposed. Second-order dynamical systems are also considered in this Chapter and the AMRGL algorithm is applied to an equivalent state space model. In the last section, some numerical examples will be introduced.

In Chapter 6, we propose an extended block Lanczos method and we obtain new algebraic properties for this process. The application of this method to model order reduction is also considered. We show how to apply the extended block Lanczos process to MIMO dynamical systems to produce a low-order dimensional systems. The last section is devoted to some numerical experiments for large and sparse problems to show the efficiency of the proposed approach.

In the last chapter (Chapter 7), a summary of the obtained results is provided, together with ideas to be explored in future work.

3

An adaptive rational block Lanczos-type algorithm for model reduction of large scale dynamical systems

As we explained in the next chapter, they are two well-known approaches for model reduction problem, SVD-based methods and Krylov-based model reduction. However, the SVD-based methods are not suitable for large-scale systems due to the use of dense matrix factorisation of $\mathcal{O}(n^3)$ and storage of $\mathcal{O}(n^2)$. As an alternative, Krylov subspace techniques become a good choices for large-scale systems because they rely on matrix-vector multiplication and they can be implemented iteratively in a numerically efficient manner [3].

Krylov based model reduction are based on matching the moments of the orig-

inal transfer function around some selected frequencies to finding a reduced order model that matches a certain number of moments of the original model around these frequencies. This is achieved by iteratively constructing matrices that span certain Krylov subspaces of A and B , and/or A^T and C . In other words, we can use the Lanczos and the Arnoldi processes to solve the problem in a recursive and numerically efficient way. In particular, the Lanczos process has been used for the SISO and MIMO dynamical systems; see [51, 50, 78, 95] and the references therein. The Padé via Lanczos method of [48], which exploits the deep connection between the Lanczos procedure and the moment matching problem at σ , is one of the leading efforts for this case. The Lanczos procedures can be used to match the moments of Σ only at a single interpolation point. However, the standard version of the Lanczos algorithm builds reduced order models that poorly approximate some frequency dynamics and to overcome this problem, one is interested in matching the moments at various interpolation points to obtain a better approximation over a broad frequency range. In this case, the problem is called the multipoint rational interpolation problem and the reduced order model can be constructed by using the rational Krylov method first proposed by Ruhe [120] and developed these last years in [58, 59, 71, 70, 140]. Since ones matched the moments of the transfer function at various frequencies, a better approximation of the transfer function over a broad frequency range is obtained. The multipoint rational interpolation was first proposed by Skelton and al. [36, 145, 146]. Grimme in his thesis showed how one can construct the required projection by Krylov methods in a numerically efficient way; see [71]. One of the main problems of the rational Krylov methods is the selection of suitable shifts to guarantee a good convergence of the process. Therefore, various methods have been proposed in the literature to construct these interpolation points. In [20, 58] Gugercin et al. proposed an Iterative Rational Krylov Algorithm (IRKA) to compute a reduced order model satisfying the first order conditions for the \mathcal{H}_2 approximation. Other adaptive methods (for the SISO case) are introduced in [22, 41, 51, 66, 70, 78] and the references therein.

In the first section of this chapter, we recall the multipoint rational interpolation problem and we give some basic results. In the second section, we propose a

rational block Lanczos-type process and we give some algebraic properties related to the this algorithm. An error expression between the original and the reduced-order transfer functions is derived in Section 3. In Section 4, we propose adaptive techniques for selecting some interpolation points. The last section is devoted to some numerical experiments.

3.1 Multipoint rational interpolation

For the multipoint rational interpolation problem by projection, the goal is to obtain the reduced system by projection. In the other words, we try to find matrices $\mathbb{V}_m \in \mathbb{R}^{n \times r}$ and $\mathbb{W}_m \in \mathbb{R}^{n \times r}$ with $\mathbb{W}_m^T \mathbb{V}_m = I_r$ and such that the reduced order model

$$\Sigma_m = \left[\begin{array}{c|c} A_m & B_m \\ \hline C_m & 0 \end{array} \right]$$

matches the moments of the original system Σ at the selected interpolation points. Consider the following projection:

$$\begin{cases} x = \mathbb{V}_m x_m, \\ \mathbb{V}_m \in \mathbb{R}^{n \times r}, x \in \mathbb{R}^n, x_m \in \mathbb{R}^r, \end{cases}$$

where $r \ll n$. By applying this projection to the original system and then multiplying the state equation by transpose of the matrix \mathbb{W}_m , a reduced model can be obtained as follows,

$$\begin{cases} \dot{x}_m(t) = \mathbb{W}_m^T A \mathbb{V}_m x_m(t) + \mathbb{W}_m^T B u(t) \\ y_m(t) = C \mathbb{V}_m x_m(t), \end{cases}$$

Then, the reduced order system in state space is identified by the following matrices:

$$A_m = \mathbb{W}_m^T A \mathbb{V}_m, B_m = \mathbb{W}_m^T B, C_m = C \mathbb{V}_m. \quad (3.1)$$

Now, the question is how to choose the projection matrices to find a reduced system that matches the original system at selected frequencies. The following theorems show how one can achieve this goal by Krylov projection methods. We start by the case of matching the Markov parameters. A special case, called Oblique Projection, for matching only the Markov parameters has been introduced in [84]. In this case, matching the Markov parameters leads to a good

approximation at high frequencies which most of the time is not desired. In the following theorem, a general case is discussed.

Theorem 3.1.1 *The matrices A_m, B_m and C_m generated by applying the block Lanczos process to the original system Σ are such that the first $2m - 1$ Markov parameters of the original and the reduced models are the same, that is,*

$$CA^j B = C_m A_m^j B_m, \quad \text{for } j = 0, 1, \dots, 2(m - 1).$$

Proof. The complete proof is found in [78].

The following result is presented in [71] for SISO systems, and is extended to the MIMO case in [57]. It shows how to construct the bi-orthogonal bases \mathbb{V}_m and \mathbb{W}_m so that the multipoint rational interpolation problem is solved, i.e., the reduced order model has to interpolate the original transfer function $F(s)$ and its first derivative at the interpolation points $\{\sigma_i\}_{i=1}^m$.

Theorem 3.1.2 *Given $F(s) = C(sI_n - A)^{-1}B$ and m interpolation points $\{\sigma_i\}_{i=1}^m$ that verify $\sigma_i \neq \sigma_j$ for $i \neq j$. Let $\mathbb{V}_m \in \mathbb{R}^{n \times r}$ and $\mathbb{W}_m \in \mathbb{R}^{n \times r}$ be obtained as follows:*

$$\text{Range}(\mathbb{V}_m) = \text{Range}\{(A - \sigma_1 I_n)^{-1}B, \dots, (A - \sigma_m I_n)^{-1}B\} \quad (3.2)$$

$$\text{Range}(\mathbb{W}_m) = \text{Range}\{(A - \sigma_1 I_n)^{-T}C^T, \dots, (A - \sigma_m I_n)^{-T}C^T\} \quad (3.3)$$

with $\mathbb{W}_m^T \mathbb{V}_m = I_r$. Then, the reduced order transfer function $F_m(s) = C_m(sI_m - A_m)^{-1}B_m$ obtained in (3.1) interpolates $F(s)$ and its first derivative at $\{\sigma_i\}_{i=1}^m$.

Theorem 3.1.2 states that for moment matching problem, one has to construct two full-rank matrices \mathbb{V}_m and \mathbb{W}_m such that $\text{Range}(\mathbb{V}_m)$ and $\text{Range}(\mathbb{W}_m)$ satisfying equations (3.2) and (3.3) respectively.

Several algorithms have been implemented in the literature: Lanczos algorithm and Arnoldi algorithm and its variants; see [71] and referenced therein. In general, these processes follow the Rational Krylov methods of Ruhe [121]. Based on the conditions (3.2) and (3.3) of theorem 3.1.2, a rational block Lanczos algorithm is proposed in the next section.

3.2 The rational block Lanczos method

The rational Krylov method was originally proposed by Ruhe [120] in the context of approximating interior eigenvalues, which with appropriately chosen shifts would accelerate convergence to the sought after spectral region. Within model order reduction, the role of rational Krylov subspaces is rather different, as they are particularly well suited for approximating the behavior of the transfer function on the imaginary axis. Indeed, it is now acknowledged as being one of the most powerful projection approaches for reducing the order of large scale linear dynamical system.

3.2.1 The rational block Lanczos algorithm

As discussed in the next section, in order to solve the multipoint rational interpolation problem by Krylov techniques, one has to construct full rank matrices \mathbb{V}_m and \mathbb{W}_m which span the required Krylov subspaces for some selected interpolation points $\sigma_1, \sigma_2, \dots, \sigma_m$. That's will be done by using the rational block Lanzos procedure.

The rational block Lanczos process presented in Algorithm 3 is a procedure for constructing bi-orthonormal bases of the union of the block Krylov subspaces defined as follows

$$\begin{aligned} \mathbb{K}_m(A, B, \Sigma_m) &= \text{Range}\{(A - \sigma_1 I_n)^{-1} B, \dots, \prod_{k \leq m} (A - \sigma_k I_n)^{-1} B\}, \quad (3.4) \\ \mathbb{K}_m(A^T, C^T, \Sigma_m) &= \text{Range}\{(A - \sigma_1 I_n)^{-T} C^T, \dots, \prod_{k \leq m} (A - \sigma_k I_n)^{-T} C^T\}, \end{aligned} \quad (3.5)$$

where $\Sigma_m = \{\sigma_1, \dots, \sigma_m\}$ is the set of interpolation points. The rational block Lanczos-type algorithm is defined as follows.

Algorithm 3 The rational block Lanczos-type algorithm(RBLA)

1. **Input:** $A \in \mathbb{R}^{n \times n}$, $B, C^T \in \mathbb{R}^{n \times p}$.
 2. **Output:** two bi-orthogonal matrices \mathbb{V}_{m+1} and \mathbb{W}_{m+1} of $\mathbb{R}^{n \times (m+1)p}$.
function $[\mathbb{V}_m, \mathbb{W}_m] = \mathbf{Rational-Block-Lanczos}(A, B, C, \{\sigma_1, \dots, \sigma_m\})$
 3. **Set** $S_0 = (A - \sigma_1 I_n)^{-1} B$ and $R_0 = (A - \sigma_1 I_n)^{-T} C^T$
 4. **Set** $S_0 = V_1 H_{1,0}$ and $R_0 = W_1 G_{1,0}$ such that $W_1^T V_1 = I_p$;
 5. **Initialize:** $\mathbb{V}_1 = [V_1]$ and $\mathbb{W}_1 = [W_1]$.
 6. **For** $k = 1, \dots, m$
 7. **if** ($k < m$)
 8. **if** $\{\sigma_{k+1} = \infty\}$; $S_k = AV_k$ and $R_k = A^T W_k$; **else**
 9. $S_k = (A - \sigma_{k+1} I_n)^{-1} V_k$ and $R_k = (A - \sigma_{k+1} I_n)^{-T} W_k$; **endif**
 10. $H_k = \mathbb{W}_k^T S_k$ and $G_k = \mathbb{V}_k^T R_k$;
 11. $S_k = S_k - \mathbb{V}_k H_k$ and $R_k = R_k - \mathbb{W}_k G_k$;
 12. $S_k = V_{k+1} H_{k+1,k}$ and $R_k = W_{k+1} G_{k+1,k}$; (QR factorization)
 13. $W_{k+1}^T V_{k+1} = P_k D_k Q_k^T$; (Singular Value Decomposition)
 14. $V_{k+1} = V_{k+1} Q_k D_k^{-1/2}$ and $W_{k+1} = W_{k+1} P_k D_k^{-1/2}$;
 15. $H_{k+1,k} = D_k^{1/2} Q_k^T H_{k+1,k}$ and $G_{k+1,k} = D_k^{1/2} P_k^T G_{k+1,k}$;
 16. $\mathbb{V}_{k+1} = [\mathbb{V}_k, V_{k+1}]$; $\mathbb{W}_{k+1} = [\mathbb{W}_k, W_{k+1}]$;
 17. **else**
 18. **if** $\{\sigma_{m+1} = \infty\}$; $S_m = AB$ and $R_m = A^T C$; **else**
 19. $S_m = A^{-1} B$ and $R_m = A^{-T} C^T$; **endif**
 20. $H_m = \mathbb{W}_m^T S_m$ and $G_m = \mathbb{V}_m^T R_m$;
 21. $S_m = S_m - \mathbb{V}_m H_m$ and $R_m = R_m - \mathbb{W}_m G_m$;
 22. $S_m = V_{m+1} H_{m+1,m}$ and $R_m = W_{m+1} G_{m+1,m}$; (QR factorization)
 23. $W_{m+1}^T V_{m+1} = P_m D_m Q_m^T$; (Singular Value Decomposition)
 24. $V_{m+1} = V_{m+1} Q_m D_m^{-1/2}$ and $W_{m+1} = W_{m+1} P_m D_m^{-1/2}$;
 25. $H_{m+1,m} = D_m^{1/2} Q_m^T H_{m+1,m}$ and $G_{m+1,m} = D_m^{1/2} P_m^T G_{m+1,m}$;
 26. $\mathbb{V}_{m+1} = [\mathbb{V}_m, V_{m+1}]$; $\mathbb{W}_{m+1} = [\mathbb{W}_m, W_{m+1}]$;
 27. **endif**
 28. **endFor**.
-

We notice that in our setting, we assume that we are not given the sequence of shifts $\sigma_1, \sigma_2, \dots, \sigma_{m+1}$ and then we need to include the procedure to automatically generate this sequence during the iterations of the process. This adaptive procedure will be defined in the next sections.

In the rational block Lanczos-type algorithm (Algorithm 3), steps 8-9 and steps 18-19 are used to generate the next Lanczos vectors. According to Algorithm 3, two residual expressions are used. At each iteration k , we used a new interpolation

point σ_{k+1} , $k = 1, \dots, m - 1$ and we initialize the subsequent Krylov subspaces corresponding to this shift by $S_k = (A - \sigma_{k+1}I_n)^{-1}V_k$ and $R_k = (A - \sigma_{k+1}I_n)^{-T}W_k$ if σ_{k+1} is finite and $S_k = AV_k$, $R_k = A^TW_k$ if $\sigma_{k+1} = \infty$.

To insure that the block vectors V_{k+1} and W_{k+1} generated in each iteration are bi-orthogonal, the QR and SVD decompositions are used (steps 12-14 and 22-24). The matrices H_k and G_k constructed in steps 10 and 20 are $kp \times p$ and they are used to construct the block upper Hessenberg matrices \mathbb{H}_m and \mathbb{G}_m , respectively (for more details see Theorem 3.2.1).

We notice that a breakdown may occur in Algorithm 3 if the smallest singular value of $W_{k+1}^TV_{k+1}$ is zero, which causes a problem in the calculating of the block V_{k+1} and W_{k+1} . In [8], a novel breakdown treatment scheme was proposed to overcome this problem for the single point block Lanczos algorithm ABLE. This method is generalized in [111] for MABLE algorithm. Here, the same technique could be used to detect and cure breakdowns. However, this problem of breakdown or near-breakdown is not developed in this thesis.

3.2.2 Analysis of the rational block Lanczos algorithm

In this subsection, we give some theoretical results which establish the rational Lanczos equations that relate the matrix A of the original system, the bases $\mathbb{V}_m, \mathbb{W}_m$ constructed by Algorithm 3 and the Hessenberg matrices generated also by this algorithm.

Theorem 3.2.1 *Let \mathbb{V}_{m+1} and \mathbb{W}_{m+1} be the matrices generated by Algorithm 3, assuming that A is non-singular and that all the interpolation points σ_i , $i = 1, \dots, m$ are finite real numbers. Then, there exist $(m + 1)p \times mp$ block upper Hessenberg matrices $\tilde{\mathbb{H}}_m, \tilde{\mathbb{G}}_m, \tilde{\mathbb{K}}_m$ and $\tilde{\mathbb{L}}_m$ such that the following relations hold for*

the left and the right Krylov subspaces:

$$A\mathbb{V}_{m+1}\tilde{\mathbb{H}}_m = \mathbb{V}_{m+1}\tilde{\mathbb{K}}_m \quad (3.6)$$

$$A^T\mathbb{W}_{m+1}\tilde{\mathbb{G}}_m = \mathbb{W}_{m+1}\tilde{\mathbb{L}}_m, \quad (3.7)$$

$$\mathbb{H}_m = \mathbb{W}_m^T A^{-1} \mathbb{V}_m \mathbb{K}_m, \quad (3.8)$$

$$\mathbb{G}_m = \mathbb{V}_m^T A^{-T} \mathbb{W}_m \mathbb{L}_m, \quad (3.9)$$

where \mathbb{H}_m , \mathbb{G}_m , \mathbb{K}_m and \mathbb{L}_m are the $mp \times mp$ block upper Hessenberg matrices obtained by deleting the last row block vectors of $\tilde{\mathbb{H}}_m$, $\tilde{\mathbb{G}}_m$, $\tilde{\mathbb{K}}_m$ and $\tilde{\mathbb{L}}_m$, respectively.

Proof. We begin by the case where $k = 1, \dots, m - 1$ which involves the execution of Step 9. Replacing the expression of S_k into the expressions of Step 11 and Step 12 yields the following relation

$$V_{k+1}H_{k+1,k} = (A - \sigma_{k+1}I_n)^{-1}V_k - \mathbb{V}_k H_k$$

which can be written as

$$[\mathbb{V}_k \ V_{k+1}] \begin{bmatrix} H_k \\ H_{k+1,k} \end{bmatrix} = (A - \sigma_{k+1}I_n)^{-1}V_k. \quad (3.10)$$

Multiplying (3.10) on the left by $(A - \sigma_{k+1}I_n)$ and replacing V_k by $\mathbb{V}_k E_k$ gives

$$(A - \sigma_{k+1}I_n)\mathbb{V}_{k+1} \begin{bmatrix} H_k \\ H_{k+1,k} \end{bmatrix} = \mathbb{V}_k E_k,$$

where E_k is an $kp \times p$ tall thin matrix with an identity matrix of dimension p at the k^{th} block and zero elsewhere. Re-arranging the expression of the last equation as

$$A\mathbb{V}_{k+1} \begin{bmatrix} H_k \\ H_{k+1,k} \end{bmatrix} = \mathbb{V}_{k+1} \left(\begin{bmatrix} E_k \\ 0 \end{bmatrix} + \sigma_{k+1} \begin{bmatrix} H_k \\ H_{k+1,k} \end{bmatrix} \right), \quad k = 1, \dots, m - 1. \quad (3.11)$$

On the other hand, for $k = 1, \dots, m - 1$, we have

$$A\mathbb{V}_{m+1} = [A\mathbb{V}_{k+1}, A\mathbb{V}_{k+2}, \dots, A\mathbb{V}_m, A\mathbb{V}_{m+1}].$$

Therefore, we can deduce from 3.11, the following expression

$$A\mathbb{V}_{m+1} \begin{bmatrix} H_k \\ H_{k+1,k} \\ \mathbf{0} \end{bmatrix} = \mathbb{V}_{m+1} \left(\begin{bmatrix} E_k \\ 0 \\ \mathbf{0} \end{bmatrix} + \sigma_{k+1} \begin{bmatrix} H_k \\ H_{k+1,k} \\ \mathbf{0} \end{bmatrix} \right), \quad (3.12)$$

where $\mathbf{0}$ is the zero matrix having $m - k$ rows.

Now, consider the case where $k = m$. Using steps 19 – 21 gives the following

relation

$$V_{m+1}H_{m+1,m} = A^{-1}B - \mathbb{V}_m H_m \quad (3.13)$$

Since $(A - \sigma_1 I_n)^{-1}B = V_1 H_{1,0}$ and $V_1 = \mathbb{V}_m E_1$, (3.13) can be rewritten as

$$\mathbb{V}_{m+1} \begin{bmatrix} H_m \\ H_{m+1,m} \end{bmatrix} = A^{-1}(A - \sigma_1 I_n) \mathbb{V}_m E_1 H_{1,0}.$$

Multiplying on the left by A and rearranging the expression results in

$$A \mathbb{V}_{m+1} \left(\begin{bmatrix} H_m \\ H_{m+1,m} \end{bmatrix} - \begin{bmatrix} E_1 \\ 0 \end{bmatrix} H_{1,0} \right) = \mathbb{V}_{m+1} \left(-\sigma_1 \begin{bmatrix} E_1 \\ 0 \end{bmatrix} H_{1,0} \right) \quad (3.14)$$

Equations (3.12) and (3.14) lead to the following expression

$$A \mathbb{V}_{m+1} \tilde{\mathbb{H}}_m = \mathbb{V}_{m+1} \tilde{\mathbb{K}}_m, \quad (3.15)$$

where $\tilde{\mathbb{H}}_m$ and $\tilde{\mathbb{K}}_m$ are the block upper Hessenberg matrices of $\mathbb{R}^{(m+1)p \times mp}$, given as follows

$$\tilde{\mathbb{H}}_m = [\tilde{\mathbb{H}}^{(1)}, \tilde{\mathbb{H}}^{(2)}, \dots, \tilde{\mathbb{H}}^{(m)}] \quad \text{and} \quad \tilde{\mathbb{K}}_m = [\tilde{\mathbb{K}}^{(1)}, \tilde{\mathbb{K}}^{(2)}, \dots, \tilde{\mathbb{K}}^{(m)}],$$

where for $k = 1, \dots, m-1$ the k -th block columns are given by

$$\tilde{\mathbb{H}}^{(k)} = \begin{bmatrix} H_k \\ H_{k+1,k} \\ \mathbf{0} \end{bmatrix} \quad \text{and} \quad \tilde{\mathbb{K}}^{(k)} = \begin{bmatrix} E_k + \sigma_{k+1} H_k \\ \sigma_{k+1} H_{k+1,k} \\ \mathbf{0} \end{bmatrix}$$

and for $k = m$ we have

$$\tilde{\mathbb{H}}^{(m)} = \begin{bmatrix} H_m - E_1 H_{1,0} \\ H_{m+1,m} \end{bmatrix} \quad \text{and} \quad \tilde{\mathbb{K}}^{(m)} = \begin{bmatrix} -\sigma_1 E_1 H_{1,0} \\ 0 \end{bmatrix}.$$

Equation (3.8) is easily derived from the relation (3.6).

In a similar way, we can show the relations (3.7) and (3.9) for the left Krylov subspace.

We notice that since $\tilde{\mathbb{K}}^{(m)} = \begin{bmatrix} \mathbb{K}^{(m)} \\ 0 \end{bmatrix}$, and $A \mathbb{V}_{m+1} = [A \mathbb{V}_m, A V_{m+1}]$ it follows that

$$A \mathbb{V}_{m+1} \tilde{\mathbb{H}}_m = \mathbb{V}_m \mathbb{K}_m. \quad (3.16)$$

In the same manner, we also have

$$A^T \mathbb{W}_{m+1} \tilde{\mathbb{G}}_m = \mathbb{W}_m \mathbb{L}_m. \quad (3.17)$$

In what follows in this chapter, we assume that all the shifts σ_i , $i = 1, \dots, m$ are finite real numbers. This was always the case in our numerical examples.

3.3 model reduction error

The computation of the exact transfer matrix error between the original and the reduced systems

$$\epsilon(s) = F(s) - F_m(s) \quad (3.18)$$

is important for the measure of the accuracy of the resulting reduced-order model. This error can be used to monitor the number of iteration required for convergence of the reduced order model. Moreover, it can be also used to know how the response of the reduced model is sufficiently close to that of the original system. Unfortunately, the exact error $\epsilon(s)$ is not available, because the higher dimension of the original system yields the computation of $F(s)$ very difficult. To remedy this situation, various approaches have been explored in the literature for estimating the error (3.18).

3.3.1 Residual error

In [71], Grimme proposed the computation of the exact error in term of two residual vectors in the case of Single-Input Single-Output systems. The result is extended here to the Multi-Input Multi-Output case. Let

$$\begin{cases} R_B(s) &= B - (sI_n - A)\mathbb{V}_m\tilde{X}_B(s), \\ R_C(s) &= C^T - (sI_n - A)^T\mathbb{W}_m\tilde{X}_C(s) \end{cases} \quad (3.19)$$

be the residual expressions, where $\tilde{X}_B(s)$ and $\tilde{X}_C(s)$ are the solutions of the matrix equations

$$\begin{cases} (sI_{mp} - A_m)\tilde{X}_B(s) &= B_m, \\ (sI_{mp} - A_m)^T\tilde{X}_C(s) &= C_m^T, \end{cases}$$

and satisfy the Petrov-Galerkin conditions

$$\begin{cases} R_B(s) \perp \text{Range}(W_1, \dots, W_m) \\ R_C(s) \perp \text{Range}(V_1, \dots, V_m), \end{cases}$$

which means that $\mathbb{W}_m^T R_B(s) = \mathbb{V}_m^T R_C(s) = 0$.

Residual expressions are a significant tool for quantifying the error in iterative linear systems solving, and simple residual expressions arise in the context of Arnoldi and Lanczos processes. The residuals are pertinent to the computation of

the exact error, and they were utilized in [84] for the partial realization problem. In the following result, we give an expression of the error $\epsilon(s)$ based on the residual expressions.

Theorem 3.3.1 *The error between the frequency responses of the original and reduced-order systems can be expressed as*

$$\epsilon(s) = R_C^T(s)(sI_n - A)^{-1}R_B(s). \quad (3.20)$$

The proof of this theorem is similar to the one of Theorem 5.1 given in [71] for SISO system.

3.3.2 An error expression for the transfer functions

In this paragraph, we compute an error estimation using the proposed rational block Lanczos-type algorithm (Algorithm 3) and the rational Lanczos equations derived in Theorem 3.2.1.

In the previous section we defined the rational block Krylov subspaces by (3.4) and (3.5). However, the inclusion of the block vectors B and C^T may be beneficial. Then for computing an error estimation, we use the following rational Krylov subspaces

$$\mathbb{K}_m(A, B, \Sigma'_m) = \text{Range}\{B, (A - \sigma_2 I_n)^{-1}B, \dots, \prod_{k=2}^m (A - \sigma_k I_n)^{-1}B\}, \quad (3.21)$$

and

$$\mathbb{K}_m(A^T, C^T, \Sigma'_m) = \text{Range}\{C^T, (A - \sigma_2 I_n)^{-T}C^T, \dots, \prod_{k=2}^m (A - \sigma_k I_n)^{-T}C^T\}, \quad (3.22)$$

where $\Sigma'_m = \{\sigma_2, \dots, \sigma_m\}$ is the set of interpolation points. Thus we have the following theorem.

Theorem 3.3.2 *Let \mathbb{V}_m and \mathbb{W}_m be the matrices computed using the rational*

block Lanczos algorithm. If $(sI_n - A)$ and $(sI_{mp} - A_m)$ are non-singular, we have $F(s) - F_m(s) = C(sI_n - A)^{-1}(\mathbb{V}_m \mathbb{W}_m^T - I_n)AV_{m+1}H_{m+1,m}E_m^T \mathbb{H}_m^{-1}(sI_{mp} - A_m)^{-1}B_m$.

$$(3.23)$$

Proof. The error between the initial and the projected transfer functions is given by

$$F(s) - F_m(s) = C(sI_n - A)^{-1}(B - (sI_n - A)\mathbb{V}_m(sI_{mp} - A_m)^{-1}B_m).$$

Since

$$A\mathbb{V}_{m+1}\tilde{\mathbb{H}}_m = \mathbb{V}_m\mathbb{K}_m, \quad (3.24)$$

then

$$A\mathbb{V}_m = (\mathbb{V}_m\mathbb{K}_m - AV_{m+1}H_{m+1,m}E_m^T)\mathbb{H}_m^{-1} \quad (3.25)$$

and

$$A_m = \mathbb{W}_m^T A\mathbb{V}_m = (\mathbb{K}_m - \mathbb{W}_m^T AV_{m+1}H_{m+1,m}E_m^T)\mathbb{H}_m^{-1}. \quad (3.26)$$

Using equations (3.25) and (3.26), we obtain

$$(sI_n - A)\mathbb{V}_m = \mathbb{V}_m(sI_{mp} - A_m) - \Gamma_m,$$

where

$$\Gamma_m = (\mathbb{V}_m \mathbb{W}_m^T - I_n)AV_{m+1}H_{m+1,m}E_m^T \mathbb{H}_m^{-1}.$$

The relation (3.23) can be obtained using this result and the fact that $\mathbb{V}_m \mathbb{W}_m^T B = B$.

3.3.3 Residual error expressions for the rational Lanczos algorithm

In [50] simple Lanczos equations for the standard rational case are proposed and used for deriving simple residual error-expressions. In this section, we use the rational Lanczos equations given in Theorem 3.2.1 to simplify the residual error expressions. To simplify calculations, we use the rational Krylov subspaces in (3.21) and (3.22). Using the rational Lanczos equations and the fact that $B \in \mathbb{K}_m(A, B, \Sigma'_m)$, $C^T \in \mathbb{K}_m(A^T, C^T, \Sigma'_m)$, the expressions of the residual $R_B(s)$ and

$R_C(s)$ could be written as

$$\begin{aligned} R_B(s) &= B - (sI_n - A)\mathbb{V}_m(sI_{mp} - A_m)^{-1}B_m \\ &= \underbrace{(\mathbb{V}_m\mathbb{W}_m^T - I)AV_{m+1}}_{\tilde{B}} \underbrace{H_{m+1,m}E_m^T\mathbb{H}_m^{-1}(sI_{mp} - A_m)^{-1}B_m}_{\tilde{R}_B(s)} \end{aligned}$$

and

$$\begin{aligned} R_C(s) &= C^T - (sI - A)^T\mathbb{W}_m(sI_{mp} - A_m)^{-T}C_m^T \\ &= \underbrace{(\mathbb{W}_m\mathbb{V}_m^T - I)A^TW_{m+1}}_{\tilde{C}^T} \underbrace{G_{m+1,m}E_m^T\mathbb{G}_m^{-1}(sI_{mp} - A_m)^{-T}C_m^T}_{\tilde{R}_C(s)}, \end{aligned}$$

where $\tilde{R}_B(s)$, $\tilde{R}_C(s)$ are the terms of the residual errors $R_B(s)$ and $R_C(s)$, respectively, depending on the frequencies. The matrices \tilde{B} , \tilde{C}^T are frequency-independent terms of $R_B(s)$ and $R_C(s)$, respectively. Therefore, the error expression in (3.20) becomes

$$\epsilon(s) = \tilde{R}_C(s)^T \tilde{C}(sI_n - A)^{-1} \tilde{B} \tilde{R}_B(s) = \tilde{R}_C(s)^T \tilde{F}(s) \tilde{R}_B(s).$$

The transfer function $\tilde{F}(s) = \tilde{C}(sI_n - A)^{-1}\tilde{B}$ contains terms related to the original system which makes the computation of $\|\tilde{R}_C^T \tilde{F} \tilde{R}_B\|_\infty$ very expensive. Then, instead of using $\tilde{F}(s)$ we can use an approximation of $\tilde{F}(s)$. Various possible approximations of the error $\epsilon(s)$ are listed in Table 3.1.

Table 3.1: Estimations of the error $\epsilon(s)$

1	$\hat{\epsilon}(s) = \tilde{R}_B(s)$
2	$\hat{\epsilon}(s) = \tilde{R}_C(s)^T$
3	$\hat{\epsilon}(s) = \tilde{F}_m(s)\tilde{R}_B(s)$
4	$\hat{\epsilon}(s) = \tilde{F}_m(s)$
5	$\hat{\epsilon}(s) = \tilde{R}_C^T(s)\tilde{F}_m(s)$
6	$\hat{\epsilon}(s) = \tilde{R}_C^T(s)\tilde{F}_m(s)\tilde{R}_B(s)$

3.4 An adaptive-order rational block Lanczos-type algorithm

3.4.1 Interpolation point selection

Model-order reduction using multipoint rational interpolation generally gives a more accurate reduced-order model than interpolation around a single point. Unfortunately, One of the drawbacks of this approach is the selection of the interpolation points [70]. The location of the interpolation points and the number of moments matched dictates the accuracy of the reduced-order model. In [18, 72] the Iterative Rational Krylov Algorithm (IRKA) has been proposed in the context of the \mathcal{H}_2 -optimal model-order reduction by using a specific way to choose the interpolation points σ_i , $i = 1, \dots, m$. Starting from an initial set of interpolation points, a reduced-order system is determined and a new set of interpolation points is chosen as the Ritz values $-\lambda_i(A_m)$, $i = 1, \dots, m$, where $\lambda_i(A_m)$ are the eigenvalues of A_m . The process continues until the Ritz values from consecutive reduced-order models stagnate. The main disadvantage of this method is that it requires the construction of many Krylov subspaces which will not be utilized in the final model and only the last subspace is used. In contrast to IRKA method, Grimme in his thesis [70] proposed an adaptive method for choosing the interpolation points. This approach is based on the residual expression derived for the rational Lanczos algorithm such that the interpolation points are selected where the residual error is large. At each iteration of the algorithm, a residual function is computed and a new interpolation point is selected so as to correspond to the maximum of this residual function. He studied also the placement and selection of interpolation points, moreover, connections are made between the locations of interpolations points and the convergence behavior of the model. In particular, Grimme is concentrated on the popular choices of purely or imaginary shifts.

In [20, 43, 70, 97, 134] some techniques for choosing good interpolation points have been proposed. The aim of these methods is the construction of the next interpolation point at every step and they are based on the idea that the shifts

should be selected such that the norm of certain approximation of the error should be minimized at every iteration.

3.4.2 An adaptive choice of interpolation points

In the following, we introduce a new adaptive approach for selecting the interpolation points. The proposed method is based on the following error-approximation expression

$$\hat{\epsilon}(s) = \tilde{R}_C^T(s)\tilde{R}_B(s).$$

Then, the procedure to generate the set of interpolation points Σ_m used in Algorithm 3 is described as follows. We start by given two initial shifts σ_1, σ_2 , and we construct the next shift $\sigma_{k+2} \in \mathbb{R}$ as

$$\tilde{\sigma}_{k+2} = \arg \max_{s \in S} \|\tilde{R}_C^T(s)\tilde{R}_B(s)\|_2, \quad k = 1, \dots, m - 2. \quad (3.27)$$

and if $\tilde{\sigma}_{k+2}$ is complex, its real part is retained and used as the next interpolation point.

Remark 3.4.1 *The choice of the approximated error expression $\hat{\epsilon}(s) = \tilde{R}_C^T(s)\tilde{R}_B(s)$ is a heuristic choice that allowed to have good shifts without much calculations as is shown in the numerical tests. We notice that for small problems, one can also use the following criterion for selecting the shifts*

$$\sigma_{k+2} = \arg \max_{s \in S} \|R_C^T(s)R_B(s)\|_2, \quad (3.28)$$

This selection gives good results but, at it is related to the dimension n of the space, it needs more computation times and arithmetic operations for large problems. In our numerical examples, we used 3.27 for large dimensions and 3.28 for small problems.

An adaptive order rational block Lanczos algorithm for the computation of the reduced-order system using the rational block Lanczos process (Algorithm 3) and the above adaptive approach for selecting the interpolation points can be summa-

rized as follows.

Algorithm 4 The Adaptive Order Rational Block Lanczos-type (AORBL) algorithm for model-order reduction(AORBL)

1. **Input:** The original system (A,B,C) , choose a tolerance tol and set $F_0 = I_p$.
 2. **Output:** The reduced system (A_m, B_m, C_m) .
 3. Define $\epsilon_m = 1$ and $m = 1$.
 4. **While** $(\epsilon_m > tol)$ **do**
 5. $[\mathbb{V}_m, \mathbb{W}_m] = \mathbf{Rational-Block-Lanczos}(A,B,C,\Sigma_m)$.
 6. Compute the reduced model $A_m = \mathbb{W}_m^T A \mathbb{V}_m, B_m = \mathbb{W}_m^T B, C_m = C \mathbb{V}_m$ and the corresponding transfer function F_m .
 7. Compute the error estimation $\epsilon_m = \|F_m - F_{m-1}\|_\infty$.
 8. Set $m = m + 1$.
 9. **end while**.
-

Notice that, for choosing the interpolation points, we can also use one of the approximated error expressions listed in Table 3.1. The way of choosing these parameters affects the speed of convergence of the algorithm.

Remark 3.4.2 *For large problems, the total number of arithmetic operations after m iterations is dominated by $\mathcal{O}(mpn^2)$ and also LU factorizations for solving shifted linear systems with the shifted matrices $A - \sigma_i I_n$ (Line 3 and Line 9 of Algorithm 3). One can also use solvers such as GMRES with a pre-conditioner for solving these shifted linear systems.*

3.5 The rational block Lanczos algorithm for the computation of the matrix exponential $e^{tA}B$

The problem of approximating the matrix

$$U(t) = e^{tA}B \tag{3.29}$$

for a fixed constant $t \in \mathbb{R}$, a given matrix B of $\mathbb{R}^{n \times p}$ and $A \in \mathbb{R}^{n \times n}$ is of considerable importance in many applications. In fact, it is the core of many exponential integrators for solving systems of ordinary differential equations (see [82, 83]) or time dependent partial differential equations [54, 62]. We denote tA by A from now on unless otherwise stated.

Over the years, several methods have been proposed to consider the numerical approximation of the matrix exponential to a block vector B for $1 \leq p \ll n$. In 1978, Moler and van Loan [106] published their famous paper discussing nineteen dubious ways to compute the exponential of a matrix. Since then, Krylov subspace methods have been an important development towards tackling the problem (3.29) when the matrix A is very large and sparse.

Generally speaking, there are two classes of Krylov subspace methods for evaluating (3.29) when A is large and sparse [114]. In the first class of methods, the matrix is projected into a much smaller subspace, then the exponential is applied to the reduced matrix, and finally the approximation is projected back to the original large space [17, 41, 43, 45, 108, 123]. The second class of methods is a direct approximation approach where e^A is replaced by an explicitly computed rational function r such that $r(A) \approx e^A$, and then the action of the matrix exponential is evaluated [55, 56, 102, 137, 138, 144]. However, all these methods have in common the fact that linear system solves with (shifted versions of) A are required, and in rational Krylov methods one typically solves one linear system per iteration. Therefore a rational Krylov iteration may be considerably more expensive (in terms of computation time) than a polynomial Krylov iteration, which involves only a matrix-vector product with A . The applicability of rational Krylov methods hinges on the efficiency by which these linear systems can be solved. Since rational functions may exhibit approximation properties superior to

polynomials, the number of overall iterations required by rational Krylov methods is hopefully smaller than that required by polynomial methods, provided that the poles of the rational functions involved have been chosen in a suitable way.

The aim of this section is to use the first class of methods described above to approximate the problem (3.29) for $p > 1$, and by using the rational block Lanczos method proposed in last sections. For the rational Krylov methods, the desired approximation has the following form

$$r_{m-1}(A)B, \quad (3.30)$$

where r_{m-1} is a rational function of the type $(m-1, m-1)$, i.e., such that the denominator has the same degree $m-1$ as the numerator. On the theoretical side we generalize some of the error estimates and a priori error bounds proved in [123] on to the rational case. The proposed results are very general and can be applied in other contexts than the approximation of the exponential.

3.5.1 Polynomial approximation

In this section we recall the problem of computing an approximation to the matrix $e^A B$ by using polynomial approximation [64, 123], i.e., in this case we seek an approximation of the form

$$e^A B \approx p_{m-1}(A)B, \quad (3.31)$$

where p_{m-1} is a polynomial of degree $m-1$. Since this approximation is an element of the block Krylov subspace

$$\mathbb{K}_m(A, B) = \text{Range}\{B, AB, \dots, A^{m-1}B\},$$

the problem can be reformulated as that of finding an element of $\mathbb{K}_m(A, B)$ that approximates $U = e^A B$. A well known algorithm for building a convenient basis of $\mathbb{K}_m(A, B)$ is the block Lanczos algorithm described in last chapter. Because the bi-orthogonality condition of the matrices \mathbb{V}_m and \mathbb{W}_m generated by this process, we have $\mathbb{T}_m = \mathbb{W}_m^T A \mathbb{V}_m$ and as a result \mathbb{T}_m represents the projection of the linear transformation A to the subspace $\mathbb{K}_m(A, B)$. The matrix $X_{opt} = \mathbb{V}_m \mathbb{W}_m^T e^A B$ is the projection of $e^A B$ on $\mathbb{K}_m(A, B)$, i.e., it is the closest approximation to $e^A B$ from $\mathbb{K}_m(A, B)$. Let $\beta \in \mathbb{R}^{p \times p}$ and $V_1 \in \mathbb{R}^{n \times p}$ as defining in algorithm 1, it follows

immediately that

$$\mathbb{V}_m \mathbb{W}_m^T e^A B = \mathbb{V}_m \mathbb{W}_m^T e^A V_1 \beta = \mathbb{V}_m \mathbb{W}_m^T e^A \mathbb{V}_m E_1 \beta,$$

the optimal fit is therefore $X_{opt} = \mathbb{V}_m Y_{opt}$ in which $Y_{opt} = \mathbb{W}_m^T e^A \mathbb{V}_m E_1 \beta$. Unfortunately, Y_{opt} is not practically computable, since it involves e^A . The alternative which will be used throughout is to approximate $\mathbb{W}_m^T e^A \mathbb{V}_m$ by $e^{\mathbb{T}_m}$, leading to the approximation $Y_{opt} \approx e^{\mathbb{T}_m} E_1 \beta$, where E_1 is first $mp \times p$ block of the identity matrix I_{mp} . Then we have

$$e^A B \approx \mathbb{V}_m e^{\mathbb{T}_m} E_1 \beta = \mathbb{V}_m e^{\mathbb{T}_m} \mathbb{W}_m^T B. \quad (3.32)$$

We are now left with the problem of computing efficiently the block vector $e^{\mathbb{T}_m} E_1 \beta$ which is similar to the problem we started with but typically of much smaller size.

3.5.2 Rational approximation

In this section, we use the rational block Lanczos procedure described in previous sections to approximate the problem (3.29). In general, rational Krylov methods for computing $e^A B$ all have in common the fact that an approximation at iteration m is of the form $r_{m-1}(A)B$, where r_{m-1} is a $(m-1, m-1)$ rational function with a prescribed denominator polynomial $q_{m-1} \in \mathcal{P}_{m-1}$.

The rational approximation is already defined for the rational Arnoldi procedure; see [40, 73], and also for the special case of extended Krylov subspaces [43, 132]. In a similar manner, and using the bi-orthogonality of the bases generated by the rational block Lanczos algorithm, the rational block Lanczos approximation for $e^A B$ can be defined as

$$U_m = \mathbb{V}_m e^{A_m} \mathbb{W}_m^T B, \quad \text{where } A_m = \mathbb{W}_m^T A \mathbb{V}_m.$$

In this case, only the computation of a matrix function e^{A_m} of size $mp \times mp$ is required, which is small compared to the original e^A problem of size $n \times n$. To simplify notations, in rational block Lanczos algorithm (Algorithm 3), we construct the $p \times p$ matrix $\tilde{\beta}$ such that $B = V_1 \tilde{\beta}$. Then the rational approximation U_m will be expressed as

$$U_m = \mathbb{V}_m e^{A_m} E_1 \tilde{\beta}. \quad (3.33)$$

The rational approximation enjoys several remarkable properties. First of all, is the exactness. This property is well known for polynomial Arnoldi approximations [38, 123], and generalizes to the rational Krylov case, either for the special case of extended Krylov subspaces [40], or for the general case in [73].

Lemma 3.5.1 *Let \mathbb{V}_m and \mathbb{W}_m be the bases generated by the rational block Lanczos algorithm (Algorithm 3), and $A_m = \mathbb{W}_m^T A \mathbb{V}_m$. Then we have*

$$r_{m-1}(A)B = \mathbb{V}_m r_{m-1}(A_m) E_1 \tilde{\beta}, \quad (3.34)$$

i.e., the approximation for $r_{m-1}(A)B$ is exact (provided that $r_{m-1}(A_m)$ is defined), where r_{m-1} is a rational function of the type $(m-1, m-1)$, i.e., such that the denominator has the same degree $m-1$ as the numerator,

The following result states that a rational approximation from a rational Krylov subspace is closely related to rational interpolation at the rational Ritz values $\Lambda(A_m)$.

Lemma 3.5.2 *Under the assumptions of Lemma 3.5.1, the following equality holds*

$$\mathbb{V}_m e^{A_m} E_1 \tilde{\beta} = r_{m-1}(A)B, \quad (3.35)$$

where r_{m-1} interpolates the exponential function at the Ritz values $\Lambda(A_m)$.

The following Lemma generalizes a result introduced in [63, 123]. It shows how to systematically exploit rational approximations to e^x , in order to establish a priori error bounds.

Lemma 3.5.3 *Let A be an arbitrary matrix and $\mathbb{V}_m, \mathbb{W}_m$ the results of m steps of Algorithm 3. Let $f(z)$ be any function such that $f(A)$ and $f(A_m)$ are defined and $A_m = \mathbb{W}_m^T A \mathbb{V}_m$. Let r_{m-1} be any rational function of the type $(m-1, m-1)$ approximating $f(z)$, and define the remainder $\tilde{r}_m(z) = e^z - r_{m-1}(z)$. Then,*

$$\|f(A)B - \mathbb{V}_m f(A_m) E_1 \tilde{\beta}\|_2 \leq (\|\tilde{r}_m(A)\|_2 + \|\mathbb{V}_m\|_2 \|\tilde{r}_m(A_m)\|_2) \|\tilde{\beta}\|_2. \quad (3.36)$$

Proof. As a result of the relation $f(z) = r_{m-1}(z) + \tilde{r}_m(z)$ we have

$$f(A)V_1 = r_{m-1}(A)V_1 + \tilde{r}_m(A)V_1. \quad (3.37)$$

Lemma (3.5.2) implies that

$$r_{m-1}(A)V_1 = \mathbb{V}_m r_{m-1}(A_m)E_1. \quad (3.38)$$

Similarly to (3.37) we can write

$$r_{m-1}(A_m)E_1 = f(A_m)E_1 - \tilde{r}_m(A_m)E_1. \quad (3.39)$$

Multiply (3.39) by \mathbb{V}_m and substitute the resulting equation in (3.38) gives

$$r_{m-1}(A)V_1 = \mathbb{V}_m f(A_m)E_1 - \mathbb{V}_m \tilde{r}_m(A_m)E_1. \quad (3.40)$$

Substitute (3.40) in (3.37) to get, after multiplying by $\tilde{\beta}$

$$f(A)B = \mathbb{V}_m f(A_m)E_1 \tilde{\beta} + (\tilde{r}_m(A)V_1 - \mathbb{V}_m \tilde{r}_m(A_m)E_1) \tilde{\beta}. \quad (3.41)$$

Finally we have

$$\|f(A)B - \mathbb{V}_m f(A_m)E_1 \tilde{\beta}\|_2 \leq (\|\tilde{r}_m(A)\|_2 + \|\mathbb{V}_m\|_2 \|\tilde{r}_m(A_m)\|_2) \|\tilde{\beta}\|_2.$$

3.6 Numerical experiments

In this section, we give some experimental results to show the effectiveness of the proposed AORBL algorithm when applied to model order reduction problem. All the experiments were performed on a computer of Intel Core i5 at 1.3GHz and 8GB of RAM. The algorithms were coded in Matlab 8.0. We give some numerical tests to show the performance of the adaptive-order rational block Lanczos-type (AORBL) algorithm. In all the presented experiments, $tol = 10^{-8}$ and the AORBL algorithm is stopped when the \mathcal{H}_∞ -error

$$\epsilon_m = \|F_m - F_{m-1}\|_\infty$$

between the previous reduced system and the current one is less than tol , where the \mathcal{H}_∞ -norm of the error is given as (cf., e.g., [4], sec.5.3)

$$\|F_m - F_{m-1}\|_\infty = \sup_{\omega \in \mathbb{R}} \|F_m(j\omega) - F_{m-1}(j\omega)\|_2,$$

$\omega \in [10^{-3}, 10^3]$ and $j = \sqrt{-1}$.

To compute the \mathcal{H}_∞ -norm, the following functions from LYAPACK [113] are used :

- `lp_lgfrq`: Generates the set of logarithmically distributed frequency sampling points.

- `lp_para`: Used for computing the initial first two shifts.
- `lp_gnorm`: Computes $\|F_m(j\omega) - F_{m-1}(j\omega)\|_2$.

In our experiments, we used some matrices from `LYAPACK`. These matrix tests are reported in the following Table 3.2. For the FOM model [110], we notice that

Table 3.2: The matrix tests.

Matrices	sizes
CD-Player	$n = 120, p = 2$
Rail3113	$n = 3113, p = 6$
Modified FOM	$n = 1006, p = 6$
ISS	$n = 270, p = 3$
fdm	$n = 40.000, p = 5$

originally, the model is SISO system and we modified the inputs and outputs to get a MIMO system. The matrices B and C are then given by

$$B = [b_1, \dots, b_6], \quad C^T = [c_1, \dots, c_6],$$

where

$$b_1^T = c_1 = (\underbrace{10, \dots, 10}_6, \underbrace{1, \dots, 1}_{1000}), \quad \text{and} \quad b_2, \dots, b_6; c_2, \dots, c_6$$

are random column vectors.

For the `fdm` model, the corresponding matrix A is obtained from the centered finite difference discretization of the operator

$$L_A(u) = \Delta u - f(x, y) \frac{\partial u}{\partial x} - g(x, y) \frac{\partial u}{\partial y} - h(x, y)u,$$

on the unit square $[0, 1] \times [0, 1]$ with homogeneous Dirichlet boundary conditions with

$$\begin{cases} f(x, y) &= \log(x + 2y), \\ g(x, y) &= e^{x+y}, \\ h(x, y) &= x + y, \end{cases}$$

and the matrices B and C were random matrices with entries uniformly distributed in $[0, 1]$. The number of inner grid points in each direction was $n_0 = 200$ and the dimension of A is $n = n_0^2 = 40.000$.

Example 1. For this experiment, we used the modified FOM model with $m = 18$. The top plots of Figure 3.1 show the frequency response of the original system (circles) compared to the frequency response of its approximation (solid plot). The bottom plot of this figure represents the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for different frequencies.

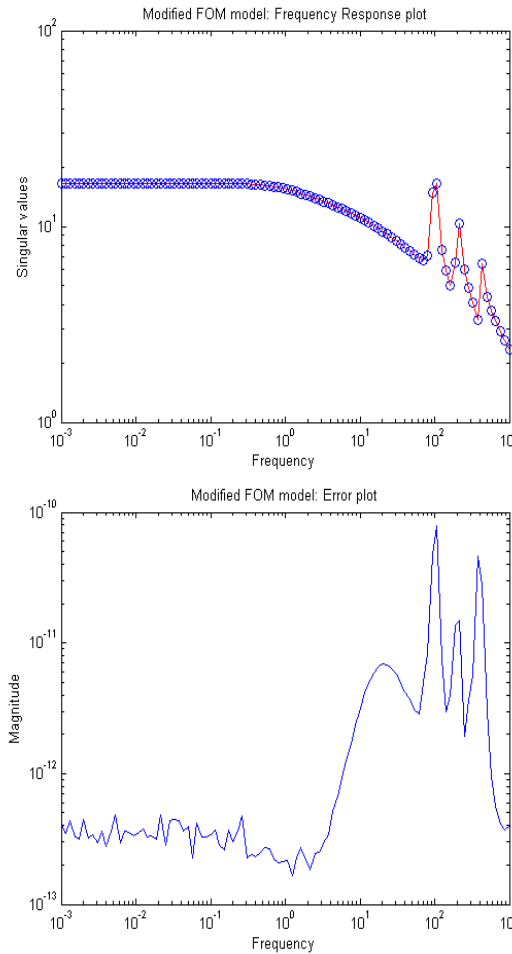


Figure 3.1: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the modified FOM model with $m = 18$.

Example 2. In this experiment and as a first test model, we considered the ISS example of dimension $n = 270$ with 3 inputs and 3 outputs, and we plotted

the \mathcal{H}_∞ error norm $\|F - F_m\|_\infty$ versus the number m of iterations. As can be shown from this plot, the AORBL algorithm gives good result with small values of m .

Example 3. We consider the well known CD player model. This is a small

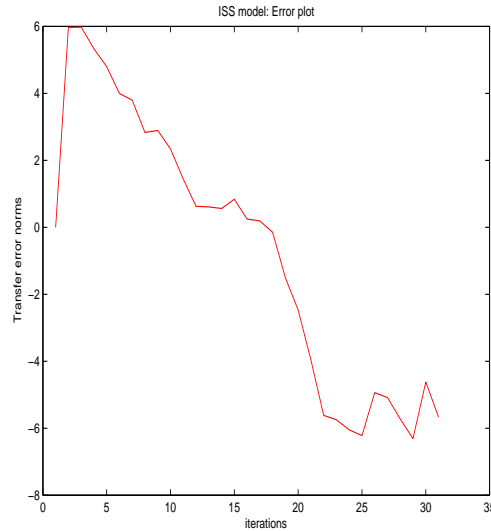


Figure 3.2: The \mathcal{H}_∞ error $\|F - F_m\|_\infty$ versus the number of iterations for the ISS model.

dimension example but generally difficult and is always considered as a benchmark test. The top plots of Figure 3.3 represent the sigma plots of the original system (circles) and the reduced order system (solid line). For the bottom curve, we plotted the error norm $\|F(s) - F_m(s)\|_2$ versus the frequencies.

Example 4. In the last example we compared the AORBL algorithm with IRKA method. We used four models: the CD player, the ISS, the Rail3113 [110] and the fdm model ($n = 40000$, $p = 5$). In Table 3.3, we listed the obtained \mathcal{H}_∞ norm of the error transfer function $\|F - F_m\|_\infty$, the corresponding cpu-time, the number of required iterations for the two methods and in parentheses we also gave the used space dimension for IRKA. A maximum number of $m_{max} = 500$ iterations

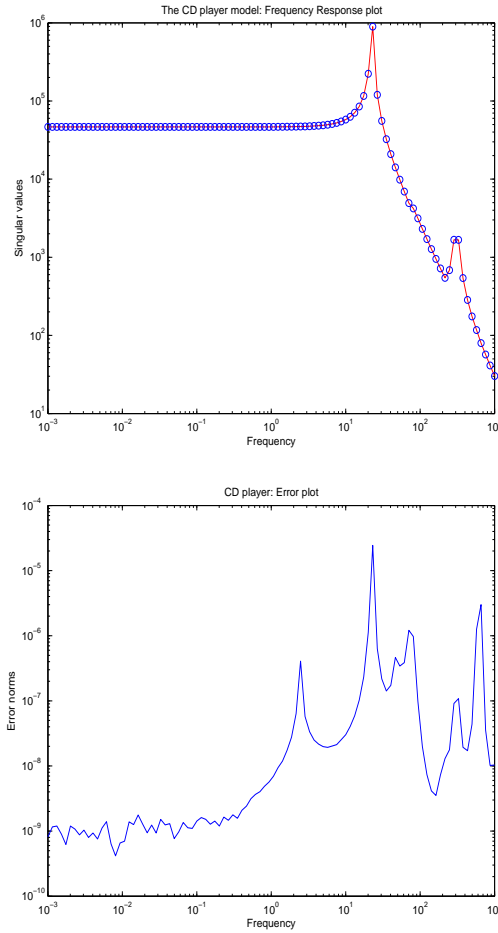


Figure 3.3: The CD player model with $m = 30$. Top: The singular values of the exact transfer function (circles) and its approximation (solid) versus the frequencies. Bottom: The error norms $\|F(s) - F_m(s)\|_2$.

was allowed to the two algorithms. As observed from Table 3.3, IRKA and AORBL return similar results (computing times and norms of the errors) for the first two models with an advantage for AORBL. However, for the last two examples, IRKA didn't converge within the allowed maximum number of iterations.

Table 3.3: Comparison between IRKA and AORBL for CD player, ISS, Rail3113 and fdm models.

	AORBL			IRKA ($tol = 10^{-4}$)		
	\mathcal{H}_∞ error	# iter.	time	\mathcal{H}_∞ error	# iter. (dim)	time
CD player	2.6e-06	25	1.1s	1.5e-04	42(35)	1.4s
ISS	3.8e-05	20	1.2s	1.2e-04	54(30)	4.9s
Rail3113	1.1e-07	30	4.5s	–	–	–
fdm ($n = 4.10^4$)	4.5e-08	35	53s	–	–	–

3.7 Conclusion

In this chapter, we proposed a new adaptive rational block Lanczos process and an adaptive method for choosing the interpolation points with applications in model order reduction of multi-input and multi-output first-order stable linear dynamical systems. Moreover, we established new Lanczos-like expressions and new error estimations between the original and the reduced transfer functions. We presented some numerical results to confirm the good performance of the rational block Lanczos subspace method compared with other known method. The proposed procedure is tested on well known benchmark problems of medium and large dimensions and the numerical results show that the adaptive approach allows one to obtain reduced order models of small dimension.

4

A modified nonsymmetric rational block Lanczos method for model reduction in large scale LTI dynamical systems

In this chapter, we propose a modified non-symmetric rational block Lanczos algorithm. This process can be considered as a generalization of the rational block Lanczos algorithm given in the last chapter where different multiplicities are considered for each interpolation point. In the second section of this chapter we show how to obtain new Lanczos-like equations for the rational block case. Simple residual error expressions are developed in Section 3. In Section 4 we propose an adaptive choice to generate the set of interpolation points. The last section is devoted to some numerical experiments to show the accuracy of the proposed methods.

4.1 The modified rational block Lanczos algorithm

Let $\Sigma_K = \{\sigma_1, \dots, \sigma_K\}$ and $\tilde{\Sigma}_K = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_K\}$ be two sets of interpolation points, with multiplicities m_1, \dots, m_K , and $\tilde{m}_1, \dots, \tilde{m}_K$, respectively. The column vectors of the matrices \mathbb{V}_m and \mathbb{W}_m generated by the modified rational block Lanczos algorithm are determined from the K block Krylov subspaces $\mathbb{K}_{m_i}(A, B, \sigma_i)$ and $\mathbb{K}_{\tilde{m}_i}(A^T, C^T, \tilde{\sigma}_i)$, respectively, for $i = 1, \dots, K$. Therefore, at each iteration i , the modified rational block Lanczos algorithm generate the matrices $\mathbb{V}_{m_i} \in \mathbb{R}^{n \times m_i}$ and $\mathbb{W}_{\tilde{m}_i} \in \mathbb{R}^{n \times \tilde{m}_i}$ whose column-space spans the block Krylov subspaces $\mathbb{K}_{m_i}(A, B, \sigma_i)$ and $\mathbb{K}_{\tilde{m}_i}(A^T, C^T, \tilde{\sigma}_i)$, respectively. From each of these subspaces, the m_i and \tilde{m}_i column vectors are used to generate the matrices \mathbb{V}_m and \mathbb{W}_m , respectively, such that

$$\mathbb{V}_m = [\mathbb{V}_{m_1}, \mathbb{V}_{m_2}, \dots, \mathbb{V}_{m_K}] \text{ and } \mathbb{W}_m = [\mathbb{W}_{\tilde{m}_1}, \mathbb{W}_{\tilde{m}_2}, \dots, \mathbb{W}_{\tilde{m}_K}],$$

where $\mathbb{V}_m, \mathbb{W}_m \in \mathbb{R}^{n \times mp}$ and $m = \sum_{i=1}^K m_i = \sum_{i=1}^K \tilde{m}_i$.

As in the last chapter, we give now the result which shows how we can construct the bi-orthogonal bases \mathbb{V}_m and \mathbb{W}_m of the rational Krylov subspaces so that the multi-point rational interpolation problem is solved. This result is proven in [71] for SISO systems and extended to the MIMO case in [57].

Theorem 4.1.1 *Let $\Sigma_K = \{\sigma_1, \dots, \sigma_K\}$ and $\tilde{\Sigma}_K = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_K\}$ be two sets of interpolation points, with multiplicities m_1, \dots, m_K , and $\tilde{m}_1, \dots, \tilde{m}_K$, respectively.*

If $\mathbb{V}_m, \mathbb{W}_m \in \mathbb{R}^{n \times mp}$ satisfy

$$\cup_{k=1}^K \mathbb{K}_{m_k}(A, B, \sigma_k) \subseteq \text{Range}(\mathbb{V}_m)$$

$$\cup_{k=1}^K \mathbb{K}_{\tilde{m}_k}(A^T, C^T, \tilde{\sigma}_k) \subseteq \text{Range}(\mathbb{W}_m)$$

where $\sum_{k=1}^K m_k = \sum_{k=1}^K \tilde{m}_k = m$. Then, assuming that $(A - \sigma I_n)^{-1}$ exists for all $\sigma \in \Sigma_K \cup \tilde{\Sigma}_K$,

- *if $\sigma_k = \tilde{\sigma}_k$, $F_m(s)$ matches the first $m_k + \tilde{m}_k$ moments of the original transfer function $F(s)$ at σ_k*
- *if $\sigma_k \neq \tilde{\sigma}_k$, $F_m(s)$ matches the first m_k of $F(s)$ at σ_k and the first \tilde{m}_k moments of $F(s)$ at $\tilde{\sigma}_k$, respectively.*

The modified rational block Lanczos process is summarized in Algorithm 5. It is a generalization of the one given in [51] to the block case. For simplicity of the presentation we assume that $m_k = \tilde{m}_k$ and also it is assumed that $\sigma_i \neq \sigma_j$ and $\tilde{\sigma}_i \neq \tilde{\sigma}_j$ for $i \neq j$.

Algorithm 5 The modified rational block Lanczos algorithm (MRBLA)

1. **Input:** $\Sigma_K = \{\sigma_1, \dots, \sigma_K\}$, $\tilde{\Sigma}_K = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_K\}$, A, B, C and $m_k = \tilde{m}_k$
 2. **Initialize:** $\mathbb{V}_m = []$, $\mathbb{W}_m = []$ and $i = 0$
 - function $[\mathbb{V}_m, \mathbb{W}_m] = \mathbf{Modified_rational_Block_Lanczos}(A, B, C, \Sigma, \tilde{\Sigma})$
 3. **for** $k = 1, \dots, K$
 4. **if** $\{\sigma_k = \infty\}$; $S_{i+1} = B$ **else** $S_{i+1} = (A - \sigma_k I_n)^{-1} B$; **end**
 5. **if** $\{\tilde{\sigma}_k = \infty\}$; $R_{i+1} = C^T$ **else** $R_{i+1} = (A - \tilde{\sigma}_k I_n)^{-T} C^T$ **end**
 6. $S_{i+1} = S_{i+1} - \mathbb{V}_m \mathbb{W}_m^T S_{i+1}$; $R_{i+1} = R_{i+1} - \mathbb{W}_m \mathbb{V}_m^T R_{i+1}$;
 7. $S_{i+1} = V_{i+1} H_{i+1,i}$; $R_{i+1} = W_{i+1} G_{i+1,i}$; (QR factorization);
 8. $W_{i+1}^T V_{i+1} = P_i D_i Q_i^T$; (Singular Value Decomposition);
 9. $V_{i+1} = V_{i+1} Q_i D_i^{-1/2}$; $W_{i+1} = W_{i+1} P_i D_i^{-1/2}$;
 10. $\mathbb{V}_m = [\mathbb{V}_m, V_{i+1}]$; $\mathbb{W}_m = [\mathbb{W}_m, W_{i+1}]$; $i = i + 1$;
 11. **for** $j = 1, \dots, m_k - 1$
 12. **if** $\{\sigma_k = \infty\}$; $S_{i+1} = A V_i$ **else** $S_{i+1} = (A - \sigma_k I_n)^{-1} V_i$; **end**
 13. **if** $\{\tilde{\sigma}_k = \infty\}$; $R_{i+1} = A^T W_i$ **else** $R_{i+1} = (A - \tilde{\sigma}_k I_n)^{-T} W_i$ **end**
 14. $S_{i+1} = S_{i+1} - \mathbb{V}_m \mathbb{W}_m^T S_{i+1}$; $R_{i+1} = R_{i+1} - \mathbb{W}_m \mathbb{V}_m^T R_{i+1}$;
 15. $S_{i+1} = V_{i+1} H_{i+1,i}$; $R_{i+1} = W_{i+1} G_{i+1,i}$ (QR factorization);
 16. $W_{i+1}^T V_{i+1} = P_i D_i Q_i^T$ (Singular Value Decomposition);
 17. $V_{i+1} = V_{i+1} Q_i D_i^{-1/2}$; $W_{i+1} = W_{i+1} P_i D_i^{-1/2}$;
 18. $\mathbb{V}_m = [\mathbb{V}_m, V_{i+1}]$; $\mathbb{W}_m = [\mathbb{W}_m, W_{i+1}]$; $i = i + 1$;
 19. **end**
 20. **if** $\{k = K\}$
 21. $S_{i+1} = A^{m_\infty} B$ and $R_{i+1} = (A^{\tilde{m}_\infty})^T C^T$;
 22. $S_{i+1} = S_{i+1} - \mathbb{V}_m \mathbb{W}_m^T S_{i+1}$; $R_{i+1} = R_{i+1} - \mathbb{W}_m \mathbb{V}_m^T R_{i+1}$;
 23. $S_{i+1} = V_{i+1} H_{i+1,i}$; $R_{i+1} = W_{i+1} G_{i+1,i}$ (QR factorization);
 24. $W_{i+1}^T V_{i+1} = P_i D_i Q_i^T$ (Singular Value Decomposition);
 25. $V_{i+1} = V_{i+1} Q_i D_i^{-1/2}$; $W_{i+1} = W_{i+1} P_i D_i^{-1/2}$;
 26. $\mathbb{V}_{m+1} = [\mathbb{V}_m, V_{m+1}]$; $\mathbb{W}_{m+1} = [\mathbb{W}_m, W_{m+1}]$;
 27. **end**
 28. **end.**
-

4.2 Lanczos-like equations for the rational block case

Rational Krylov algorithms are always related to a set of equations that relate the bases constructed by these procedures and the matrices of the original system; see [13, 59, 57, 111, 122] and references therein. However, these equations are not in the standard form. Her, we show that the modified rational block Lanczos process proposed in last section allow to obtain Lanczos-like equations for the rational case. This result is first proposed in [51, 50] for the standard rational Lanczos algorithm and extended her to the block case.

Theorem 4.2.1 *Let \mathbb{V}_{m+1} and \mathbb{W}_{m+1} be the matrices generated by the modified rational block Lanczos algorithm (Algorithm 5), then we have*

$$\text{Range}[\mathbb{V}_m, A^{m_\infty} B] \subseteq \text{Range}\{\mathbb{V}_{m+1}\}, \quad \text{Range}[\mathbb{W}_m, (A^{\tilde{m}_\infty})^T C^T] \subseteq \text{Range}\{\mathbb{W}_{m+1}\},$$

and

$$\mathbb{W}_{m+1}^T \mathbb{V}_{m+1} = I_{m+1},$$

where m_∞ and \tilde{m}_∞ are the multiplicities of ∞ in Σ and $\tilde{\Sigma}$, respectively.

Moreover, we have the following Lanczos-like relations

$$A\mathbb{V}_m = \mathbb{V}_m A_m + V_{m+1} P_{m+1}, \quad (4.1)$$

$$B = \mathbb{V}_m B_m + V_{m+1} b_m, \quad (4.2)$$

$$A^T \mathbb{W}_m = \mathbb{W}_m A_m^T + W_{m+1} Q_{m+1}, \quad (4.3)$$

$$C^T = \mathbb{W}_m C_m^T + W_{m+1} c_m^T, \quad (4.4)$$

where $b_m = W_{m+1}^T B$, $c_m = C V_{m+1}$, $P_{m+1} = W_{m+1}^T A \mathbb{V}_m$ and $Q_{m+1} = V_{m+1}^T A^T \mathbb{W}_m$. Furthermore, $b_m = 0$ if $m_\infty > 0$ and $c_m = 0$ if $\tilde{m}_\infty > 0$.

Proof. We first prove the result for $m_\infty = 0$. Let \mathbb{V}_m be defined as in Theorem 4.2.1. Then we extend \mathbb{V}_m to $\mathbb{V}_{m+1} = [\mathbb{V}_m, V_{m+1}]$ such that

$$\text{Range}[\mathbb{V}_m, B] \subseteq \text{Range}\{\mathbb{V}_{m+1}\}$$

by biorthogonalising B against all previous columns of \mathbb{W}_m with the Lanczos algorithm. Then the following relations are true:

$$\text{Range}\{(A - \sigma_1 I_n)^{-1} B\} \subset \text{Range}\{\mathbb{V}_1\}, \quad (4.5)$$

$$\begin{aligned} \text{Range}\{(A - \sigma_k I_n)^{-(i-1)} B\} &\subset \text{Range}\{\mathbb{V}_{j-1}\} \\ \text{Range}\{(A - \sigma_k I_n)^{-i} B\} &\subset \text{Range}\{\mathbb{V}_j\} \end{aligned} \quad (4.6)$$

and

$$\text{Range}\{B\} \subset \text{Range}\{\mathbb{V}_{m+1}\},$$

where $\mathbb{V}_m = [V_1, V_2, V_3, \dots, V_m] \in \mathbb{R}^{n \times mp}$, $m = \sum_{i=1}^K m_i$ and \mathbb{V}_j is the matrix constructed at the $(k)_{th}$ interpolation point and for the i^{th} multiplicity, i.e., $j = \sum_{l=1}^{k-1} m_l + i$. We start by proving the theorem for the first block column of \mathbb{V}_m . Multiply (4.5) by $(A - \sigma_1 I_n)$ from the left and rearrange to get

$$\text{Range}\{AV_1\} \subset \text{Range}\{\mathbb{V}_1, B\}$$

Then we have

$$\text{Range}\{AV_1\} \subset \text{Range}\{\mathbb{V}_{m+1}\}.$$

We proceed the proof by induction. We assume that the result holds for an arbitrary interpolation point σ_k of the \mathbb{K}_{σ_k} Krylov subspace up to the $(i-1)^{th}$ multiplicity. We will prove the result for the next multiplicity. Therefore we assume

$$\text{Range}\{A\mathbb{V}_{j-1}\} \subset \text{Range}\{\mathbb{V}_{m+1}\} \quad (4.7)$$

and then we prove that

$$\text{Range}\{A\mathbb{V}_j\} \subset \text{Range}\{\mathbb{V}_{m+1}\}.$$

Multiply (4.6) from the left by $(A - \sigma_k I_n)$ and rearrange to get:

$$\text{Range}\{A\mathbb{V}_j\} \subset \text{Range}\{\mathbb{V}_j, A\mathbb{V}_{j-1}\}$$

which gives

$$\text{Range}\{A\mathbb{V}_j\} \subset \text{Range}\{\mathbb{V}_{m+1}\}.$$

Combining the last relation with the assumption made in (4.7) gives

$$\text{Range}\{A\mathbb{V}_j\} \subset \text{Range}\{\mathbb{V}_{m+1}\}. \quad (4.8)$$

Therefore it is easy to see that the result in (4.8) holds for all columns in \mathbb{V}_m , i.e.,

$$\text{Range}\{A\mathbb{V}_m\} \subset \text{Range}\{\mathbb{V}_{m+1}\},$$

and then there exists a matrix $Y \in \mathbb{R}^{(m+1)p \times mp}$ such that

$$A\mathbb{V}_m = \mathbb{V}_{m+1}Y.$$

Setting $Y = \begin{pmatrix} \tilde{A}_m \\ P_{m+1} \end{pmatrix}$, we obtain

$$A\mathbb{V}_m = \mathbb{V}_{m+1} \begin{pmatrix} \tilde{A}_m \\ P_{m+1} \end{pmatrix}.$$

Using the bi-orthogonality between \mathbb{W}_{m+1} and \mathbb{V}_{m+1} gives $\tilde{A}_m = A_m = \mathbb{W}_m^T A \mathbb{V}_m$ and $P_{m+1} = W_{m+1}^T A \mathbb{V}_m$. Therefore we get

$$A\mathbb{V}_m = \mathbb{V}_m A_m + V_{m+1} P_{m+1}.$$

Similarly to prove (4.2) we proceed as follows: Since $\text{Range}\{B\} \subset \text{Range}\{\mathbb{V}_{m+1}\}$ there exists a matrix $Z \in \mathbb{R}^{(m+1)p \times p}$ such that

$$B = \mathbb{V}_{m+1} Z. \quad (4.9)$$

Setting $Z = \begin{pmatrix} \tilde{B}_m \\ b_m \end{pmatrix}$, and multiplying (4.9) by \mathbb{W}_{m+1}^T from the left we get

$$\mathbb{W}_{m+1}^T B = \begin{pmatrix} \tilde{B}_m \\ b_m \end{pmatrix}. \quad (4.10)$$

Then $\tilde{B}_m = B_m = \mathbb{W}_m^T B$, $b_m = W_{m+1}^T B$ and consequently we obtain the following relation

$$B = \mathbb{V}_m B_m + V_{m+1} b_m.$$

Assume now that $m_\infty > 0$. The matrix \mathbb{V}_m verifies

$$\text{Range}[B \ AB \ \dots \ A^{p-1}B \ \mathbb{V}_{m-p}] \subset \text{Range}\{\mathbb{V}_m\}$$

where $p < m$. Since B is already in the Range of \mathbb{V}_m it is easy to see that the relation (4.1) will be satisfied if

$$\text{Range}\{AV_1, \dots, AV_p\} \subset \text{Range}\{\mathbb{V}_{m+1}\}. \quad (4.11)$$

This can be shown by setting $\mathbb{V}_{m+1} = [\mathbb{V}_m, V_{m+1}]$ and then

$$\text{Range}[\mathbb{V}_m, A^p B] \subset \text{Range}\{\mathbb{V}_{m+1}\} \text{ and } \mathbb{W}_{m+1}^T \mathbb{V}_{m+1} = I_{m+1}$$

which is obtained by bi-orthogonalising $A^p B$ against all the previous columns of \mathbb{W}_m . It follows that (4.11) holds since by construction we have that,

$$\text{Range}\{A^k B\} \subset \begin{cases} \text{Range}\{V_1, \dots, V_{k+1}\}, & \text{for } 0 < k < p \\ \text{Range}\{V_1, \dots, V_{m+1}\}, & \text{for } k = p \end{cases} \quad (4.12)$$

which completes the proof of (4.1) and (4.2).

To prove the last part, note that if $m_\infty > 0$ then $B \in \text{Range}\{\mathbb{V}_m\}$ from which it follows that $b_m = 0$.

In a similar way, the relations (4.3) and (4.4) can be shown.

4.3 Residual error expressions

In last chapter, we utilized the rational Lanczos equations to simplify the residual error expressions. More simplified residual expressions will be established here using Lanczos-like equations obtained in last section.

4.3.1 Simplified Lanczos residual errors in the rational block Lanczos

Using the results of Theorem 4.2.1, we give new simple expressions of the residuals $R_B(s)$ and $R_C(s)$. In fact, from the residual expression $R_B(s)$ defined in (3.19) we get

$$\begin{aligned} R_B(s) &= B - (sI - A)\mathbb{V}_m(sI_{mp} - A_m)^{-1}B_m. \\ &= B - \mathbb{V}_m(sI_{mp} - A_m)(sI_{mp} - A_m)^{-1}B_m + V_{m+1}P_{m+1}(sI_{mp} - A_m)^{-1}B_m, \\ &= V_{m+1}(P_{m+1}(sI_{mp} - A_m)^{-1}B_m + b_m), \\ &= \tilde{B}\tilde{R}_B(s) \end{aligned}$$

where $\tilde{R}_B(s) = P_{m+1}(sI_{mp} - A_m)^{-1}B_m + b_m$ is the frequency dependent term of the residual error $R_B(s)$, and $\tilde{B} = V_{m+1}$ the non frequency dependent term of $R_B(s)$. In a similar way, we can use the expression of $R_C(s)$ in (3.19) to obtain the following relations

$$\begin{aligned} R_C(s) &= C^T - (sI - A)^T\mathbb{W}_m(sI_{mp} - A_m)^{-T}C_m^T \\ &= \tilde{C}^T\tilde{R}_C(s), \end{aligned}$$

where $\tilde{R}_C(s) = Q_{m+1}(sI_{mp} - A_m)^{-T}C_m^T + c_m^T$ is the frequency dependent term of the residual error $R_C(s)$, and $\tilde{C} = W_{m+1}$ is the frequency-independent terms of $R_C(s)$.

4.3.2 Error approximations

As in last chapter, the error expression in Theorem 3.3.1 can be expressed as

$$\epsilon(s) = \tilde{R}_C(s)^T \tilde{F}(s) \tilde{R}_B(s),$$

and we can use an approximation of $\tilde{F}(s)$ to simplify calculations. The different possible approximations of the error $\epsilon(s)$ are listed in the following table.

Table 4.1: Various estimations of the error

$\hat{\epsilon}(s) = \tilde{R}_B(s)$
$\hat{\epsilon}(s) = \tilde{R}_C(s)^T$
$\hat{\epsilon}(s) = \tilde{F}_m(s) \tilde{R}_B(s)$
$\hat{\epsilon}(s) = \tilde{F}_m(s)$
$\hat{\epsilon}(s) = \tilde{R}_C^T(s) \tilde{F}_m(s)$
$\hat{\epsilon}(s) = \tilde{R}_C^T(s) \tilde{F}_m(s) \tilde{R}_B(s)$

The simple approximations in Table 4.1 are the first two ones for which the computations require a small work as compared to the other choices. In the last section, we will give some numerical tests comparing these approximations.

4.4 An adaptive modified rational block Lanczos algorithm

As we already mentioned, the rational Krylov method has proven to be very effective for model reduction [71, 59, 60], but it has the drawback that the selection of interpolation points is a difficult task since it is an ad-hoc process. For a discussion on the choice of the interpolation points, see [75]. In this section we address this issue and we give an adaptive approach to choose these shifts. This adaptive method is first proposed in [20] for the case of standard rational Arnoldi algorithm. It is based on an approximation of upper bound of the error norm between

the original and the reduced transfer functions. Here, we use the same approach for our modified rational block Lanczos algorithm. The following result is the key ingredient of this method.

Proposition 4.4.1 *Let Σ_m and $\tilde{\Sigma}_m$ denote two given sets of interpolation points and let $\mathbb{V}_m, \mathbb{W}_m \in \mathbb{R}^{n \times mp}$ be the bi-orthogonal matrices computed by the modified rational block Lanczos algorithm. The following relation holds*

$$\|F(s) - F_m(s)\|_2 \leq \|CP^{-1}(s)\|_2 \|R_B(s)\|_2, \quad (4.13)$$

where $R_B(s) = B - P(s)\mathbb{V}_m P_m^{-1}(s)B_m$, $P(s) = sI - A$ and $B_m = \mathbb{W}_m^T B$.

Proof. From the expression of original and reduced transfer functions $F(s)$ and $F_m(s)$, we have

$$\begin{aligned} \|F(s) - F_m(s)\|_2 &= \|C(sI_n - A)^{-1}B - C_m(sI_{mp} - A_m)^{-1}B_m\|_2 \\ &= \|C(sI_n - A)^{-1}(B - (sI_n - A)\mathbb{V}_m(sI_{mp} - A_m)^{-1}B_m)\|_2 \\ &= \|CP(s)^{-1}(B - P(s)\mathbb{V}_m P_m(s)^{-1}B_m)\|_2 \\ &\leq \|CP(s)^{-1}\|_2 \|R_B(s)\|_2. \end{aligned}$$

Now, we can approximate the upper bound by employing the reduced order matrix triplet (A_m, B_m, C_m) and then $\|CP(s)^{-1}\|$ could be approximated by $\|C_m P_m(s)^{-1}\|$, where $P_m(s) = sI_{mp} - A_m$ and $A_m = \mathbb{W}_m^T A \mathbb{V}_m$.

Using the above approximation, the next shift σ_{k+1} can be selected as

$$\sigma_{k+1} = \arg \max_{s \in S} \|C_m P_m(s)^{-1}\|_2 \|R_B(s)\|_2. \quad (4.14)$$

We notice here that another simple way of choosing the shifts, is to consider in (5.16) only the second part which gives

$$\tilde{\sigma}_{k+1} = \arg \max_{s \in S} \|R_B(s)\|_2, \quad (4.15)$$

however, for some problems, choosing the interpolation points by using the formulation (4.14) gives more accurate results than those obtained with the expression (4.15).

Next, we combine the modified rational block Lanczos algorithm (Algorithm 5) and the adaptive approach explained above for selecting the interpolation points to

have an adaptive order rational block Lanczos algorithm for computing reduced-order system. This process can be summarized as follows.

Algorithm 6 The Adaptive Modified Rational Block Lanczos (AMRBL) algorithm for model-order reduction

1. **Input:** The original system (A, B, C) , the initial values $\sigma_1 = \tilde{\sigma}_1$, choose a tolerance tol and set $F_0 = I_p$.
 2. **Output:** The reduced system (A_m, B_m, C_m) .
 3. Initialize $\Sigma_1 = \{\sigma_1\}; \tilde{\Sigma}_1 = \{\tilde{\sigma}_1\}; m_1 = \tilde{m}_1 = 3; \epsilon_m = 1$ and $K = 1$;
 4. **While** ($\epsilon_m > tol$) **do**
 5. $[\mathbb{V}_m, \mathbb{W}_m] = \mathbf{Modified_rational_Block_Lanczos}(A, B, C, \Sigma_K, \tilde{\Sigma}_K)$
 6. Compute the reduced model $A_m = \mathbb{W}_m^T A \mathbb{V}_m, B_m = \mathbb{W}_m^T B, C_m = C \mathbb{V}_m$ and the corresponding transfer function F_m .
 7. Compute the next interpolation point $\sigma_{K+1} = \tilde{\sigma}_{K+1}$ using (4.14).
 8. Set $\Sigma_{k+1} = \{\Sigma_K, \sigma_{K+1}\}; \tilde{\Sigma}_{k+1} = \{\tilde{\Sigma}_K, \tilde{\sigma}_{K+1}\}; m_{K+1} = \tilde{m}_{K+1} = 3$
 9. Compute the error estimation $\epsilon_m = \|F_m - F_{m-1}\|_\infty$
 10. Set $K = K + 1$
 11. **end while.**
-

Remark : For choosing the interpolation points, we can also use one of the error approximation expressions listed in Table 4.1. In this case the interpolation points are selected to be the frequencies $\sigma \in \Sigma_K$ and $\tilde{\sigma} \in \tilde{\Sigma}_K$ at which one of the approximated error expressions achieves its maximum, i.e.,

$$\Sigma_K = \{\sigma : |\hat{e}(\sigma)| = \|\hat{e}\|_\infty\} \text{ and } \tilde{\Sigma}_K = \{\tilde{\sigma} : |\hat{e}(\tilde{\sigma})| = \|\hat{e}\|_\infty\}.$$

4.5 Numerical results

In this section, we give some experimental results to show the effectiveness of the adaptive modified rational block Lanczos (AMRBL) algorithm for model reduction in large LTI dynamical systems. All the experiments were performed on a computer of Intel Core i5 at 1.3GHz and 8Go of RAM. The algorithms were coded in Matlab 8.0. In all the presented numerical testes, we used as a tolerance $tol = 10^{-5}$ and the while-loop in Algorithm 6 is stopped when the error

$$\epsilon_m = \|F_m - F_{m-1}\|_\infty < tol$$

where $\omega \in [10^{-6}, 10^6]$.

In all the experiments of this section, we consider the special case where the sequences of shifts $\{\sigma_i\}_{i=1}^m$ and $\{\tilde{\sigma}_i\}_{i=1}^m$ are equal. To compute the set of frequency S and ω , we used the function `lp_lgfrq`.

Example 1. For this example, we applied the AMRBL on the modified FOM model and ISS model to get a reduced order systems of dimensions $m = 40$ and $m = 45$, respectively. The top plots of Figure 4.1 (modified FOM) and Figure 4.2 (ISS) show the frequency responses of the original system (circles) compared with the frequency responses of its approximation (solid plot). The bottom plot of these figures represent the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for different frequencies $\omega \in [10^{-6}, 10^6]$.

Example 2. For this experiment, we considered the `fdm` [113] and the `Rail821` [110] models. We plotted the \mathcal{H}_∞ relative error norm

$$\frac{\|F - F_m\|_\infty}{\|F\|_\infty}$$

versus the number m of iterations. For the `fdm` model, we consider the operator

$$L_A(u) = \Delta u - f(x, y) \frac{\partial u}{\partial x} - g(x, y) \frac{\partial u}{\partial y} - h(x, y)u,$$

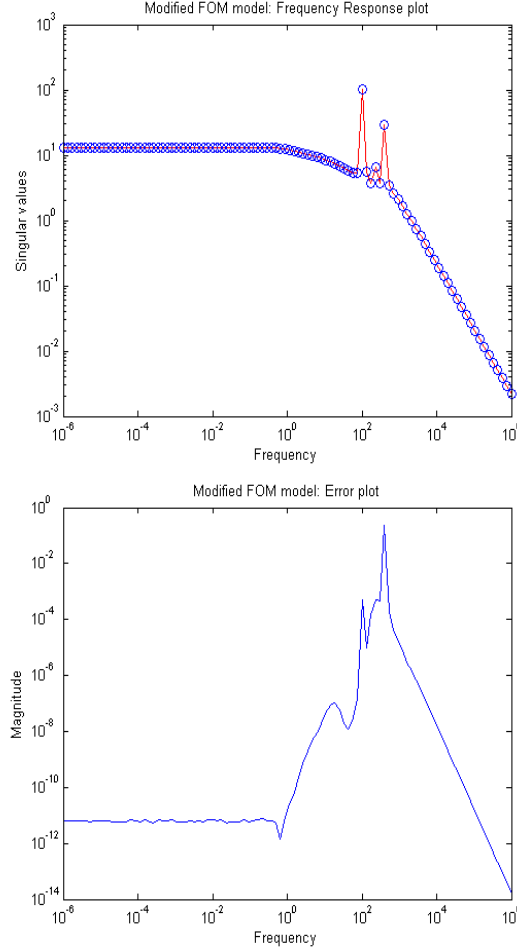


Figure 4.1: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the modified FOM model with $m = 40$.

such that

$$\begin{cases} f(x, y) = \log(x + 2y), \\ g(x, y) = e^{x+y}, \\ h(x, y) = x + y. \end{cases}$$

The matrices B and C were random matrices with entries uniformly distributed in $[0, 1]$. The number of inner grid points in each direction was $n_0 = 100$ and the dimension of A is $n = n_0^2 = 10,000$. For this experiment, we used $p = 6$. The Rail821 model is a first-order system of dimension $n = 821$ and $p = 6$. As can

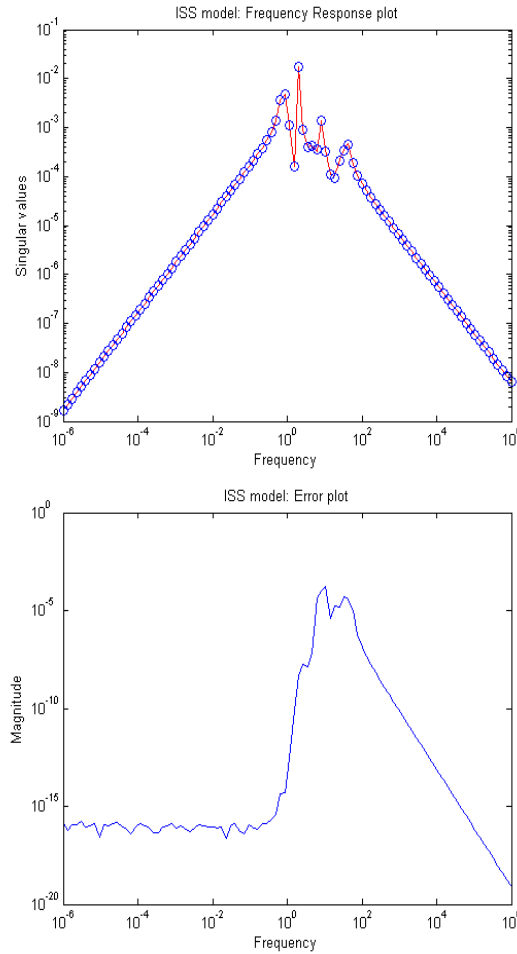


Figure 4.2: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the ISS circuit model with $m = 45$.

be shown from Figure 4.3, the relative error decreases rapidly to zero.

In Table 4.2 we reported the results obtained with different matrix tests. In this table, we listed the exact \mathcal{H}_∞ -error norm, the corresponding iteration (It.) and the obtained cpu-time.

Example 3. For the last example, we compared the exact \mathcal{H}_∞ -error with different approximations using the methods described in last section for choosing the interpolation points (Table 4.1). For this experiment, the matrix test was Rail821.

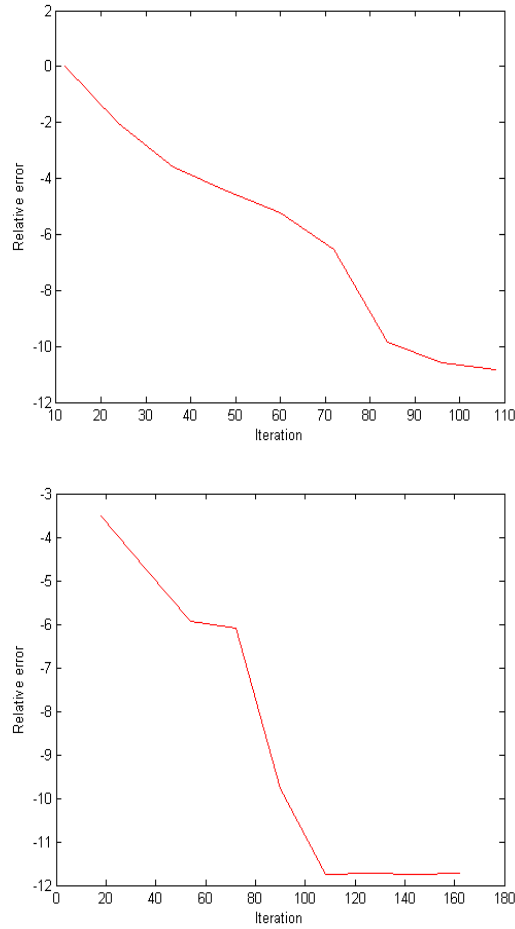


Figure 4.3: Relative error norms. Top: the Rail821 model and bottom: the fdm model.

Table 4.2: The exact \mathcal{H}_∞ -error for different matrix tests.

Matrices	Iteration	$\ F - F_m\ _\infty$
Rail821	16	2.12×10^{-11}
Rail3113	26	5.32×10^{-9}
fdm, $n = 10.000, p = 6$	40	30.66×10^{-9}

As shown from Table 4.3, the results are similar when using the different proposed approaches for selecting the shifts except for the set Σ_5 for which one needs many iterations to get a good approximation. Therefore, we can choose simple sets such

as Σ_1 or Σ_2 to get good interpolation points that could be used in the adaptive modified rational block Lanczos algorithm.

Table 4.3: Results with the Rail821 model.

Error Expressions	It.	$\ F - F_m\ _\infty$
$\Sigma_1 = \{\sigma : \ \tilde{R}_B(\sigma)\ _2 = \ \tilde{R}_B\ _\infty\}$	16	4.5×10^{-10}
$\Sigma_2 = \{\sigma : \ \tilde{R}_C^T(\sigma)\ _2 = \ \tilde{R}_C^T\ _\infty\}$	14	5.2×10^{-11}
$\Sigma_3 = \{\sigma : \ \tilde{H}_m(\sigma)\tilde{R}_B(\sigma)\ _2 = \ \tilde{H}_m\tilde{R}_B\ _\infty\}$	16	1.8×10^{-11}
$\Sigma_4 = \{\sigma : \ \tilde{R}_C^T(\sigma)\tilde{H}_m(\sigma)\ _2 = \ \tilde{R}_C^T\tilde{H}_m\ _\infty\}$	16	2.5×10^{-10}
$\Sigma_5 = \{\sigma : \ \tilde{H}_m(\sigma)\ _2 = \ \tilde{H}_m\ _\infty\}$	50	9.6×10^{-10}
$\Sigma_6 = \{\sigma : \ \tilde{R}_C^T(\sigma)\tilde{H}_m(\sigma)\tilde{R}_B(\sigma)\ _2 = \ \tilde{R}_C^T\tilde{H}_m\tilde{R}_B\ _\infty\}$	16	2.0×10^{-11}
$\Sigma_7 = \{\sigma : \ C_m P_m^{-1}(\sigma)\ _2 \ R_B(\sigma)\ _2 = \ C_m P_m^{-1}\ _\infty \ R_B\ _\infty\}$	16	5.3×10^{-11}

4.6 Conclusion

In this Chapter, we proposed a new adaptive algorithm based on a modified rational block Lanczos process and an adaptive method for choosing the interpolation points for in model order reduction of MIMO first-order stable linear dynamical systems. Moreover, we obtained simple Lanczos equations in rational block case. Numerical experiments show the applicability of the proposed algorithm.

5

A matrix rational Lanczos method for model reduction in large scale first and second order dynamical systems

This Chapter presents numerical and theoretical foundations of global Krylov subspace method for model order reductions. This method is an another extension of the Krylov subspace method for multiple-inputs multiple-outputs (MIMO) systems, as the block Krylov method introduced in last chapters.

If the global Lanczos method and the block Lanczos method are applied to the same matrix pairs (A, B) and (A^T, C^T) the resulting matrices both span the same input and output Krylov subspaces, respectively. The bi-orthogonalization of the bases vectors of Krylov subspaces is the only difference whether constructed by

the block or the global Lanczos methods. In [70] (Chapter 3), it is shown that the moment matching property does only depend on the fact that the columns of the bases generated by the global and the block Lanczos methods span the input and the output Krylov subspaces. It does not depend on the way these bases are computed or whether their columns have a certain additional property. Hence, the moment matching property holds for reduction methods based on the global Lanczos algorithm as well as for reduction methods based on the block Lanczos algorithm.

In this chapter, we describe an adaptive modified rational global Lanczos (**AMRGL**) algorithm for model-order reduction problems using multipoint moment matching based methods. In the first section, we start by proposing a modified rational global Lanczos process and then we show that these proposed algorithm allows to obtain the Lanczos-like equations also for the rational global case. Next, since the major problem of the rational Krylov methods is the selection of some interpolation points, we propose in the second section some adaptive techniques for choosing these shifts. Second-order dynamical systems are also considered in this Chapter and the **AMRGL** algorithm is applied to an equivalent state space model. Finally some numerical examples will be given.

5.1 The modified rational global Lanczos method

In this section, two versions of the global Lanczos procedure are given. we start by introduce the general form of the global Lanczos process and we derive the rational global Lanczos equations related to this algorithm. Next, we modified the first algorithm in such a way the Lanczos-like equations remains valid also in the rational global case.

5.1.1 The rational global Lanczos algorithm

The rational global Lanczos procedure is an algorithm allowing to construct two F-biorthogonal bases $\{V_1, \dots, V_m\}$ and $\{W_1, \dots, W_m\}$ of the rational global Krylov subspaces $\mathcal{K}_m(A, B, \Sigma_m)$ and $\mathcal{K}_m(A^T, C^T, \tilde{\Sigma}_m)$ respectively, where

$$\begin{aligned}\mathcal{K}_m(A, B, \Sigma_m) &= \text{Span}\{(A - \sigma_1 I_n)^{-1}B, \dots, \prod_{i=1}^m (A - \sigma_i I_n)^{-1}B\}, \\ \mathcal{K}_m(A^T, C^T, \tilde{\Sigma}_m) &= \text{Span}\{(A - \tilde{\sigma}_1 I_n)^{-T}C^T, \dots, \prod_{i=1}^m (A - \tilde{\sigma}_i I_n)^{-T}C^T\}.\end{aligned}$$

The sets $\Sigma_m = \{\sigma_1, \dots, \sigma_m\}$ and $\tilde{\Sigma}_m = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_m\}$ contain the interpolation points which will be specified later. The rational global Lanczos algorithm is summarized as follows.

Algorithm 7 The Rational Global Lanczos Algorithm (RGLA)

1. **Input:** $A \in \mathbb{R}^{n \times n}$, $B, C^T \in \mathbb{R}^{n \times p}$, $\Sigma_{m+1} = \{\sigma_1, \dots, \sigma_{m+1}\}$, $\tilde{\Sigma}_{m+1} = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_{m+1}\}$ and a fixed integer m .
 2. **Set** $\tilde{V}_1 = (A - \sigma_1 I_n)^{-1}B$, $\tilde{W}_1 = (A - \tilde{\sigma}_1 I_n)^{-T}C^T$ and construct the initial block vectors V_1 and W_1 such that $\langle W_1, V_1 \rangle_F = 1$; $\mathcal{V}_1 = [V_1]$ and $\mathcal{W}_1 = [W_1]$;
 3. **for** $j = 1, \dots, m$
 4. $\tilde{V}_{j+1} = (A - \sigma_{j+1} I_n)^{-1}V_j$ and $\tilde{W}_{j+1} = (A - \tilde{\sigma}_{j+1} I_n)^{-T}W_j$;
 5. **for** $i = 1, \dots, j$,
 6. $h_{i,j} = \langle W_i, \tilde{V}_{j+1} \rangle_F$ and $g_{i,j} = \langle V_i, \tilde{W}_{j+1} \rangle_F$
 7. $\tilde{V}_{j+1} = \tilde{V}_{j+1} - h_{i,j}V_i$ and $\tilde{W}_{j+1} = \tilde{W}_{j+1} - g_{i,j}W_i$;
 8. **end**
 9. $h_{j+1,j} = \sqrt{|\text{Tr}(\tilde{V}_{j+1}^T \tilde{W}_{j+1})|}$ and $g_{j+1,j} = \frac{\text{Tr}(\tilde{V}_{j+1}^T \tilde{W}_{j+1})}{h_{j+1,j}}$;
 10. $V_{j+1} = \frac{\tilde{V}_{j+1}}{h_{j+1,j}}$ and $W_{j+1} = \frac{\tilde{W}_{j+1}}{g_{j+1,j}}$;
 11. $\mathcal{V}_{j+1} = [\mathcal{V}_j, V_{j+1}]$; $\mathcal{W}_{j+1} = [\mathcal{W}_j, W_{j+1}]$;
 12. **end.**
 13. **Outputs** $\mathcal{V}_{m+1} = [V_1, \dots, V_{m+1}]$, $\mathcal{W}_{m+1} = [W_1, \dots, W_{m+1}]$.
-

Next, we show how to obtain some equations that describe the rational global Lanczos algorithm (Algorithm 7).

5.1.2 The rational global Lanczos equations

Let H_m and G_m be the $m \times m$ upper Hessenberg matrices whose nonzero entries are defined by Algorithm 7, and $\tilde{H}_m, \tilde{G}_m, \tilde{K}_m$ and \tilde{L}_m are the $(m+1) \times m$ matrices defined as

$$\tilde{H}_m = \begin{pmatrix} H_m \\ h_{m+1,m}e_m^T \end{pmatrix}, \quad \tilde{K}_m = \begin{pmatrix} I_m + H_m D_m \\ h_{m+1,m}\sigma_{m+1}e_m^T \end{pmatrix},$$

$$\tilde{G}_m = \begin{pmatrix} G_m \\ g_{m+1,m}e_m^T \end{pmatrix} \quad \text{and} \quad \tilde{L}_m = \begin{pmatrix} I_m + G_m \tilde{D}_m \\ g_{m+1,m}\tilde{\sigma}_{m+1}e_m^T \end{pmatrix},$$

where D_m and \tilde{D}_m are the diagonal matrices $\text{diag}(\sigma_2, \dots, \sigma_{m+1})$ and $\text{diag}(\tilde{\sigma}_2, \dots, \tilde{\sigma}_{m+1})$, respectively, and the sets $\{\sigma_2, \dots, \sigma_{m+1}\}$ and $\{\tilde{\sigma}_2, \dots, \tilde{\sigma}_{m+1}\}$ contain the interpolation points used in Algorithm 7. The following result gives some algebraic properties obtained from the rational global Lanczos algorithm.

Theorem 5.1.1 *Let \mathcal{V}_{m+1} and \mathcal{W}_{m+1} be the F -biorthogonal matrices of $\mathbb{R}^{n \times (m+1)p}$ constructed by Algorithm 7. Then we have the following relations*

$$A\mathcal{V}_{m+1}(\tilde{H}_m \otimes I_p) = \mathcal{V}_{m+1}(\tilde{K}_m \otimes I_p), \quad (5.1)$$

$$A^T\mathcal{W}_{m+1}(\tilde{G}_m \otimes I_p) = \mathcal{W}_{m+1}(\tilde{L}_m \otimes I_p), \quad (5.2)$$

and

$$\mathcal{T}_m = K_m H_m^{-1} + h_{m+1,m}[\sigma_{m+1}\mathcal{W}_m^T \diamond (V_{m+1}E_m^T) - \mathcal{W}_m^T \diamond (AV_{m+1}E_m^T)]H_m^{-1}, \quad (5.3)$$

where $\mathcal{T}_m = \mathcal{W}_m^T \diamond AV_m$.

Proof. From Algorithm 7, we have

$$h_{j+1,j}V_{j+1} = (A - \sigma_{j+1}I_n)^{-1}V_j - \sum_{i=1}^j h_{i,j}V_i \quad \text{for } j = 1, \dots, m. \quad (5.4)$$

Multiplying (5.4) on the left by $(A - \sigma_{j+1}I_n)$, we get

$$A \sum_{i=1}^{j+1} h_{i,j}V_i = V_j + \sigma_{j+1} \sum_{i=1}^{j+1} h_{i,j}V_i \quad \text{for } j = 1, \dots, m$$

which gives the following relation

$$\begin{aligned} AV_{m+1}(\tilde{H}_m \otimes I_p) &= \mathcal{V}_m(I_m + H_m D_m) \otimes I_p + \sigma_{m+1} h_{m+1,m} V_{m+1} (e_m^T \otimes I_p), \\ &= \mathcal{V}_{m+1}(\tilde{K}_m \otimes I_p), \end{aligned}$$

and then the equation (5.1) is satisfied. In a similar way, the relation (5.2) can be shown.

To prove the relation (5.3), we proceed as follows. We use the relation (5.1) to have

$$AV_m(H_m \otimes I_p) = \mathcal{V}_m(K_m \otimes I_p) + \sigma_{m+1} h_{m+1,m} V_{m+1} E_m^T - AV_{m+1} h_{m+1,m} E_m^T,$$

Using the F -biorthogonality between \mathcal{V}_m and \mathcal{W}_m and the properties of the kronecker product gives

$$\mathcal{T}_m = \mathcal{W}_m^T \diamond AV_m = K_m H_m^{-1} + h_{m+1,m} [\sigma_{m+1} \mathcal{W}_m^T \diamond (V_{m+1} E_m^T) - \mathcal{W}_m^T \diamond (AV_{m+1} E_m^T)] H_m^{-1}.$$

Let us now write the transfer function of the dynamical system (2.1) as $F(s) = CX$ where $X \in \mathbb{R}^{n \times p}$ is the solution of the following matrix linear system

$$(sI_n - A)X = B, \quad (5.5)$$

assuming that the matrix $sI_n - A$ is non-singular. In the context of solving the shifted system (5.5), an approximate solution $X_m = \mathcal{V}_m Y_m$ can be determined by imposing the Petrov-Galerkin condition

$$R_B(s) \perp_F \text{Span}\{W_1, \dots, W_m\},$$

where $R_B(s) = B - (sI_n - A)X_m$ is the residual associated to the approximation X_m . Therefore we obtain

$$X_m = \mathcal{V}_m [((sI_m - \mathcal{T}_m)^{-1} (\mathcal{W}_m^T \diamond B)) \otimes I_p].$$

Using the properties of the kronecker product, X_m can be expressed as

$$X_m = \mathcal{V}_m (sI_{mp} - (\mathcal{T}_m \otimes I_p))^{-1} ((\mathcal{W}_m^T \diamond B) \otimes I_p). \quad (5.6)$$

Then the reduced-order transfer function can be written as

$$\begin{aligned} F_m(s) &= CX_m = C\mathcal{V}_m (sI_{mp} - (\mathcal{T}_m \otimes I_p))^{-1} ((\mathcal{W}_m^T \diamond B) \otimes I_p) \\ &= C_m (sI_{mp} - A_m)^{-1} B_m, \end{aligned}$$

where

$$A_m = (\mathcal{T}_m \otimes I_p), \quad B_m = ((\mathcal{W}_m^T \diamond B) \otimes I_p) \quad \text{and} \quad C_m = C\mathcal{V}_m. \quad (5.7)$$

One possibility of choosing the two sets of shifts could be derived by using the

following theorem presented in [71] for SISO systems, and extended to MIMO case in [57].

Theorem 5.1.2 *Let $\Sigma_m = \{\sigma_1, \dots, \sigma_m\}$ and $\tilde{\Sigma}_m = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_m\}$ be two sets of interpolation points such that the matrices $A - \sigma_i I_n$ and $A - \tilde{\sigma}_i I_n$ are invertible for $i = 1, \dots, m$. If*

$$\text{Span}\{(A - \sigma_1 I_n)^{-1} B, \dots, (A - \sigma_m I_n)^{-1} B\} \subseteq \text{Range}(\mathcal{V}_m),$$

and

$$\text{Span}\{(A - \tilde{\sigma}_1 I_n)^{-T} C^T, \dots, (A - \tilde{\sigma}_m I_n)^{-T} C^T\} \subseteq \text{Range}(\mathcal{W}_m)$$

with $\mathcal{W}_m^T \diamond \mathcal{V}_m = I_m$, then

- If $\sigma_i = \tilde{\sigma}_i$, the approximate transfer function F_m defined in (5.7) interpolates the original transfer function $F(s)$ and its first derivative at the selected points $\sigma_i, i = 1, \dots, m$, i.e.,

$$F_m(\sigma_i) = F(\sigma_i) \text{ and } F'_m(\sigma_i) = F'(\sigma_i), \quad i = 1, \dots, m.$$

- If $\sigma_i \neq \tilde{\sigma}_i$, the reduced order transfer function F_m interpolates the values of the original transfer function $F(s)$ at the points $\sigma_i, i = 1, \dots, m$ and $\tilde{\sigma}_i, i = 1, \dots, m$, i.e.,

$$F_m(\sigma_i) = F(\sigma_i) \text{ and } F_m(\tilde{\sigma}_i) = F(\tilde{\sigma}_i), \quad i = 1, \dots, m.$$

Later, we will propose two adaptive techniques for choosing the two sets of shifts.

5.1.3 Lanczos-like equations in the rational Global case

The relations derived in Theorem 5.1.1 are known as the rational global Lanczos equations that relate the F-biorthogonal matrices $\mathcal{V}_{m+1}, \mathcal{W}_{m+1}$ and the matrix A. We can also derive simple Lanczos equations in the rational case. For this reason, we modified the rational global Lanczos algorithm by allowing some interpolation points to be equal to infinity. Such a result is given for the block case in last chapter [14]. The modified rational global Lanczos algorithm is summarized as follows.

Algorithm 8 The Modified Rational Global Lanczos Algorithm (MRGLA)

1. **Input:** $A \in \mathbb{R}^{n \times n}, B, C^T \in \mathbb{R}^{n \times p}, \Sigma_{m+1} = \{\sigma_1, \dots, \sigma_{m+1}\}, \tilde{\Sigma}_{m+1} = \{\tilde{\sigma}_1, \dots, \tilde{\sigma}_{m+1}\}$ and a fixed integer m .
 2. **Set** $\tilde{V}_1 = (A - \sigma_1 I_n)^{-1} B, \tilde{W}_1 = (A - \tilde{\sigma}_1 I_n)^{-T} C^T$ and construct the initial block vectors V_1 and W_1 such that $\langle W_1, V_1 \rangle_F = 1; \mathcal{V}_1 = [V_1]$ and $\mathcal{W}_1 = [W_1]$;
 3. **for** $j = 1, \dots, m$
 4. **if** $\{\sigma_{j+1} = \infty\}; \tilde{V}_{j+1} = AV_j$ **else** $\tilde{V}_{j+1} = (A - \sigma_{j+1} I_n)^{-1} V_j$; **end**
 5. **if** $\{\tilde{\sigma}_{j+1} = \infty\}; \tilde{W}_{j+1} = A^T W_j$ **else** $\tilde{W}_{j+1} = (A - \tilde{\sigma}_{j+1} I_n)^{-T} W_j$ **end**
 6. **for** $i = 1, \dots, j$,
 7. $h_{i,j} = \langle W_i, V_{j+1} \rangle_F$ and $g_{i,j} = \langle V_i, \tilde{W}_{j+1} \rangle_F$
 8. $\tilde{V}_{j+1} = \tilde{V}_{j+1} - h_{i,j} V_i$ and $\tilde{W}_{j+1} = \tilde{W}_{j+1} - g_{i,j} W_i$;
 9. **end**
 10. $h_{j+1,j} = \sqrt{|\text{Tr}(\tilde{V}_{j+1}^T \tilde{W}_{j+1})|}$ and $g_{j+1,j} = \frac{\text{Tr}(\tilde{V}_{j+1}^T \tilde{W}_{j+1})}{h_{j+1,j}}$;
 11. $V_{j+1} = \frac{\tilde{V}_{j+1}}{h_{j+1,j}}$ and $W_{j+1} = \frac{\tilde{W}_{j+1}}{g_{j+1,j}}$;
 12. $\mathcal{V}_{j+1} = [\mathcal{V}_j, V_{j+1}]; \mathcal{W}_{j+1} = [\mathcal{W}_j, W_{j+1}];$
 13. **end.**
 14. **Outputs** $\mathcal{V}_{m+1} = [V_1, \dots, V_{m+1}], \mathcal{W}_{m+1} = [W_1, \dots, W_{m+1}].$
-

We notice that in our setting, we assume that we are not given the sequences of shifts $\sigma_1, \sigma_2, \dots, \sigma_{m+1}$ and $\tilde{\sigma}_1, \tilde{\sigma}_2, \dots, \tilde{\sigma}_{m+1}$ and then we need to include the procedure to automatically generate this sequences during the iterations of the process. This adaptive procedure will be defined in the next section. Using the modified rational global Lanczos algorithm for $\sigma_{m+1} = \tilde{\sigma}_{m+1} = \infty$, we can obtain Lanczos-like equations such those obtained in the standard Lanczos case. The simple equations allow us to have simple residual error expressions for the rational case or to derive error bounds.

Theorem 5.1.3 *Let \mathcal{V}_{m+1} and \mathcal{W}_{m+1} be the matrices generated by the modified rational global Lanczos algorithm (Algorithm 8) for the extra interpolation points*

at $\sigma_{m+1} = \tilde{\sigma}_{m+1} = \infty$. Then we have the following Lanczos-like relations

$$A\mathcal{V}_m = \mathcal{V}_m(\mathcal{T}_m \otimes I_p) + V_{m+1}(P_{m+1} \otimes I_p), \quad (5.8)$$

$$A^T\mathcal{W}_m = \mathcal{W}_m(\mathcal{T}_m^T \otimes I_p) + W_{m+1}(Q_{m+1} \otimes I_p), \quad (5.9)$$

$$B = \mathcal{V}_m B_m + V_{m+1}(b_m \otimes I_p), \quad (5.10)$$

$$C^T = \mathcal{W}_m C_m^T + W_{m+1}(c_m^T \otimes I_p), \quad (5.11)$$

where $\mathcal{T}_m = \mathcal{W}_m^T \diamond A\mathcal{V}_m$, $P_{m+1} = W_{m+1}^T \diamond A\mathcal{V}_m$, $Q_{m+1} = V_{m+1}^T \diamond A^T\mathcal{W}_m$, $B_m = ((\mathcal{W}_m^T \diamond B) \otimes I_p)$, $C_m = ((C \diamond \mathcal{V}_m) \otimes I_p)$, $b_m = W_{m+1}^T \diamond B$ and $c_m = C \diamond V_{m+1}$.

Proof. According to Algorithm 8, we have

$$\text{Span}\{V_1, \dots, V_m, AV_m\} = \text{Span}\{\mathcal{V}_{m+1}\}$$

with $\mathcal{V}_{m+1} = [V_1, \dots, V_{m+1}]$ and $\mathcal{W}_{m+1}^T \diamond \mathcal{V}_{m+1} = I_{m+1}$, and we need to prove that

$$\text{Span}\{AV_1, \dots, AV_m\} \subset \text{Span}\{\mathcal{V}_{m+1}\}.$$

In fact, after $m - 1$ iterations of Algorithm 8 and assuming that H_{m-1} is non-singular, the result of Theorem 5.1.1 gives

$$A\mathcal{V}_m(\tilde{H}_{m-1} \otimes I_p) = \mathcal{V}_m(\tilde{K}_{m-1} \otimes I_p).$$

therefore

$$\begin{aligned} A\mathcal{V}_{m-1} &= \mathcal{V}_{m-1}(K_{m-1} \otimes I_p)(H_{m-1}^{-1} \otimes I_p) + \sigma_m h_{m,m-1} V_m E_{m-1}^T (H_{m-1}^{-1} \otimes I_p) \\ &\quad - h_{m,m-1} AV_m E_{m-1}^T (H_{m-1}^{-1} \otimes I_p), \end{aligned}$$

which gives $\text{Span}\{A\mathcal{V}_{m-1}\} \subset \text{Span}\{\mathcal{V}_{m+1}\}$.

Now, as $A\mathcal{V}_m = [A\mathcal{V}_{m-1}, AV_m]$ and $AV_m \in \text{Span}\{\mathcal{V}_{m+1}\}$, we have

$$\text{Span}\{A\mathcal{V}_m\} \subset \text{Span}\{\mathcal{V}_{m+1}\}.$$

Then, there exists a matrix $Y \in \mathbb{R}^{(m+1) \times m}$ such that

$$A\mathcal{V}_m = \mathcal{V}_{m+1}(Y \otimes I_p).$$

Setting $Y = \begin{pmatrix} \tilde{\mathcal{T}}_m \\ P_{m+1} \end{pmatrix}$, we obtain

$$A\mathcal{V}_m = \mathcal{V}_{m+1} \left(\begin{pmatrix} \tilde{\mathcal{T}}_m \\ P_{m+1} \end{pmatrix} \otimes I_p \right).$$

Using the F-biorthogonality between \mathcal{V}_{m+1} and \mathcal{W}_{m+1} gives $\tilde{\mathcal{T}}_m = \mathcal{T}_m = \mathcal{W}_m^T \diamond A\mathcal{V}_m$ and $P_{m+1} = W_{m+1}^T \diamond A\mathcal{V}_m$. Therefore we get

$$A\mathcal{V}_m = \mathcal{V}_m(\mathcal{T}_m \otimes I_p) + V_{m+1}(P_{m+1} \otimes I_p).$$

Similarly, to prove (5.10) we proceed as follows: Since $B \in \text{Span}\{\mathcal{V}_1\} \subset \text{Span}\{\mathcal{V}_{m+1}\}$,

there exists a matrix $Z \in \mathbb{R}^{(m+1) \times 1}$ such that

$$B = \mathcal{V}_{m+1}(Z \otimes I_p). \quad (5.12)$$

Setting $Z = \begin{pmatrix} \tilde{B}_m \\ b_m \end{pmatrix}$, and multiplying (5.12) by \mathcal{W}_{m+1}^T from the left we get

$$\mathcal{W}_{m+1}^T \diamond B = \begin{pmatrix} \tilde{B}_m \\ b_m \end{pmatrix}. \quad (5.13)$$

Then $\tilde{B}_m = \mathcal{W}_m^T \diamond B$, $b_m = \mathcal{W}_{m+1}^T \diamond B$ and consequently we obtain the following relation

$$\begin{aligned} B &= \mathcal{V}_m(\tilde{B}_m \otimes I_p) + V_{m+1}(b_m \otimes I_p) \\ &= \mathcal{V}_m B_m + V_{m+1}(b_m \otimes I_p). \end{aligned} \quad (5.14)$$

In a similar way, we can show the relations (5.9) and (5.11).

5.2 An adaptive modified rational global Lanczos algorithm

5.2.1 Adaptive choice of interpolation points

As in the previous chapters, rational Krylov methods always require a good selection of interpolation points for a good convergence of the reduced order model process. This subsection deals with this problem and propose some adaptive techniques to chose the set of shifts.

5.2.1.1 First approach

The first approach is a generalization to the one used in last chapter for the global case. The following result is the key ingredient of this method.

Proposition 5.2.1 *Let $\Sigma_m = \{\sigma_1, \dots, \sigma_m\}$ denote a given set of interpolation*

points and let $\mathcal{V}_m, \mathcal{W}_m \in \mathbb{R}^{n \times mp}$ be the F -biorthogonal matrices computed by the modified rational global Lanczos algorithm. The following relation holds

$$\|F(s) - F_m(s)\|_2 \leq \|CP^{-1}(s)\|_2 \|R_B(s)\|_2, \quad (5.15)$$

where $R_B(s) = B - P(s)\mathcal{V}_m P_m^{-1}(s)B_m$, $P(s) = sI - A$ and $B_m = ((\mathcal{W}_m^T \diamond B) \otimes I_p)$.

Proof. From the expression of transfer functions $F(s)$ and $F_m(s)$, we have

$$\begin{aligned} \|F(s) - F_m(s)\|_2 &= \|C(sI_n - A)^{-1}B - C_m(sI_{mp} - A_m)^{-1}B_m\|_2 \\ &= \|C(sI_n - A)^{-1}(B - (sI_n - A)\mathcal{V}_m(sI_{mp} - A_m)^{-1}B_m)\|_2 \\ &= \|CP(s)^{-1}(B - P(s)\mathcal{V}_m P_m(s)^{-1}B_m)\|_2 \\ &\leq \|CP(s)^{-1}\|_2 \|R_B(s)\|_2. \end{aligned}$$

The preceding result suggests us to approximate the upper bound by using the reduced order matrix triplet (A_m, B_m, C_m) and then $\|CP(s)^{-1}\|$ could be approximated by $\|C_m P_m(s)^{-1}\|$, where $P_m(s) = sI_{mp} - A_m$ and $A_m = (\mathcal{T}_m \otimes I_p)$. Using the above approximation, the next shift σ_{m+1} can be selected as

$$\sigma_{m+1} = \arg \max_{s \in S} \|C_m P_m(s)^{-1}\|_2 \|R_B(s)\|_2. \quad (5.16)$$

Since the expression of $R_B(s)$ contains terms related to the dimension n of the space, the computation of the next shift σ_{m+1} needs more computation times and arithmetic operations for large problems. Then we use the result of Theorem 5.1.3 to simplify the residual expression. We have

$$\begin{aligned} R_B(s) &= B - (sI_n - A)\mathcal{V}_m(sI_{mp} - A_m)^{-1}B_m \\ &= B - (s\mathcal{V}_m - \mathcal{V}_m(\mathcal{T}_m \otimes I_p) - V_{m+1}(P_{m+1} \otimes I_p))(sI_{mp} - A_m)^{-1}B_m, \\ &= B - \mathcal{V}_m(sI_{mp} - A_m)(sI_{mp} - A_m)^{-1}B_m \\ &\quad + V_{m+1}(P_{m+1} \otimes I_p)(sI_{mp} - A_m)^{-1}B_m, \\ &= V_{m+1}((P_{m+1} \otimes I_p)(sI_{mp} - A_m)^{-1}B_m + (b_m \otimes I_p)), \\ &= V_{m+1} \tilde{R}_B(s) \end{aligned}$$

where $\tilde{R}_B(s) = (P_{m+1} \otimes I_p)(sI_{mp} - A_m)^{-1}B_m + (b_m \otimes I_p)$ is the frequency-dependent term of the residual $R_B(s)$. Therefore, we can choose the next interpolation points σ_{m+1} and $\tilde{\sigma}_{m+1}$ by using just the frequency-dependent term of the residual $R_B(s)$

instead of using $R_B(s)$, such that

$$\sigma_{m+1} = \arg \max_{s \in S} \|C_m P_m(s)^{-1}\|_2 \|\tilde{R}_B(s)\|_2 \quad \text{and} \quad \tilde{\sigma}_{m+1} = \arg \max_{s \in S} \|\tilde{R}_B(s)\|_2. \quad (5.17)$$

5.2.1.2 Second approach

In [43] Druskin and al. proposed an adaptive approach for computing real and complex interpolation points. This method is a generalization of the one proposed in [40] for the symmetric case. A similar approach is used here for the modified rational global Lanczos algorithm in the context of real shifts. First, notice that any vector $u \in \mathcal{K}_m(A, B, \Sigma_m)$ can be written as

$$u = p_{m-1}(A)q_m(A)^{-1}B,$$

where p_{m-1} is a polynomial of degree at most $m - 1$, while q_m is a polynomial of degree m , whose roots are the components of $\Sigma_m \in \mathbb{R}^m$. Let X_m be the approximate solution defined in (5.6) and $f_{\theta_1, \dots, \theta_m, \sigma_1, \dots, \sigma_m}(\theta, s)$ is the so-called skeleton approximation introduced in [139], which is an $[m - 1/m]$ rational function of each variable, interpolating $(\theta + s)^{-1}$ at $\theta = \theta_i, s = \sigma_i, i = 1, \dots, m$. Using the same techniques in [94], we can show that

$$X_m = f_{\lambda_1, \dots, \lambda_m, \sigma_1, \dots, \sigma_m}(A, s)B, \quad (5.18)$$

where $\lambda_j, j = 1, \dots, m$ are the eigenvalues of $\mathcal{T}_m = \mathcal{W}_m^T \diamond A \mathcal{V}_m$. Since the relative error of the skeleton approximation is given by

$$\delta(\lambda, s) = \left(\frac{1}{\lambda + s} - f_{\lambda_1, \dots, \lambda_m, \sigma_1, \dots, \sigma_m}(\lambda, s) \right) / \frac{1}{\lambda + s} = \frac{r_m(\lambda)}{r_m(s)},$$

where $r_m(z) = \prod_{j=1}^m \frac{z - \lambda_j}{z - \sigma_j}$; see [40], and using the relation (5.18), the residual $R_B(s)$ can be expressed as

$$R_B(s) = \delta(A, s)B = B - (sI_n - A)X_m = \frac{r_m(A)B}{r_m(s)}. \quad (5.19)$$

From Proposition 2 in [35], the characteristic polynomial of \mathcal{T}_m minimizes $\|p(A)V_1\|_F$ over all monic polynomial of degree m , so that the numerator in (5.19) satisfies

$$\|r_m(A)B\|_F = \min_{\theta_1, \dots, \theta_m} \left\| \prod_{j=1}^m (\theta_j I_n - A)(\sigma_j I_n - A)^{-1} B \right\|_F. \quad (5.20)$$

With this result, the next interpolation point σ_{m+1} is selected as

$$\sigma_{m+1} = \arg \left(\max_{s \in S} \frac{1}{|r_m(s)|} \right), \quad (5.21)$$

where the choice of the set S will be discussed later. For the other next interpolation point, we can consider $\tilde{\sigma}_{m+1} = \sigma_{m+1}$. The algorithm for constructing the new shift σ_{m+1} is given as follows.

Algorithm 9 The second procedure for selecting the shifts

• **Input :** $\{\lambda_j\}_{j=1}^m, \{\sigma_j\}_{j=1}^m$ and the set $\{\eta_1, \dots, \eta_l\}$;

1. For $j = 1, \dots, l - 1$
 2. $\mu_j = \arg \max_{\mu \in [\eta_j, \eta_{j+1}]} \frac{1}{|r_m(\mu)|}, r_m(z) = \prod_{i=1}^m \frac{z - \lambda_i}{z - \sigma_i};$
 3. end
 4. $\sigma_{m+1} = \arg \max_{j=1, \dots, l-1} \frac{1}{|r_m(\mu_j)|}.$
-

$\{\lambda_i\}_{i=1}^m$ are the eigenvalues of the matrix \mathcal{T}_m at the iteration m and $\{\sigma_i\}_{i=1}^m$ are the previously chosen interpolation points. The set $\{\eta_1, \dots, \eta_l\}$ contains two given initial values $\sigma_0^{(1)}, \sigma_0^{(2)}$ and the previously chosen shifts (increasingly ordered) such that $\sigma_0^{(1)}$ is used as the first interpolation point.

5.2.2 The adaptive modified rational global Lanczos algorithm

Combining the modified rational global Lanczos algorithm (Algorithm 7) and one of the adaptive approaches explained above to construct the interpolation points gives the adaptive modified rational global Lanczos (AMRGL) algorithm for reduced the order of large scale dynamical systems.

5.3 Model Reduction of Second order Systems

Second order systems are sets of second order differential equations. In some fields like electrical circuits and mechanical systems, modeling (for instance by FEM) leads to a large number of second order differential equations [9, 117, 119, 130]. It is then advisable to construct a reduced order model that approximates the behavior of the original system while preserving its second-order structure [12, 100, 127, 135].

An extension of balancing and truncation methods to reduce the order of second order system was first introduced by Meyer and Srinivasan in [105]. However, it is not recommended to use this approach for the reduction of large scale systems for numerical reasons. To reduce the order of large scale second order systems, it is required to implement more reliable and faster algorithms and preferably iterative procedures. The first idea is of course extending the numerically efficient algorithms like Arnoldi and Lanczos, which are used in Krylov subspace methods as well-accepted approaches for the reduction of large scale state space models.

One of the oldest extensions of moment matching method for second order model was proposed by Su and Craig [135] which is equivalent to a recent work in [12] where the reduced system is found in a way different from [135]. In both papers, the Krylov subspaces were used and the structure of the original system is preserved.

Recently, in several works, it is tried to extend the Krylov subspace approach to reducing the order of second order systems. In [52, 53, 98, 141], it is proposed to reduce the equivalent state space system by applying a projection such that the structure of the state space matrices does not change and an algorithm is given to find the desired projection matrices.

5.3.1 Model order reduction techniques for second order systems

We consider the second order dynamical system expressed as follows

$$\begin{cases} M\ddot{q}(t) + D\dot{q}(t) + Kq(t) = Bu(t), \\ y(t) = Cq(t), \end{cases} \quad (5.22)$$

where $q(t) \in \mathbb{R}^n$ is a vector of the state variables, $u(t), y(t) \in \mathbb{R}^p$ are the input force and the output measurement functions, respectively. The matrices M, D and $K \in \mathbb{R}^{n \times n}$ are known as the mass, damping and stiffness matrices, respectively. $B, C^T \in \mathbb{R}^{n \times p}$ are the input distribution and the output measurement matrices, respectively. If $D = 0$, the system (5.22) is said to be undamped. Second order systems arise naturally in many areas of engineering; see for example [116, 118, 142].

The mass matrix is assumed to be invertible and we write

$$K_M = M^{-1}K, \quad D_M = M^{-1}D, \quad B_M = M^{-1}B, \quad C_M = C. \quad (5.23)$$

To simplify notations, we still denote K, D, B and C instead of K_M, D_M, B_M and C_M , respectively. Then, the transfer function associated with the system (5.22) by direct Laplace transform is

$$F(s) = C(s^2I_n + sD + K)^{-1}B. \quad (5.24)$$

The original system is too large to allow the efficient solution of various control or simulation tasks. In order to address this problem, many methods have been developed to produce a reduced-order system of size $r \ll n$ such that the essential properties of the original system are preserved. Then we need to construct a reduced model having the form

$$\begin{cases} \ddot{\hat{q}}(t) + \hat{D}\dot{\hat{q}}(t) + \hat{K}\hat{q}(t) = \hat{B}u(t), \\ \hat{y}(t) = \hat{C}\hat{q}(t), \end{cases} \quad (5.25)$$

where $\hat{q}(t) \in \mathbb{R}^r, \hat{D}, \hat{K} \in \mathbb{R}^{r \times r}, \hat{B}, \hat{C}^T \in \mathbb{R}^{r \times p}$, such that its transfer function is close to the original transfer function. The associated low-order transfer function is denoted by

$$\hat{F}(s) = \hat{C}(s^2I_r + s\hat{D} + \hat{K})^{-1}\hat{B}.$$

Second order systems can be considered as a particular class of linear systems by

rewriting the system (5.22) as follows

$$\begin{cases} \begin{bmatrix} I_n & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} \dot{q}(t) \\ \ddot{q}(t) \end{bmatrix} = \begin{bmatrix} 0 & I_n \\ -K & -D \end{bmatrix} \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ B \end{bmatrix} u(t), \\ y(t) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix}. \end{cases} \quad (5.26)$$

Defining

$$x(t) = \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix}, \quad \mathcal{A} = \begin{bmatrix} 0 & I_n \\ -K & -D \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} 0 \\ B \end{bmatrix}, \quad \mathcal{C} = \begin{bmatrix} C & 0 \end{bmatrix}, \quad (5.27)$$

the system (5.26) and the transfer function in (5.24) can be written as

$$\begin{cases} \dot{x}(t) = \mathcal{A}x(t) + \mathcal{B}u(t), \\ y(t) = \mathcal{C}x(t), \end{cases} \quad (5.28)$$

and

$$\mathcal{F}(s) = \mathcal{C}(sI_{2n} - \mathcal{A})^{-1}\mathcal{B}.$$

If the dimension of the state $q(t)$ of the original second order system (5.22) is equal to n , then the order of its corresponding linearized state space realization (5.27) (called also the McMillan degree of $F(s)$) is equal to $N = 2n$.

We notice that, $\mathcal{F}(s) = F(s)$. In fact, setting

$$X = (sI_{2n} - \mathcal{A})^{-1}\mathcal{B} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix},$$

it follows that $\mathcal{F}(s) = \mathcal{C}X$ where X satisfies $(sI_{2n} - \mathcal{A})X = \mathcal{B}$. Therefore, replacing \mathcal{A} , \mathcal{B} and \mathcal{C} by the expressions in (5.27) gives

$$(s^2I_n + sD + K)X_1 = B, \quad \text{and} \quad \mathcal{F}(s) = CX_1.$$

Hence,

$$\mathcal{F}(s) = F(s) = C(s^2I_n + sD + K)^{-1}B.$$

We can produce a reduced model for the second order system (5.22) by applying classical linear model reduction techniques to $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ in (5.27). Unfortunately, there is no guarantee that the resulting reduced system would be a second-order system, which requires the development of second-order structure preserving model reduction techniques, see [16, 27, 28, 141] and the references therein.

5.3.1.1 Second Order Structure-Preserving Krylov Techniques

As the Krylov subspace-type methods do not preserve second order structure when applied to the linear system (5.27), the authors in [27, 141] proposed to modify it to satisfy the interpolation condition and produce a second order reduced system. The results are given for the SISO systems and extended here to the MIMO case. The following result, which is not difficult to prove, gives a simple sufficient condition for obtaining a second order reduced system.

Lemma 5.3.1 *Let $(\mathcal{A}, \mathcal{B}, \mathcal{C})$ be the state space realization defined in (5.27). If one projects such a state space realization with $2n \times 2mp$ bloc diagonal matrices*

$$\mathcal{V}_m = \begin{bmatrix} \mathcal{V}_m^1 & 0 \\ 0 & \mathcal{V}_m^2 \end{bmatrix}, \quad \mathcal{W}_m = \begin{bmatrix} \mathcal{W}_m^1 & 0 \\ 0 & \mathcal{W}_m^2 \end{bmatrix}, \quad \mathcal{W}_m^T \diamond \mathcal{V}_m = I_{2m},$$

where $\mathcal{V}_m^1, \mathcal{V}_m^2, \mathcal{W}_m^1$ and $\mathcal{W}_m^2 \in \mathbb{R}^{n \times mp}$, then the reduced transfer function

$$\mathcal{F}_m(s) = \mathcal{C}\mathcal{V}_m[sI_{2mp} - (\mathcal{W}_m^T \diamond \mathcal{A}\mathcal{V}_m) \otimes I_p]^{-1}[(\mathcal{W}_m^T \diamond \mathcal{B}) \otimes I_p]$$

is a second order transfer function, provided that the matrix $((\mathcal{W}_m^1)^T \diamond \mathcal{V}_m^2) \otimes I_p$ is non-singular.

Using the above result and the result of Theorem 4.1, the following theorem can be proved.

Theorem 5.3.1 *Let $\mathcal{F}(s) = C(s^2I_n + sD + K)^{-1}B = C(sI_{2n} - \mathcal{A})^{-1}\mathcal{B}$, with*

$$\mathcal{A} = \begin{bmatrix} 0 & I_n \\ -K & -D \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} 0 \\ B \end{bmatrix}, \quad \mathcal{C} = \begin{bmatrix} C & 0 \end{bmatrix},$$

be a second order transfer function of McMillan degree $2n$. Let $\mathbb{V}_m, \mathbb{W}_m \in \mathbb{R}^{2n \times mp}$ be defined as

$$\mathbb{V}_m = \begin{bmatrix} \mathbb{V}_m^1 \\ \mathbb{V}_m^2 \end{bmatrix}, \quad \mathbb{W}_m = \begin{bmatrix} \mathbb{W}_m^1 \\ \mathbb{W}_m^2 \end{bmatrix}$$

with $\mathbb{V}_m^1, \mathbb{V}_m^2, \mathbb{W}_m^1, \mathbb{W}_m^2 \in \mathbb{R}^{n \times mp}$ and $(\mathbb{W}_m^1)^T \diamond \mathbb{V}_m^1 = (\mathbb{W}_m^2)^T \diamond \mathbb{V}_m^2 = I_m$. Let us define the $2n \times 2mp$ projecting matrices

$$\mathcal{V}_m = \begin{bmatrix} \mathbb{V}_m^1 & 0 \\ 0 & \mathbb{V}_m^2 \end{bmatrix}, \quad \text{and} \quad \mathcal{W}_m = \begin{bmatrix} \mathbb{W}_m^1 & 0 \\ 0 & \mathbb{W}_m^2 \end{bmatrix}.$$

Define the second order transfer function of order m (and of McMillan degree $2m$) by

$$\begin{aligned}\mathcal{F}_m(s) &= \mathcal{C}\mathcal{V}_m(sI_{2mp} - (\mathcal{W}_m^T \diamond \mathcal{A}\mathcal{V}_m) \otimes I_p)^{-1}((\mathcal{W}_m^T \diamond \mathcal{B}) \otimes I_p) \\ &= \mathcal{C}_m(sI_{2mp} - \mathcal{A}_m)^{-1}\mathcal{B}_m.\end{aligned}\tag{5.29}$$

Assume that

$$\text{Span}\{(\mathcal{A} - \sigma_1 I)^{-1}\mathcal{B}, \dots, (\mathcal{A} - \sigma_m I)^{-1}\mathcal{B}\} \subseteq \text{Range}(\mathbb{V}_m)$$

and

$$\text{Span}\{(\mathcal{A} - \tilde{\sigma}_1 I)^{-T}\mathcal{C}^T, \dots, (\mathcal{A} - \tilde{\sigma}_m I)^{-T}\mathcal{C}^T\} \subseteq \text{Range}(\mathbb{W}_m)$$

where the interpolation points σ_k and $\tilde{\sigma}_k$ are chosen such that the matrices $\mathcal{A} - \sigma_k I$ and $\mathcal{A} - \tilde{\sigma}_k I$ are invertible $\forall k \in \{1, \dots, m\}$.

Then, if the matrix $((\mathbb{W}_m^1)^T \diamond \mathbb{V}_m^2) \otimes I_p$ is non-singular, the reduced order transfer function $\mathcal{F}_m(s) = \mathcal{C}_m(sI_{2mp} - \mathcal{A}_m)^{-1}\mathcal{B}_m$ interpolates the values of the original transfer function $\mathcal{F}(\cdot)$ at the interpolation points $\{\sigma_i\}_{i=1}^m$ and $\{\tilde{\sigma}_i\}_{i=1}^m$. If $\sigma_i = \tilde{\sigma}_i$, the reduced order model interpolates the original transfer function $\mathcal{F}(\cdot)$ and its first derivative at the selected points $\{\sigma_i\}_{i=1}^m$.

Proof. The second order structure of $\mathcal{F}_m(s)$ follows from Lemma 5.3.1. It is clear that

$$\text{Range}(\mathbb{V}_m) \subset \text{Range}(\mathcal{V}_m) \quad \text{and} \quad \text{Range}(\mathbb{W}_m) \subset \text{Range}(\mathcal{W}_m).$$

Then using the results of Theorem 5.1.2, the interpolation conditions are satisfied.

5.3.1.2 Proportionally Damped Systems

The idea of this subsection is to reduce second order models by applying a projection directly in the second order system. To this end, the definition of the standard Krylov subspace is extended to the so called Second Order Krylov Subspace which was first introduced in [128] to find the projection matrices and matching the moments, and more investigated and generalized in [101].

Next, we define the second order Krylov subspaces, see [9], and the references therein.

Definition 5.3.1 *The Second Order Krylov Subspace is defined as,*

$$\tilde{\mathcal{K}}_m(P_1, P_2, Q) = \text{span}\{X_0, X_1, \dots, X_{m-1}\}$$

where where

$$\begin{aligned} X_0 &= Q \\ X_1 &= P_1 X_0 \\ X_i &= P_1 X_{i-1} + P_2 X_{i-2}, \quad i = 2, 3, \dots, m-1 \end{aligned} \tag{5.30}$$

and $P_1, P_2 \in \mathbb{R}^{n \times n}$, $Q \in \mathbb{R}^{n \times p}$ are constant matrices. The columns of Q are called the starting vectors and the matrices X_i are called basic blocks.

In this paragraph, we concentrate on the spacial case of proportionally damped systems where the damping matrix D is given by $D = \alpha M + \beta K = \alpha I_n + \beta K$, with $\alpha, \beta > 0$ and $\alpha\beta < 1$. For this case, it was shown in [16, 44] that the second-order Krylov subspaces used for moment matching about an expansion point $\sigma \neq 0$ can be reduced to the classical Krylov subspaces for one-sided method ($\mathcal{V}_m = \mathcal{W}_m$). The result remains valid also for two-sided method.

Theorem 5.3.2 *If the damping matrix D verify $D = \alpha I_n + \beta K$, with $\alpha, \beta > 0$ and $\alpha\beta < 1$ then we have*

$$\tilde{\mathcal{K}}_m(K_\sigma^{-1}(2\sigma I_n + D), K_\sigma^{-1}, K_\sigma^{-1}B) = \mathcal{K}_m(K_\sigma^{-1}, K_\sigma^{-1}B),$$

and

$$\tilde{\mathcal{K}}_m(K_{\tilde{\sigma}}^{-T}(2\tilde{\sigma} I_n + D^T), K_{\tilde{\sigma}}^{-T}, K_{\tilde{\sigma}}^{-T}C^T) = \mathcal{K}_m(K_{\tilde{\sigma}}^{-T}, K_{\tilde{\sigma}}^{-T}C^T),$$

where $K_\sigma = \sigma^2 I_n + \sigma D + K$ and $K_{\tilde{\sigma}} = \tilde{\sigma}^2 I_n + \tilde{\sigma} D + K$.

This result can be naturally extended to the case where multiple interpolation points are chosen. In this case, we use the union of the input and the output Krylov subspaces corresponding to each interpolation point σ_i and $\tilde{\sigma}_i$, respectively. As in the case of first-order systems, we can use the approximation of the upper bound of the error

$\epsilon(s) = F(s) - F_m(s)$ to construct the next shifts σ_{m+1} and $\tilde{\sigma}_{m+1}$ as

$$\sigma_{m+1} = \arg \max_{s \in S} \|C_m P_m(s)^{-1}\|_2 \|R_B(s)\|_2 \quad \text{and} \quad \tilde{\sigma}_{m+1} = \arg \max_{s \in S} \|R_B(s)\|_2,$$

where

$$P_m(s) = C_m(s^2 I_m + sD_m + K_m),$$

and

$$R_B(s) = B - (s^2 I_n + sD + K)\mathcal{V}_m(s^2 I_m + sD_m + K_m)^{-1}B_m.$$

Let $\mathcal{V}_m, \mathcal{W}_m \in \mathbb{R}^{n \times mp}$ be the F-biorthogonal matrices generated by the modified rational global Lanczos algorithm (Algorithm 8) applied to the matrix Krylov subspaces $\mathcal{K}_m((\sigma_i^2 I_n + \sigma_i D + K), B, \Sigma_m)$ and $\mathcal{K}_m((\tilde{\sigma}_i^2 I_n + \tilde{\sigma}_i D + K)^T, C^T, \tilde{\Sigma}_m)$, where

$$\begin{aligned} \mathcal{K}_m((\sigma_i^2 I_n + \sigma_i D + K), B, \Sigma_m) &= \text{Span}\{(\sigma_1^2 I_n + \sigma_1 D + K)^{-1}B, \dots, \\ &\quad \prod_{i=1}^m (\sigma_i^2 I_n + \sigma_i D + K)^{-1}B\}, \end{aligned}$$

and

$$\begin{aligned} \mathcal{K}_m((\tilde{\sigma}_i^2 I_n + \tilde{\sigma}_i D + K)^T, C^T, \tilde{\Sigma}_m) &= \text{Span}\{(\tilde{\sigma}_1^2 I_n + \tilde{\sigma}_1 D + K)^{-T}C^T, \dots, \\ &\quad \prod_{i=1}^m (\tilde{\sigma}_i^2 I_n + \tilde{\sigma}_i D + K)^{-T}C^T\}. \end{aligned}$$

The reduced-order model of the second-order system (5.22) is given by

$$\begin{aligned} D_m &= ((\mathcal{W}_m^T \diamond D\mathcal{V}_m) \otimes I_p), \quad K_m = ((\mathcal{W}_m^T \diamond K\mathcal{V}_m) \otimes I_p), \\ B_m &= ((\mathcal{W}_m^T \diamond B) \otimes I_p) \quad \text{and} \quad C_m = C\mathcal{V}_m, \end{aligned}$$

where $D_m, K_m \in \mathbb{R}^{mp \times mp}$ and $B_m, C_m^T \in \mathbb{R}^{mp \times p}$. As in the case of first order systems, the bases \mathcal{V}_m and \mathcal{W}_m should span the union of the input and the output Krylov subspaces corresponding to each interpolation point σ_i and $\tilde{\sigma}_i$, respectively, for that the moment matching property holds, i.e.,

$$\begin{aligned} \text{Span}\{(\sigma_1^2 I_n + \sigma_1 D + K)^{-1}B, \dots, (\sigma_m^2 I_n + \sigma_m D + K)^{-1}B\} &\subseteq \text{Range}(\mathcal{V}_m) \\ \text{Span}\{(\tilde{\sigma}_1^2 I_n + \tilde{\sigma}_1 D + K)^{-T}C^T, \dots, (\tilde{\sigma}_m^2 I_n + \tilde{\sigma}_m D + K)^{-T}C^T\} &\subseteq \text{Range}(\mathcal{W}_m), \end{aligned}$$

see theorem 4.5 in [129].

5.4 Numerical experiments

In this section, we give some experimental results to show the effectiveness of the proposed approaches. All the experiments were performed on a computer of Intel Core i5 at 1.3GHz and 8GB of RAM. The algorithms were coded in Matlab 8.0. In order to show the performance of the adaptive modified rational global Lanczos (AMRGL) algorithm for state-space systems, two different methods were considered for selecting the shifts:

- **Method 1:** We give two initial shifts and we choose the shift σ_{m+1} as

$$\sigma_{m+1} = \arg \max_{s \in S} \|C_m P_m(s)^{-1}\|_2 \|\tilde{R}_B(s)\|_2.$$

- **Method 2:** We give the first initial shift and then σ_{m+1} is chosen as

$$\sigma_{m+1} = \arg \left(\max_{s \in S} \frac{1}{|r_m(s)|} \right).$$

We mention that when it was necessary, a global F -rebiorthogonalization procedure was used in the (AMRGL) algorithm. In all the experiments, the subspace dimension will be fixed a-priori and we consider the special case where the sequences of shifts $\{\sigma_i\}_{i=1}^m$ and $\{\tilde{\sigma}_i\}_{i=1}^m$ are equal. To compute the set of frequency S for the **Method 1**, we use always the function `lp_lgfrq` from LYAPACK [113].

Example 1. For this example, we applied the AMRGL algorithm to the first-order systems. The first model is the **modified FOM** model from [110]. We applied AMRGL to get a reduced order model of dimension 40 by using **Method 2** to choose the set of interpolation points with $\sigma_0^{(1)} = 10$ and $\sigma_0^{(2)} = 10^3$.

In the second experiment, we considered the **Modified RLC circuit** model of dimension $n = 3000$ by modifying the inputs and outputs of the RLC circuit model to get a MIMO systems with $p = 4$. We used **Method 2** to choose the interpolation points with $\sigma_0^{(1)} = -10^{-3}$ and $\sigma_0^{(2)} = 10^3$ and the reduced system was of order 25.

The top plots of Figure 5.1 (**modified FOM**) and Figure 5.2 (**Modified RLC circuit**) show the frequency responses of the original system (circles) compared with the

frequency responses of its approximation (solid plot). The bottom plot of these figures represent the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for different frequencies $\omega \in [10^{-6}, 10^6]$.

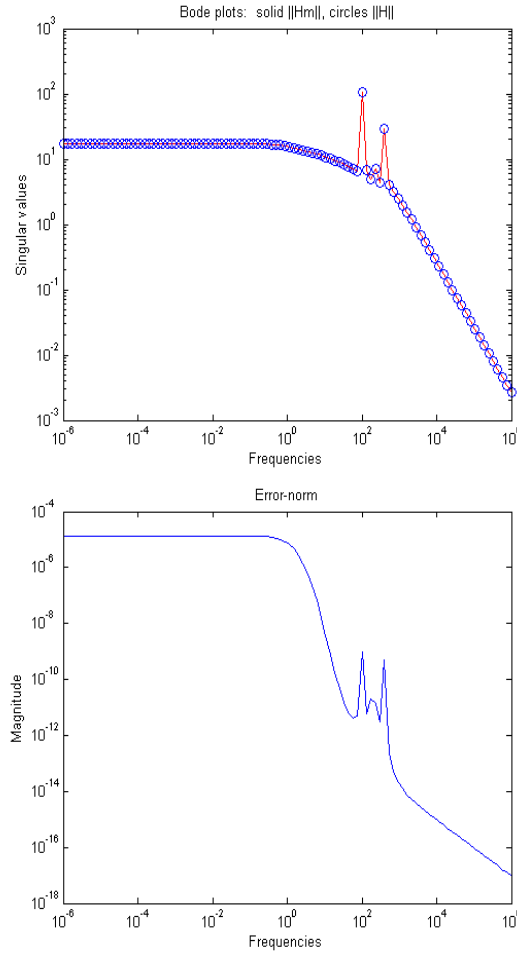


Figure 5.1: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the modified FOM model with $m = 40$.

As a third experiment, we considered the well known CD player model from [110] ($n = 120$ and $p = 2$). For this model test we used $\sigma_0^{(1)} = -10^{-4}$ and $\sigma_0^{(2)} = 10^4$ and we got a reduced order model of dimension 23. The obtained plots for this

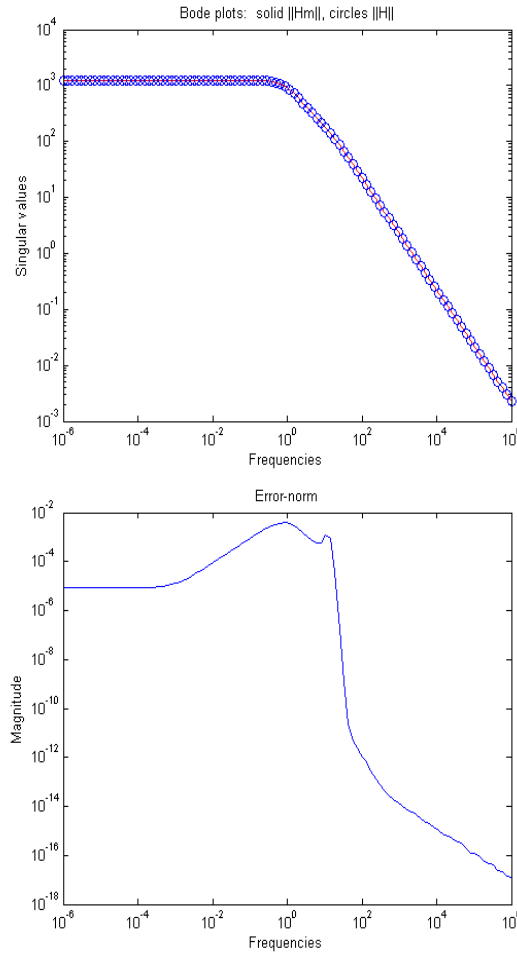


Figure 5.2: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the Modified RLC circuit model with $m = 25$

experiment are given in Figure 5.3.

Example 2. In this example, we considered the `fdm` and the `Rail3113` models. We plotted the \mathcal{H}_∞ error norm $\|F - F_m\|_\infty$ versus the number m of iterations using `Method 1` to select the shifts. For the `fdm` model, the corresponding matrix A is obtained from the centered finite difference discretization of the operator

$$L_A(u) = \Delta u - f(x, y) \frac{\partial u}{\partial x} - g(x, y) \frac{\partial u}{\partial y} - h(x, y)u,$$

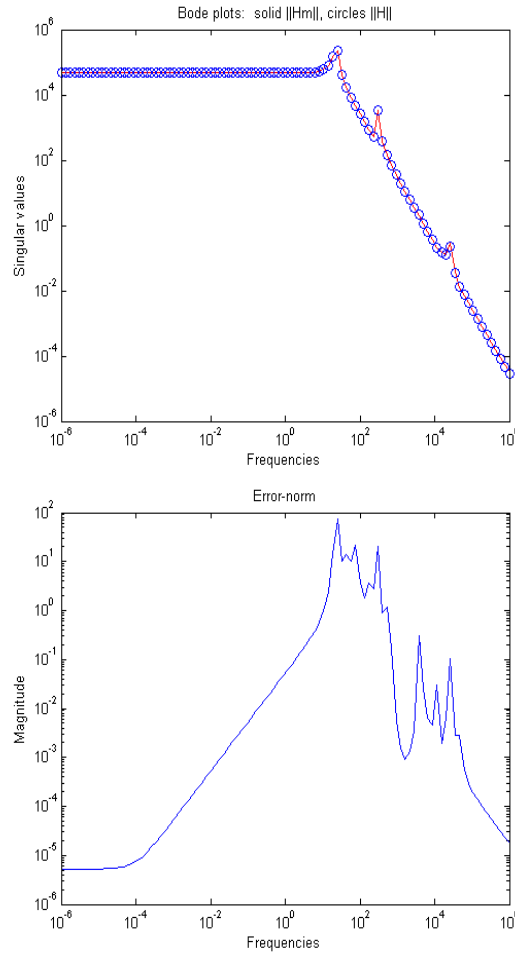


Figure 5.3: Top: $\|F(j\omega)\|_2$ and the approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the CD player model with $m = 23$.

on the unit square $[0, 1] \times [0, 1]$ with homogeneous Dirichlet boundary conditions with

$$f(x, y) = e^{xy}, \quad g(x, y) = \sin(xy) \quad \text{and} \quad h(x, y) = y^2 - x^2.$$

The matrices B and C were random matrices with entries uniformly distributed in $[0, 1]$. The number of inner grid points in each direction was $n_0 = 100$ and the dimension of A is $n = 10,000$. For this experiment, we used $p = 5$. The Rail3113 model is a first-order system of dimension $n = 3113$ and $p = 6$. As can be shown from Figure 5.4, the AMRGL algorithm gives good result with small values of m .

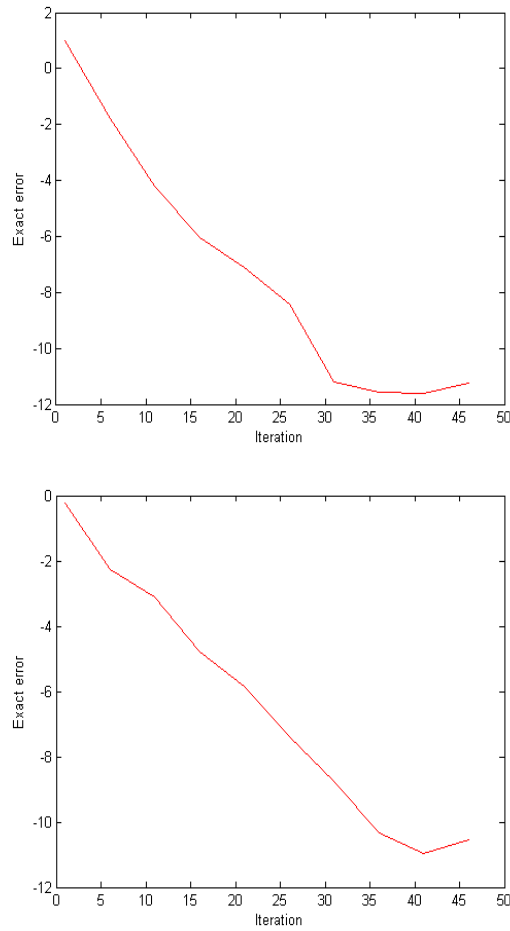


Figure 5.4: The \mathcal{H}_∞ error norms $\|F - F_m\|_\infty$ versus the number of iterations for the `fdm` model (top curve) and the `Rail3113` model (bottom curve).

Example 3. For this experiment, we considered second-order systems. We find an equivalent state space model and we applied the `AMRGL` algorithm to get a state space reduced system. As a first test model, we considered the `ISS` example. This system is a second order model of dimension $n = 135$, and the order of its corresponding linearized state space realization is equal to $N = 270$ with 3 inputs and 3 outputs. The reduced second order system was of dimension 15. The top curves of Figure 5.5 show the frequency responses of the original system

(circles) compared to the frequency responses of its approximations (solid). The bottom curve of this figure represents the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for frequencies $\omega \in [10^{-6}, 10^6]$. We used **Method 2** to choose the interpolation points with $\sigma_0^{(1)} = -10^2$ and $\sigma_0^{(1)} = 10^2$. We also applied the **AMRGL** algorithm to the

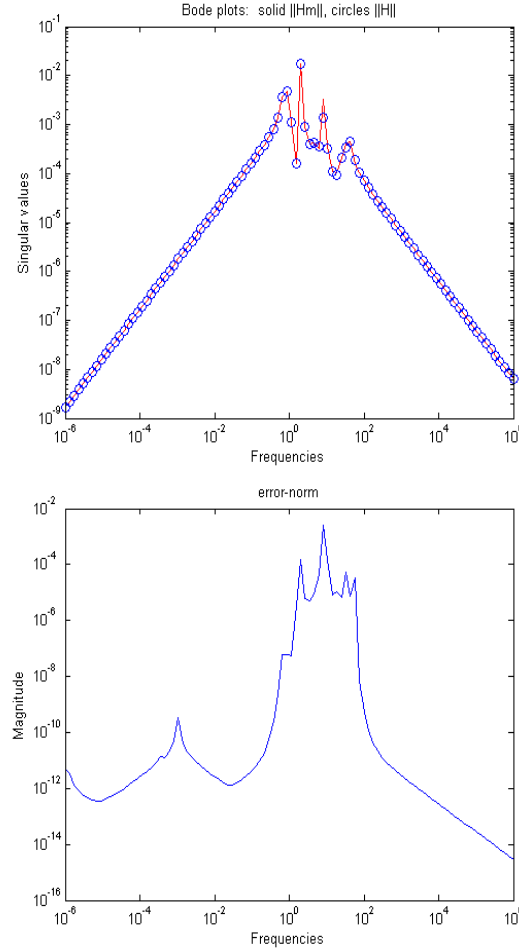


Figure 5.5: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the ISS model with $m = 30$.

modified beam model [110]. This system is a second-order model of dimension $n = 174$ with one input and one output. We modified the matrices B and C (random matrices) to get a MIMO system with four inputs and four outputs. The order of the reduced model was 16 and we used **Method 1** to choose the set

of interpolation points.

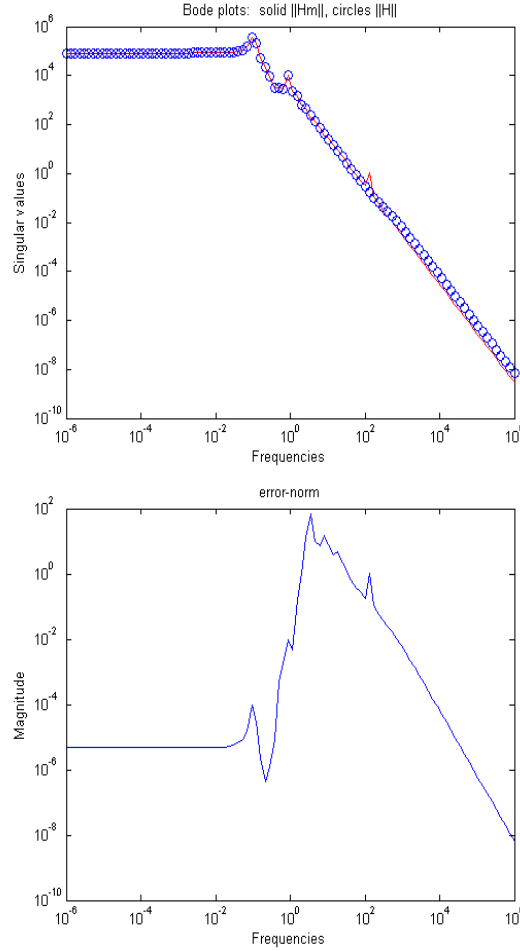


Figure 5.6: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the Modified Beam model with $m = 16$.

Example 4. For the last example, we considered the case of proportionally damped systems and we applied the AOMRGL directly to the second order models. The first test model considered is the second order system with exact condenser distribution used in [16]. Originally, this model is a SISO system and we modified the inputs and outputs to get a MIMO system. The mass matrix and stiffness matrix are, respectively, defined as

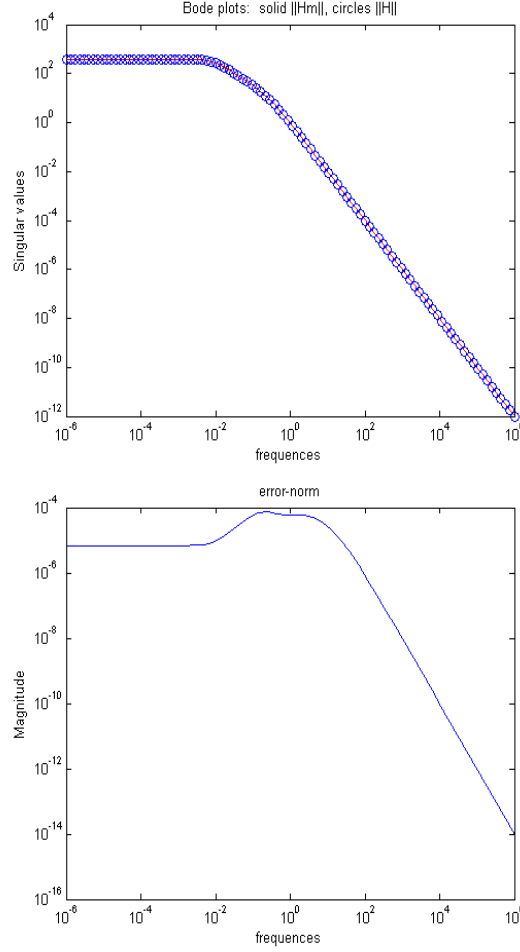


Figure 5.7: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the Exact Condenser Distribution model with $m = 25$.

The second test model in this experiment represents a system whose matrices are from FEM (Finite Element Modeling) software ANSYS. The FEM-based model yields a second order system where $M, K, D \in \mathbb{R}^{20 \times 20}$, $C \in \mathbb{R}^{2 \times 20}$ and $B \in \mathbb{R}^{20}$. Then we modified the input $B = [b_1, b_2]$ to be a 20×2 matrix such that b_2 is a random column vector. The second-order system is considered to be proportional, i.e., $D = \alpha M + \beta K$ where the damping parameters α and β are chosen as $\alpha = \beta = 2 \times 10^{-1}$. As in the last example, we modified the system matrices such that the mass matrix is the identity as in (5.23) and we reduced

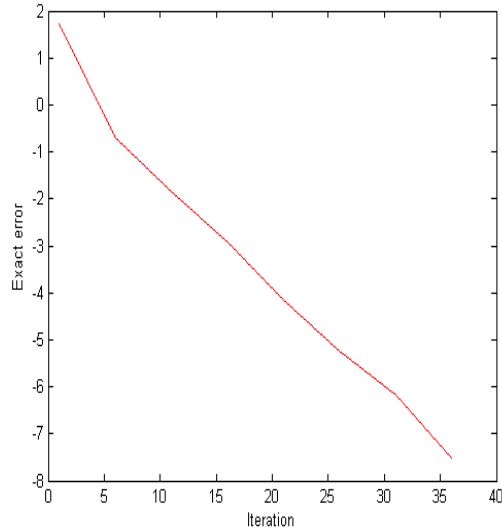


Figure 5.8: The \mathcal{H}_∞ error $\|F - F_m\|_\infty$ versus the number of iterations for the Exact Condenser Distribution model.

the system to order $m = 4$ in second-order form ($m = 8$ in state space). Figure 5.9 represent the largest singular value σ_{max} of the original system (circles) and the reduced order system (solid line) and the largest singular value of the error systems for different frequencies $\omega = [10^{-3}, 10^3]$.

5.5 Conclusion

In this Chapter, we proposed a new adaptive algorithm based on a modified rational global Lanczos process. The method was applied to get reduced order models that approximate large-scale MIMO and LTI linear dynamical systems. We derived new algebraic rational global Lanczos equations. We also applied our proposed approach to get reduced second-order models from second-order dynamical systems. We gave some theoretical results and present numerical experiments on some well known benchmark examples. We mention that preliminary experiments showed the competitiveness of the AMRGL method compared to the adap-

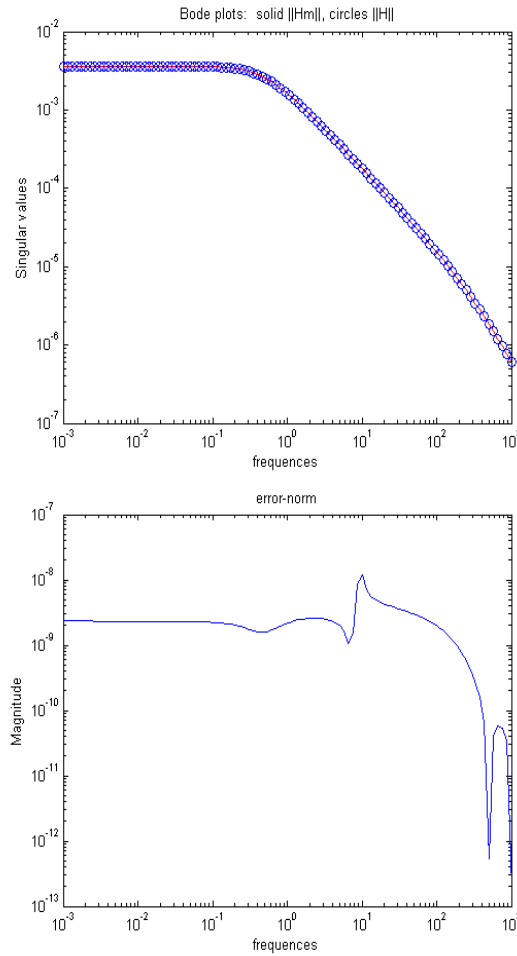


Figure 5.9: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the ANSYS model with $m = 4$.

tive Tangential Rational Krylov Subspace (TRKS) approach, proposed in [41] for the problems with multiple-input multiple-output, but more detail comparison is needed to determine the areas of the comparable strength of each method.

6

An extended block Lanczos algorithm for model reduction of large scale dynamical MIMO systems

The extended Krylov subspace can be considered as a special case of the rational Krylov subspace by taking $\sigma_{2i+1} = 0$ and $\sigma_{2i} = \infty$ for $i \geq 0$. Therefore, the advantage of this method is that we are not even mentioning the numerical difficulties that might arise for the selections of poles of the rational Krylov subspace [103].

Let $A \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{n \times p}$, the extended block Krylov subspace $\mathbb{K}_m^e(A, V)$ can be considered as the subspace of \mathbb{R}^n spanned by the columns of the matrices

$A^k V, k = -m, \dots, m-1$, i.e.,

$$\mathbb{K}_m^e(A, V) = \text{Range}\{A^{-m}V, \dots, A^{-2}V, A^{-1}V, V, AV, A^2V, \dots, A^{m-1}V\}.$$

It is clear that the subspace $\mathbb{K}_m^e(A, V)$ is a sum of two block Krylov subspaces. More precisely,

$$\mathbb{K}_m^e(A, V) = \mathbb{K}_m(A, V) + \mathbb{K}_m(A^{-1}, A^{-1}V),$$

where $\mathbb{K}_m(A, V) = \text{Range}\{V, AV, A^2V, \dots, A^{m-1}V\}$ is the classical block Krylov subspace related to A and $\mathbb{K}_m(A^{-1}, A^{-1}V)$ is related to the inverse of A .

In order to numerically approximate the action of a matrix function $f(A)$ on a vector v where $A \in \mathbb{R}^{n \times n}$ is a symmetric matrix and $v \in \mathbb{R}^n$, Druskin and Knizhnerman introduced in [41] the extended Arnoldi process. Simoncini in [131] exploited the extended block Arnoldi process to solve Lyapunov equations. In [80], authors showed that the extended block Arnoldi process still satisfies the well Arnoldi recursions and used it for computing approximate solutions to large scale continuous-time algebraic Riccati equations.

In this chapter, we show how to derive an extended block Lanczos process which is devoted to compute two bi-orthogonal matrices for the extended Krylov subspaces $\mathbb{K}_m^e(A, V)$ and $\mathbb{K}_m^e(A^T, W)$, where $W \in \mathbb{R}^{n \times p}$ and such that the first subspace is associated with A and A^{-1} , while the second one is related to A^T and A^{-T} . Another aim of this chapter is to show that the extended block Lanczos algorithm can be applied to model order reduction problems by combining it with moment matching techniques. More precisely, we show that the moments of the original transfer function are approximated by those of the reduced transfer function.

This chapter is organized as follows. In Section 1, we describe the extended block Lanczos algorithm and we explain how to obtain some new algebraic properties. The application of this method to model order reduction is considered in Section 2 where we show how to apply the extended block Lanczos process to MIMO dynamical systems in order to produce low-order dimensional systems. The last section is devoted to some numerical experiments for large and sparse problems to show the efficiency of the proposed method.

6.1 The extended block Lanczos algorithm

6.1.1 Description of the process

Let $A \in \mathbb{R}^{n \times n}$ and let V, W be two initial blocks of $\mathbb{R}^{n \times 2p}$. In this section, we first introduce the extended block Lanczos process for constructing two bi-orthogonal bases \mathbb{V}_{2m} and \mathbb{W}_{2m} of the Krylov subspaces $\mathbb{K}_m^e(A, V)$ and $\mathbb{K}_m^e(A^T, W)$.

Letting $\mathbb{V}_{2m} = \{V_1, V_2, \dots, V_m\}$ and $\mathbb{W}_{2m} = \{W_1, W_2, \dots, W_m\}$ where V_i, W_i (for $i = 1, \dots, m$) are $n \times 2p$ matrices. Then the bases are said to be bi-orthogonal if and only if the $n \times 2p$ matrices V_i and W_j satisfy the following biorthogonality condition

$$\begin{cases} W_j^T V_i = 0_p, & \text{if } i \neq j, \\ W_j^T V_i = I_{2p}, & \text{if } i = j. \end{cases} \quad (6.1)$$

Now, we describe the procedure that allows to compute the bi-orthogonal bases of the extended block Lanczos algorithm.

Initialization. Let's partition the two first block vectors V_1 and W_1 of the extended block Lanczos process as $V_1 = [v_1, v_2]$ and $W_1 = [w_1, w_2]$ where each $v_i, w_i \in \mathbb{R}^{n \times p}$ for $i = 1, 2$. To obtain V_1 and W_1 , we start by computing the QR decomposition of the $n \times 2p$ matrices $[V, A^{-1}V]$ and $[W, A^{-T}W]$, i.e.,

$$\begin{cases} [V, A^{-1}V] & = V_1 \Lambda_V, \\ [W, A^{-T}W] & = W_1 \Lambda_W \end{cases} \quad (6.2)$$

where Λ_V and Λ_W are $2p \times 2p$ upper triangular matrices and V_1, W_1 are $n \times 2p$ orthogonal matrices. Then, letting

$$W_1^T V_1 = P_0 D_0 Q_0^T,$$

be the SVD decomposition of $W_1^T V_1$, we define the new initial block vectors V_1 and W_1 as

$$V_1 = V_1 Q_0 D_0^{-\frac{1}{2}} \quad \text{and} \quad W_1 = W_1 P_0 D_0^{-\frac{1}{2}}.$$

Hence, thanks to the orthogonality of the matrices P_0, Q_0 and since D_0 is a diagonal matrix, we have

$$\begin{aligned} W_1^T V_1 &= D_0^{-\frac{1}{2}} P_0^T W_1^T V_1 Q_0 D_0^{-\frac{1}{2}} \\ &= I_{2p}. \end{aligned}$$

Iteration k. We assume that V_1, \dots, V_k and W_1, \dots, W_k have been computed. Next, we seek for $V_{k+1}, W_{k+1} \in \mathbb{R}^{n \times 2p}$ under the form $V_{k+1} = [v_{2k+1}, v_{2k+2}]$ and $W_{k+1} = [w_{2k+1}, w_{2k+2}]$ where the block vectors $v_{2k+1}, w_{2k+1} \in \mathbb{R}^{n \times p}$ are computed by orthogonalizing the matrix-vector products Av_{2k-1} and $A^T w_{2k-1}$ against v_1, v_2, \dots, v_{2k} and w_1, w_2, \dots, w_{2k} respectively, i.e., the block vectors v_{2k+1}, w_{2k+1} are computed via

$$\begin{cases} v_{2k+1} h_{2k+1,2k-1} &= A v_{2k-1} - \sum_{i=1}^{2k} v_i h_{i,2k-1}, \\ w_{2k+1} g_{2k+1,2k-1} &= A^T w_{2k-1} - \sum_{i=1}^{2k} w_i g_{i,2k-1}, \end{cases} \quad (6.3)$$

where the coefficients $h_{1,2k-1}, \dots, h_{2k,2k-1}$ and $g_{1,2k-1}, \dots, g_{2k,2k-1}$ are $p \times p$ matrices obtained respectively by imposing the orthogonalities

$$v_{2k+1} \perp [w_1, w_2, \dots, w_{2k}] \quad \text{and} \quad w_{2k+1} \perp [v_1, v_2, \dots, v_{2k}]. \quad (6.4)$$

In this case, we have

$$h_{i,2k-1} = w_i^T A v_{2k-1} \quad \text{and} \quad g_{i,2k-1} = v_i^T A^T w_{2k-1}, \quad \text{for } i = 1, 2, \dots, 2k.$$

Similarly, the block vectors $v_{2k+2}, w_{2k+2} \in \mathbb{R}^{n \times p}$ are computed by orthogonalizing the matrix-vector products $A^{-1}v_{2k}$ and $(A^T)^{-1}w_{2k}$ against $v_1, v_2, \dots, v_{2k+1}$ and $w_1, w_2, \dots, w_{2k+1}$ respectively, i.e., we generate the vectors v_{2k+2}, w_{2k+2} satisfying:

$$\begin{cases} v_{2k+2} h_{2k+2,2k} &= A^{-1} v_{2k} - \sum_{i=1}^{2k+1} v_i h_{i,2k}, \\ w_{2k+2} g_{2k+2,2k} &= (A^T)^{-1} w_{2k} - \sum_{i=1}^{2k+1} w_i g_{i,2k}, \end{cases} \quad (6.5)$$

where again imposing the orthogonality conditions

$$v_{2k+2} \perp w_1, \dots, w_{2k+1} \quad \text{and} \quad w_{2k+2} \perp v_1, \dots, v_{2k+1}, \quad (6.6)$$

we easily verify that the $p \times p$ coefficient matrices $h_{1,2k}, \dots, h_{2k+1,2k}$ and $g_{1,2k}, \dots, g_{2k+1,2k}$, are respectively given by :

$$h_{i,2k} = w_i^T A^{-1} v_{2k} \quad \text{and} \quad g_{i,2k} = v_i^T (A^T)^{-1} w_{2k}, \quad \text{for } i = 1, 2, \dots, 2k+1.$$

$h_{2k+1,2k-1}$ and $g_{2k+1,2k-1}$ are also $p \times p$ matrices that normalize the block vectors v_{2k+1} and w_{2k+1} . They are computed using the QR and SVD decompositions (see Algorithm 10). $h_{2k+2,2k}$ and $g_{2k+2,2k}$ are $p \times p$ matrices that normalize the block vectors v_{2k+2} and w_{2k+2} , and they are also computed using the QR and the SVD decomposition (see Algorithm 10).

The extended block Lanczos process described above allows to compute two bi-orthogonal matrices $\mathbb{V}_{2m+2} = [V_1, \dots, V_{m+1}]$ and $\mathbb{W}_{2m+2} = [W_1, \dots, W_{m+1}]$, such that $V_k = [v_{2k-1}, v_{2k}]$ and $W_k = [w_{2k-1}, w_{2k}]$ for $k = 1, \dots, m+1$. This algorithm constructs also two $2(m+1)p \times 2mp$ upper block Hessenberg matrices $\tilde{\mathbb{H}}_{2m} = [h_{i,j}]$ and $\tilde{\mathbb{G}}_{2m} = [g_{i,j}]$, where $h_{i,j}, g_{i,j} \in \mathbb{R}^{p \times p}$ for $i = 1, \dots, 2m+2, j = 1, \dots, 2m$.

Next, we give some properties for the biorthogonal matrices $\mathbb{V}_{2m+2}, \mathbb{W}_{2m+2}$, and the upper block Hessenberg matrices $\tilde{\mathbb{H}}_{2m}, \tilde{\mathbb{G}}_{2m}$. We consider the following notations :

\mathbb{V}_m^o and \mathbb{W}_m^o are matrices of $\mathbb{R}^{n \times mp}$ formed by the block columns of odd indices of the matrices \mathbb{V}_{2m} and \mathbb{W}_{2m} , respectively.

\mathbb{V}_m^e and \mathbb{W}_m^e are the matrices formed by the block columns of even indices of the bases \mathbb{V}_{2m} and \mathbb{W}_{2m} , respectively.

\mathbb{H}_m^o and \mathbb{G}_m^o are the matrices of $\mathbb{R}^{(2m+1)p \times mp}$ formed by the block columns of odd indices of the matrices $\tilde{\mathbb{H}}_{2m}$ and $\tilde{\mathbb{G}}_{2m}$, respectively.

\mathbb{H}_m^e and \mathbb{G}_m^e are the matrices of $\mathbb{R}^{2(m+1)p \times mp}$ formed by the block columns of even indices of the matrices $\tilde{\mathbb{H}}_{2m}$ and $\tilde{\mathbb{G}}_{2m}$, respectively.

The bloc Hessenberg matrices $\hat{\mathbb{H}}_m^o$ and $\hat{\mathbb{G}}_m^o$ correspond to the block columns and the block rows of odd indices of the matrices $\mathbb{H}_{2m} = [h_{i,j}]_{i=1, \dots, 2m}^{j=1, \dots, 2m}$ and $\mathbb{G}_{2m} = [g_{i,j}]_{i=1, \dots, 2m}^{j=1, \dots, 2m}$, respectively. Finally, $\check{\mathbb{H}}_m^e$ and $\check{\mathbb{G}}_m^e$ are formed by the block columns and the block row of even indices of \mathbb{H}_{2m} and \mathbb{G}_{2m} , respectively.

We have the following result.

Proposition 6.1.1 *Using the above notations, and let $\mathbb{V}_{2m+1} = [\mathbb{V}_{2m}, v_{2m+1}]$ and $\mathbb{W}_{2m+1} = [\mathbb{W}_{2m}, w_{2m+1}]$. Then, we have*

$$A\mathbb{V}_m^o = \mathbb{V}_{2m+1}\mathbb{H}_m^o, \quad (6.7)$$

$$A^T\mathbb{W}_m^o = \mathbb{W}_{2m+1}\mathbb{G}_m^o, \quad (6.8)$$

$$A^{-1}\mathbb{V}_m^e = \mathbb{V}_{2m+2}\mathbb{H}_m^e, \quad (6.9)$$

$$A^{-T}\mathbb{W}_m^e = \mathbb{W}_{2m+2}\mathbb{G}_m^e. \quad (6.10)$$

Furthermore, the matrices \mathbb{H}_{2m} and \mathbb{G}_{2m} are $2p \times 2p$ tridiagonal matrices.

Proof. Equations (6.7)-(6.10) can be easily proven by considering the relations (6.3) and (6.5), for $k = 1, \dots, m$, and the biorthogonality condition.

Now, using equations (6.7) and (6.8), and the biorthogonality condition, we get

$$\begin{aligned} (\mathbb{W}_m^o)^T A \mathbb{V}_m^o &= (\mathbb{W}_m^o)^T \mathbb{V}_{2m+1} \mathbb{H}_m^o \\ &= \widehat{\mathbb{H}}_m^o, \end{aligned}$$

and

$$\begin{aligned} (\mathbb{V}_m^o)^T A^T \mathbb{W}_m^o &= (\mathbb{V}_m^o)^T \mathbb{W}_{2m+1} \mathbb{G}_m^o \\ &= \widehat{\mathbb{G}}_m^o, \end{aligned}$$

which gives

$$\widehat{\mathbb{H}}_m^o = (\widehat{\mathbb{G}}_m^o)^T. \quad (6.11)$$

In the same manner, we can use equations (6.9) and (6.10) to show that

$$\check{\mathbb{H}}_m^e = (\check{\mathbb{G}}_m^e)^T. \quad (6.12)$$

Comparing the members of equalities (6.11) and (6.12), we note that the first member in both equalities is an upper block Hessenberg matrix while the second is a lower block Hessenberg matrix. Then, $\widehat{\mathbb{H}}_m^o$, $\widehat{\mathbb{G}}_m^o$, $\check{\mathbb{H}}_m^e$ and $\check{\mathbb{G}}_m^e$ are $p \times p$ tridiagonal matrices. Therefore, \mathbb{H}_{2m} and \mathbb{G}_{2m} are $2p \times 2p$ tridiagonal matrices.

Using the fact that \mathbb{H}_{2m} and \mathbb{G}_{2m} are $2p \times 2p$ tridiagonal matrices, and the sub-diagonal blocks are $2p \times 2p$ upper triangular matrices. then, the relations given in (6.3) and (6.5) can be simplified as

$$\begin{aligned} v_{2k+1} h_{2k+1,2k-1} &= A v_{2k-1} - v_{2k-3} h_{2k-3,2k-1} - v_{2k-2} h_{2k-2,2k-1} \\ &\quad - v_{2k-1} h_{2k-1,2k-1} - v_{2k} h_{2k,2k-1} \end{aligned} \quad (6.13)$$

$$\begin{aligned} w_{2k+1} g_{2k+1,2k-1} &= A^T w_{2k-1} - w_{2k-3} g_{2k-3,2k-1} - w_{2k-2} g_{2k-2,2k-1} \\ &\quad - w_{2k-1} g_{2k-1,2k-1} - w_{2k} g_{2k,2k-1}, \end{aligned} \quad (6.14)$$

and

$$\begin{aligned} v_{2k+2} h_{2k+2,2k} &= A^{-1} v_{2k} - v_{2k-2} h_{2k-2,2k} - v_{2k-1} h_{2k-1,2k} \\ &\quad - v_{2k} h_{2k,2k} - v_{2k+1} h_{2k+1,2k} \end{aligned} \quad (6.15)$$

$$\begin{aligned} w_{2k+2} g_{2k+2,2k} &= (A^T)^{-1} w_{2k} - w_{2k-2} g_{2k-2,2k} - w_{2k-1} g_{2k-1,2k} \\ &\quad - w_{2k} g_{2k,2k} - w_{2k+1} g_{2k+1,2k}. \end{aligned} \quad (6.16)$$

Now, we write equations (6.13)-(6.16) in the following form

$$\begin{aligned}
 [v_{2k+1} \ v_{2k+2}] \begin{pmatrix} h_{2k+1,2k-1} & h_{2k+1,2k} \\ 0 & h_{2k+2,2k} \end{pmatrix} &= [Av_{2k-1} \ A^{-1}v_{2k}] \\
 &- [v_{2k-1} \ v_{2k}] \begin{pmatrix} h_{2k-1,2k-1} & h_{2k-1,2k} \\ h_{2k,2k-1} & h_{2k,2k} \end{pmatrix} \\
 &- [v_{2k-3} \ v_{2k-2}] \begin{pmatrix} h_{2k-3,2k-1} & 0 \\ h_{2k-2,2k-1} & h_{2k-2,2k} \end{pmatrix}, \tag{6.17}
 \end{aligned}$$

and

$$\begin{aligned}
 [w_{2k+1} \ w_{2k+2}] \begin{pmatrix} g_{2k+1,2k-1} & g_{2k+1,2k} \\ 0 & g_{2k+2,2k} \end{pmatrix} &= [A^T w_{2k-1} \ A^{-T} w_{2k}] \\
 &- [w_{2k-1} \ w_{2k}] \begin{pmatrix} g_{2k-1,2k-1} & g_{2k-1,2k} \\ g_{2k,2k-1} & g_{2k,2k} \end{pmatrix} \\
 &- [w_{2k-3} \ w_{2k-2}] \begin{pmatrix} g_{2k-3,2k-1} & 0 \\ g_{2k-2,2k-1} & g_{2k-2,2k} \end{pmatrix}. \tag{6.18}
 \end{aligned}$$

Set

$$\left\{ \begin{array}{l} V_{k-1} = [v_{2k-3} \ v_{2k-2}], \ V_k = [v_{2k-1} \ v_{2k}], \ V_{k+1} = [v_{2k+1} \ v_{2k+2}] \\ W_{k-1} = [w_{2k-3} \ w_{2k-2}], \ W_k = [w_{2k-1} \ w_{2k}], \ W_{k+1} = [w_{2k+1} \ w_{2k+2}] \\ U_{k+1} = [Av_{2k-1} \ A^{-1}v_{2k}], \ S_{k+1} = [A^T w_{2k-1} \ A^{-T} w_{2k}] \end{array} \right.$$

and

$$\left\{ \begin{array}{l} N_k = \begin{pmatrix} h_{2k-3,2k-1} & 0 \\ h_{2k-2,2k-1} & h_{2k-2,2k} \end{pmatrix}, \ \tilde{N}_k = \begin{pmatrix} g_{2k-3,2k-1} & 0 \\ g_{2k-2,2k-1} & g_{2k-2,2k} \end{pmatrix} \\ C_k = \begin{pmatrix} h_{2k-1,2k-1} & h_{2k-1,2k} \\ h_{2k,2k-1} & h_{2k,2k} \end{pmatrix}, \ \tilde{C}_k = \begin{pmatrix} g_{2k-1,2k-1} & g_{2k-1,2k} \\ g_{2k,2k-1} & g_{2k,2k} \end{pmatrix} \\ A_k = \begin{pmatrix} h_{2k+1,2k-1} & h_{2k+1,2k} \\ 0 & h_{2k+2,2k} \end{pmatrix}, \ \tilde{A}_k = \begin{pmatrix} g_{2k+1,2k-1} & g_{2k+1,2k} \\ 0 & g_{2k+2,2k} \end{pmatrix} \end{array} \right.$$

Therefore, equations (6.17) and (6.18) can be written as

$$\begin{cases} V_{k+1}A_k = U_{k+1} - V_kC_k - V_{k-1}N_k, \\ W_{k+1}\tilde{A}_k = S_{k+1} - W_k\tilde{C}_k - W_{k-1}\tilde{N}_k. \end{cases}$$

Finally, the extended block Lanczos algorithm is summarized as follows.

Algorithm 10 The extended block Lanczos algorithm (EBLA)

1. **Input:** $A \in \mathbb{R}^{n \times n}$, $V, W \in \mathbb{R}^{n \times 2p}$.
 2. **Initialize:** $V_0 = W_0 = 0_{2p}$ and $N_1 = \tilde{N}_1 = 0_{2p}$.
 3. **Set** $U_1 = [V, A^{-1}V]$ and $S_1 = [W, A^{-T}W]$ and construct $V_1, W_1 \in \mathbb{R}^{n \times 2p}$ such that $W_1^T V_1 = I_{2p}$;
 4. **Initialize:** $\mathbb{V}_2 = [V_1]$ and $\mathbb{W}_2 = [W_1]$.
 5. **For** $k = 1, \dots, m$
 6. $U_{k+1} = [Av_{2k-1}, A^{-1}v_{2k}]$ and $S_{k+1} = [A^T w_{2k-1}, A^{-T}w_{2k}]$;
 7. $N_k = W_{k-1}^T U_{k+1}$, $C_k = W_k^T U_{k+1}$ and $\tilde{N}_k = V_{k-1}^T S_{k+1}$, $\tilde{C}_k = V_k^T S_{k+1}$,
 8. $U_{k+1} = U_{k+1} - V_k C_k - V_{k-1} N_k$ and $S_{k+1} = S_{k+1} - W_k \tilde{C}_k - W_{k-1} \tilde{N}_k$;
 9. $U_{k+1} = V_{k+1} A_{k+1}$ and $S_{k+1} = W_{k+1} \tilde{A}_{k+1}$; (QR factorization)
 10. $W_{k+1}^T V_{k+1} = P_k D_k Q_k^T$; (Singular Value Decomposition)
 11. $V_{k+1} = V_{k+1} Q_k D_k^{-1/2}$ and $W_{k+1} = W_{k+1} P_k D_k^{-1/2}$;
 12. $A_{k+1} = D_k^{1/2} Q_k^T A_{k+1}$ and $\tilde{A}_{k+1} = D_k^{1/2} P_k^T \tilde{A}_{k+1}$;
 13. $\mathbb{V}_{2k+2} = [\mathbb{V}_{2k}, V_{k+1}]$; $\mathbb{W}_{2k+2} = [\mathbb{W}_{2k}, W_{k+1}]$;
 14. **endFor.**
-

After m steps, Algorithm 10 builds two bi-orthogonal bases \mathbb{V}_{2m+2} and \mathbb{W}_{2m+2} , and two $2(m+1)p \times 2mp$ upper block Hessenberg matrices $\tilde{\mathbb{H}}_{2m}$ and $\tilde{\mathbb{G}}_{2m}$ defined as follows

$$\tilde{\mathbb{H}}_{2m} = \begin{bmatrix} \mathbb{H}_{2m} \\ A_{m+1}E_m^T \end{bmatrix} \quad \text{and} \quad \tilde{\mathbb{G}}_{2m} = \begin{bmatrix} \mathbb{G}_{2m} \\ \tilde{A}_{m+1}E_m^T \end{bmatrix},$$

where

$$\mathbb{H}_{2m} = \begin{pmatrix} C_1 & N_2 & & & & \\ A_2 & C_2 & & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & N_m & \\ & & & A_m & C_m & \end{pmatrix} \quad \text{and} \quad \mathbb{G}_{2m} = \begin{pmatrix} \tilde{C}_1 & \tilde{N}_2 & & & & \\ \tilde{A}_2 & \tilde{C}_2 & & & & \\ & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \tilde{N}_m & \\ & & & \tilde{A}_m & \tilde{C}_m & \end{pmatrix}, \quad (6.19)$$

and E_m is last $2mp \times 2p$ block of the identity matrix I_{2mp} .

6.1.2 Theoretical results

In this subsection, we derive some theoretical results of the extended block Lanczos Algorithm.

Let $\mathbb{T}_{2m} = \mathbb{W}_{2m}^T A \mathbb{V}_{2m} \in \mathbb{R}^{2mp \times 2mp}$. Using the same technique in [131] (for the extended Arnoldi process), we can easily verified that \mathbb{T}_{2m} is block upper Hessenberg with $2p \times 2p$ blocks. In the following, we will also consider the $2mp \times 2mp$ matrix defined as

$$\mathbb{L}_{2m} = \mathbb{W}_{2m}^T A^{-1} \mathbb{V}_{2m}.$$

Notice that we can check that \mathbb{L}_{2m} is also $2p \times 2p$ block upper Hessenberg matrix.

Proposition 6.1.2 *Suppose that m steps of Algorithm 10 have been carried out, and let $\tilde{\mathbb{T}}_{2m} = \mathbb{W}_{2m+2}^T A \mathbb{V}_{2m}$ and $\tilde{\mathbb{L}}_{2m} = \mathbb{W}_{2m+2}^T A^{-1} \mathbb{V}_{2m}$. Then, the following relations hold*

$$A \mathbb{V}_{2m} = \mathbb{V}_{2m} \mathbb{T}_{2m} + V_{m+1} T_{m+1,m} E_m^T \quad (6.20)$$

$$A^{-1} \mathbb{V}_{2m} = \mathbb{V}_{2m} \mathbb{L}_{2m} + V_{m+1} L_{m+1,m} E_m^T \quad (6.21)$$

$$A^T \mathbb{W}_{2m} = \mathbb{W}_{2m} \mathbb{T}_{2m}^T + W_{m+1} \tilde{T}_{m+1,m} E_m^T \quad (6.22)$$

$$A^{-T} \mathbb{W}_{2m} = \mathbb{W}_{2m} \mathbb{L}_{2m}^T + W_{m+1} \tilde{L}_{m+1,m} E_m^T, \quad (6.23)$$

where $\tilde{T}_{m+1,m} = V_{m+1}^T A^T \mathbb{W}_{2m}$ and $\tilde{L}_{m+1,m} = V_{m+1}^T A^{-T} \mathbb{V}_{2m}$.

Proof. To prove the first equation, we start by using the fact that $A\mathbb{K}_m^e(A, V) \subseteq \mathbb{K}_{m+1}^e(A, V)$, and the bi-orthogonality condition. Then, there exists a matrix T such that $A\mathbb{V}_{2m} = \mathbb{V}_{2m+2}T$. Hence $T = \mathbb{W}_{2m+2}^T A\mathbb{V}_{2m}$ which gives $T = \tilde{\mathbb{T}}_{2m}$. Therefore

$$A\mathbb{V}_{2m} = \mathbb{V}_{2m+2}\tilde{\mathbb{T}}_{2m}.$$

Now, since we have $\mathbb{V}_{2m+2} = [\mathbb{V}_{2m}, V_{m+1}]$, $\mathbb{W}_{2m+2} = [\mathbb{W}_{2m}, W_{m+1}]$ and as $\mathbb{T}_{2m+2} = \mathbb{W}_{2m+2}^T A\mathbb{V}_{2m+2}$ is block upper Hessenberg matrix, then

$$T_{m+1,m}E_m^T = W_{m+1}^T A\mathbb{V}_{2m}.$$

Hence

$$\tilde{\mathbb{T}}_{2m} = \mathbb{W}_{2m+2}^T A\mathbb{V}_{2m} = \begin{bmatrix} \mathbb{T}_{2m} \\ W_{m+1}^T A\mathbb{V}_{2m} \end{bmatrix} = \begin{bmatrix} \mathbb{T}_{2m} \\ T_{m+1,m}E_m^T \end{bmatrix}$$

which completes the proof of (6.20).

For the second relation, we will follow the same procedure. As $\mathbb{L}_{2m+2} = \mathbb{W}_{2m+2}^T A^{-1}\mathbb{V}_{2m+2}$ is block upper Hessenberg matrix, we have

$$W_{m+1}^T A^{-1}\mathbb{V}_{2m} = L_{m+1,m}E_m^T,$$

and then the upper block Hessenberg matrix $\tilde{\mathbb{L}}_{2m}$ can be written as

$$\tilde{\mathbb{L}}_{2m} = \mathbb{W}_{2m+2}^T A^{-1}\mathbb{V}_{2m} = \begin{bmatrix} \mathbb{L}_{2m} \\ W_{m+1}^T A^{-1}\mathbb{V}_{2m} \end{bmatrix} = \begin{bmatrix} \mathbb{L}_{2m} \\ L_{m+1,m}E_m^T \end{bmatrix}.$$

Using the bi-orthogonality condition and the fact that $A^{-1}\mathbb{K}_m^e(A, V) \subseteq \mathbb{K}_{m+1}^e(A, V)$, then there exists a matrix L such that

$$A^{-1}\mathbb{V}_{2m} = \mathbb{V}_{2m+2}L.$$

Hence

$$L = \mathbb{W}_{2m+2}^T A^{-1}\mathbb{V}_{2m},$$

which gives

$$L = \tilde{\mathbb{L}}_{2m}.$$

Therefore

$$A^{-1}\mathbb{V}_{2m} = \mathbb{V}_{2m+2}\tilde{\mathbb{L}}_{2m} = \mathbb{V}_{2m}\mathbb{L}_{2m} + V_{m+1}L_{m+1,m}E_m^T.$$

In a similar way, we can use $A^T\mathbb{K}_m^e(A^T, W) \subseteq \mathbb{K}_{m+1}^e(A^T, W)$ and $A^{-T}\mathbb{K}_m^e(A^T, W) \subseteq$

$\mathbb{K}_{m+1}^e(A^T, W)$ to show equations (6.22) and (6.23), respectively.

Let $\tilde{\mathbb{T}}_{2m}$ and $\tilde{\mathbb{H}}_{2m}$ be the block upper Hessenberg matrices defined earlier. The computation of $\tilde{\mathbb{T}}_{2m}$ seems to require additional matrix-vector products with A and extra inner products of long vectors. To completely avoids this expensive step, we next derive a recursions to compute the block columns of the matrices $\tilde{\mathbb{T}}_{2m}$ directly from the block columns of the upper block Hessenberg matrix $\tilde{\mathbb{H}}_{2m}$ without requiring the matrix-vector products with A . We start by defining the following notations which will be used later.

1. For $k = 1, \dots, m$, we define now V_k and W_k as

$$V_k = [V_k^{(1)}, V_k^{(2)}] \text{ and } W_k = [W_k^{(1)}, W_k^{(2)}],$$

where $V_k^{(1)}$ (.resp $W_k^{(1)}$) is the first p columns of V_k (.resp W_k) and $V_k^{(2)}$ (.resp $W_k^{(2)}$) is the second p columns of V_k (.resp W_k).

2. For $k = 1, \dots, m$, we partition the upper triangular matrix $A_{k+1} \in \mathbb{R}^{2p \times 2p}$, computed from Algorithm 10, as

$$A_{k+1} = \begin{bmatrix} A_{k+1}^{(1,1)} & A_{k+1}^{(1,2)} \\ 0 & A_{k+1}^{(2,2)} \end{bmatrix}.$$

3. Let \mathbb{H}_{2m} be the $2mp \times 2mp$ block upper Hessenberg matrix defined in (6.19), and $\tilde{e}_i = e_i \otimes I_p$ where e_i is the vectors of the canonical basis. From Algorithm 10, we have

$$U_{m+1} = [U_{m+1}^{(1)}, U_{m+1}^{(2)}] = [AV_m^{(1)}, A^{-1}V_m^{(2)}] - \mathbb{V}_{2m}\mathbb{H}_{2m}[\tilde{e}_{2k-1}, \tilde{e}_{2k}] \quad (6.24)$$

and

$$U_{m+1} = V_{m+1}A_{m+1}. \quad (6.25)$$

Then

$$V_{m+1} = U_{m+1}A_{m+1}^{-1}, \quad (6.26)$$

where A_{m+1}^{-1} is also $2p \times 2p$ upper triangular matrix.

Proposition 6.1.3 *Let $\tilde{\mathbb{T}}_{2m}$ and $\tilde{\mathbb{H}}_{2m}$ be the upper Hessenberg matrices defined earlier. Then, for $k = 1$*

$$\tilde{\mathbb{T}}_{2m}\tilde{e}_1 = \tilde{\mathbb{H}}_{2m}\tilde{e}_1$$

$$\tilde{\mathbb{T}}_{2m}\tilde{e}_2 = [\tilde{e}_1\Lambda_V^{(1,1)} - \tilde{\mathbb{H}}_{2m}\tilde{e}_1\Lambda_V^{(1,2)}](\Lambda_V^{(2,2)})^{-1}. \quad (6.27)$$

While ($k = 2, \dots, m$)

$$\tilde{\mathbb{T}}_{2m}\tilde{e}_{2k-1} = \tilde{\mathbb{H}}_{2m}\tilde{e}_{2k-1},$$

$$\tilde{\mathbb{T}}_{2m}\tilde{e}_{2k} = \tilde{\mathbb{T}}_{2m}\tilde{e}_{2k-1}\chi^{(k)} + \left(\tilde{e}_{2k-2} - \begin{bmatrix} \tilde{\mathbb{T}}_{2k-2} \\ 0_{2(m-k+1)p \times (2k-2)p} \end{bmatrix} \mathbb{H}_{2k-2}\tilde{e}_{2k-2} \right) (A_k^{-1})^{(2,2)},$$

where

$$\chi^{(k)} = A_k^{(1,1)}(A_k^{-1})^{(1,2)}.$$

Proof. To prove the odd block columns, we start by considering relations (6.24) and (6.25) to get

$$\begin{aligned} AV_k^{(1)} &= U_{k+1}\tilde{e}_1 + \mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k-1} \\ &= V_{k+1}A_{k+1}\tilde{e}_1 + \mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k-1} \\ &= \mathbb{V}_{2k+2}\tilde{\mathbb{H}}_{2k}\tilde{e}_{2k-1}. \end{aligned} \quad (6.28)$$

Pre-multiplying the above equality on the left by \mathbb{W}_{2m+2}^T , then

$$\mathbb{W}_{2m+2}^T AV_k^{(1)} = \begin{bmatrix} I_{2(k+1)p} \\ 0_{2(m-k)p \times 2(k+1)p} \end{bmatrix} \tilde{\mathbb{H}}_{2k}\tilde{e}_{2k-1},$$

hence

$$\tilde{\mathbb{T}}_{2m}\tilde{e}_{2k-1} = \begin{bmatrix} \tilde{\mathbb{H}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \tilde{e}_{2k-1} = \tilde{\mathbb{H}}_{2m}\tilde{e}_{2k-1}.$$

To prove (6.27), we start by using the QR decomposition of $[V, A^{-1}V]$ defined in (6.2) such that

$$\begin{aligned} [V, A^{-1}V] = V_1\Lambda_V &= [V_1^{(1)}, V_1^{(2)}] \begin{bmatrix} \Lambda_V^{(1,1)} & \Lambda_V^{(1,2)} \\ 0 & \Lambda_V^{(2,2)} \end{bmatrix} \\ &= [V_1^{(1)}\Lambda_V^{(1,1)}, V_1^{(1)}\Lambda_V^{(1,2)} + V_1^{(2)}\Lambda_V^{(2,2)}]. \end{aligned} \quad (6.29)$$

If $\Lambda_V^{(1,1)}$ and $\Lambda_V^{(2,2)}$ are non-singular, we obtain

$$A^{-1}V_1^{(1)} = A^{-1}V(\Lambda_V^{(1,1)})^{-1} = [V_1^{(1)}\Lambda_V^{(1,2)} + V_1^{(2)}\Lambda_V^{(2,2)}](\Lambda_V^{(1,1)})^{-1},$$

then

$$AV_1^{(2)} = [V_1^{(1)}\Lambda_V^{(1,1)} - AV_1^{(1)}\Lambda_V^{(1,2)}](\Lambda_V^{(2,2)})^{-1}.$$

Pre-multiplying on the left by \mathbb{W}_{2m+2} to get equation (6.27).

For the other even block columns, we proceed as follows. From (6.24), we have

$$AU_{k+1}^{(2)} = V_k^{(2)} - AV_{2k}\mathbb{H}_{2k}\tilde{e}_{2k},$$

hence

$$\begin{aligned} \mathbb{W}_{2m+2}^T AU_{k+1}^{(2)} &= \mathbb{W}_{2m+2}^T V_k^{(2)} - \mathbb{W}_{2m+2}^T AV_{2k}\mathbb{H}_{2k}\tilde{e}_{2k} \\ &= \tilde{e}_{2k} - \begin{bmatrix} \tilde{\mathbb{T}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k}. \end{aligned} \quad (6.30)$$

On the other hand, we use (6.26) to get

$$V_{k+1}^{(2)} = U_{k+1}^{(1)}(A_{k+1}^{-1})^{(1,2)} + U_{k+1}^{(2)}(A_{k+1}^{-1})^{(2,2)}.$$

We pre-multiply the last equation on the left by A , and then we use (6.25) to obtain

$$\begin{aligned} AV_{k+1}^{(2)} &= AU_{k+1}^{(1)}(A_{k+1}^{-1})^{(1,2)} + AU_{k+1}^{(2)}(A_{k+1}^{-1})^{(2,2)} \\ &= AV_{k+1}^{(1)}A_{k+1}^{(1,1)}(A_{k+1}^{-1})^{(1,2)} + AU_{k+1}^{(2)}(A_{k+1}^{-1})^{(2,2)}. \end{aligned} \quad (6.31)$$

We pre-multiply on the left by \mathbb{W}_{2m+2}^T and we use (6.30) to get

$$\begin{aligned} \mathbb{W}_{2m+2}^T AV_{k+1}^{(2)} &= \mathbb{W}_{2m+2}^T AV_{k+1}^{(1)}A_{k+1}^{(1,1)}(A_{k+1}^{-1})^{(1,2)} + \left(\tilde{e}_{2k} - \begin{bmatrix} \tilde{\mathbb{T}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k} \right) (A_{k+1}^{-1})^{(2,2)} \\ &= \tilde{\mathbb{T}}_{2m}\tilde{e}_{2k+1}A_{k+1}^{(1,1)}(A_{k+1}^{-1})^{(1,2)} + \left(\tilde{e}_{2k} - \begin{bmatrix} \tilde{\mathbb{T}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k} \right) (A_{k+1}^{-1})^{(2,2)}. \end{aligned}$$

Then the proof is completed since we have $\mathbb{W}_{2m+2}^T AV_{k+1}^{(2)} = \tilde{\mathbb{T}}_{2m}\tilde{e}_{2k+2}$. Now we show the same result for the matrix $\tilde{\mathbb{L}}_{2m}$, and we derive recursions to compute the block columns of this matrices directly from the block columns of the upper block Hessenberg matrix $\tilde{\mathbb{H}}_{2m}$, without requiring the matrix-vector products with A^{-1} .

Proposition 6.1.4 *Let $\tilde{\mathbb{L}}_{2m}$ and $\tilde{\mathbb{H}}_{2m}$ be the upper Hessenberg matrices defined*

earlier. Then, the following relations hold

$$\tilde{\mathbb{L}}_{2m}\tilde{e}_1 = [\tilde{e}_1\Lambda_V^{(1,2)} + \tilde{e}_2\Lambda_V^{(2,2)}](\Lambda_V^{(1,1)})^{-1}, \quad (6.32)$$

and for $k = 1, \dots, m$, we have

$$\tilde{\mathbb{L}}_{2m}\tilde{e}_{2k} = \tilde{\mathbb{H}}_{2m}\tilde{e}_{2k}, \quad (6.33)$$

$$\tilde{\mathbb{L}}_{2m}\tilde{e}_{2k+1} = \left(\tilde{e}_{2k-1} - \begin{bmatrix} \tilde{\mathbb{L}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k-1} \right) (A_{k+1}^{(1,1)})^{-1}. \quad (6.34)$$

Proof. To prove (6.32), we use the QR decomposition of $[V, A^{-1}V]$ as in (6.29). Then if $\Lambda_V^{(1,1)}$ is non-singular, we obtain

$$A^{-1}V_1^{(1)} = A^{-1}V(\Lambda_V^{(1,1)})^{-1} = [V_1^{(1)}\Lambda_V^{(1,2)} + V_1^{(2)}\Lambda_V^{(2,2)}](\Lambda_V^{(1,1)})^{-1}.$$

We pre-multiply the above equality on the left by \mathbb{W}_{2m+2}^T and we use the biorthogonality condition to get

$$\mathbb{W}_{2m+2}^T A^{-1}V_1^{(1)} = [\tilde{e}_1\Lambda_V^{(1,2)} + \tilde{e}_2\Lambda_V^{(2,2)}](\Lambda_V^{(1,1)})^{-1}.$$

Then, equation (6.32) is obtained by using the fact that $\mathbb{V}_{2m+2}A^{-1}V_1^{(1)} = \tilde{\mathbb{L}}_{2m}\tilde{e}_1$.

For the other even block vectors, we proceed as follows. We start by using (6.24) and (6.25) to have

$$\begin{aligned} A^{-1}V_k^{(2)} &= U_{k+1}\tilde{e}_2 + \mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k}, \\ &= V_{k+1}A_{k+1}\tilde{e}_2 + \mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k} \\ &= \mathbb{V}_{2k+2}\tilde{\mathbb{H}}_{2k}\tilde{e}_{2k}. \end{aligned} \quad (6.35)$$

Now, multiplying on the left by \mathbb{W}_{2m+2}^T to get

$$\mathbb{W}_{2m+2}^T A^{-1}V_k^{(2)} = \mathbb{W}_{2m+2}^T \mathbb{V}_{2k+2}\tilde{\mathbb{H}}_{2k}\tilde{e}_{2k},$$

hence

$$\mathbb{W}_{2m+2}^T A^{-1}V_k^{(2)} = \begin{bmatrix} I_{2(k+1)p} \\ 0_{2(m-k)p \times 2(k+1)p} \end{bmatrix} \tilde{\mathbb{H}}_{2k}\tilde{e}_{2k}$$

therefore

$$\tilde{\mathbb{L}}_{2m}\tilde{e}_{2k} = \begin{bmatrix} \tilde{\mathbb{H}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} = \tilde{\mathbb{H}}_{2m}\tilde{e}_{2k},$$

which gives relation (6.33).

For the odd blocks, we multiply (6.24) on the left by A^{-1} and we consider only the first p columns of each block to obtain

$$A^{-1}U_{k+1}\tilde{e}_1 = V_k^{(1)} - A^{-1}\mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k-1}.$$

Since $U_{k+1} = V_{k+1}A_{k+1}$, we have

$$U_{k+1}\tilde{e}_1 = V_{k+1}^{(1)}A_{k+1}^{(1,1)},$$

if $A_{k+1}^{(1,1)}$ is non-singular, we obtain

$$A^{-1}V_{k+1}^{(1)} = A^{-1}U_{k+1}\tilde{e}_1(A_{k+1}^{(1,1)})^{-1} = (V_k^{(1)} - A^{-1}\mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k-1})(A_{k+1}^{(1,1)})^{-1}.$$

Multiplying from the left by \mathbb{W}_{2m+2}^T , we get

$$\mathbb{W}_{2m+2}^T A^{-1}V_{k+1}^{(1)} = (\mathbb{W}_{2m+2}^T V_k^{(1)} - \mathbb{W}_{2m+2}^T A^{-1}\mathbb{V}_{2k}\mathbb{H}_{2k}\tilde{e}_{2k-1})(A_{k+1}^{(1,1)})^{-1},$$

and then

$$\begin{aligned} \tilde{\mathbb{L}}_{2m}\tilde{e}_{2k+1} &= \left(\mathbb{W}_{2m+2}^T \mathbb{V}_{2m+2}\tilde{e}_{2k-1} - \mathbb{W}_{2m+2}^T A^{-1}\mathbb{V}_{2m} \begin{bmatrix} I_{2kp} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k-1} \right) (A_{k+1}^{(1,1)})^{-1} \\ &= \left(\tilde{e}_{2k-1} - \tilde{\mathbb{L}}_{2m} \begin{bmatrix} I_{2kp} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k-1} \right) (A_{k+1}^{(1,1)})^{-1} \\ &= \left(\tilde{e}_{2k-1} - \begin{bmatrix} \tilde{\mathbb{L}}_{2k} \\ 0_{2(m-k)p \times 2kp} \end{bmatrix} \mathbb{H}_{2k}\tilde{e}_{2k-1} \right) (A_{k+1}^{(1,1)})^{-1}, \end{aligned} \quad (6.36)$$

which gives the relation (6.34).

The results of the next two propositions will be used to prove other properties in the next section which is devoted to the application of the extended block Lanczos method to obtain reduced order models in large scale dynamical systems. As we will see, the method allow one to approximate low and high frequencies of the corresponding transfer function at the same time.

Proposition 6.1.5 *Let \mathbb{V}_{2m} and \mathbb{W}_{2m} be the matrices generated by Algorithm 10, and let $\mathbb{L}_{2m} = \mathbb{W}_{2m}^T A^{-1}\mathbb{V}_{2m}$. Then we have*

$$A^{-j}\mathbb{V}_{2m}\mathbb{E}_1 = \mathbb{V}_{2m}\mathbb{L}_{2m}^j\mathbb{E}_1, \quad \text{for } j = 0, \dots, m-1, \quad (6.37)$$

$$(A^{-T})^j\mathbb{W}_{2m}\mathbb{E}_1 = \mathbb{W}_{2m}(\mathbb{L}_{2m}^T)^j\mathbb{E}_1, \quad \text{for } j = 0, \dots, m-1. \quad (6.38)$$

Moreover, we have

$$\mathbb{T}_{2m}^{-1}\mathbb{E}_j = \mathbb{L}_{2m}\mathbb{E}_j, \quad \text{for } j = 1, \dots, m-1, \quad (6.39)$$

where \mathbb{E}_j is an $2mp \times 2p$ tall thin matrix with an identity matrix of dimension p at the j^{th} block and zero elsewhere.

Proof. Using equation (6.21) of proposition (6.1.2), we have

$$A^{-1}\mathbb{V}_{2m} = \mathbb{V}_{2m}\mathbb{L}_{2m} + V_{m+1}L_{m+1,m}\mathbb{E}_m^T, \quad (6.40)$$

we pre-multiply on the left by A^{-1} j times, we re-arrange the result and then we multiply from the right by \mathbb{E}_1 to get

$$A^{-j}\mathbb{V}_{2m}\mathbb{E}_1 = \mathbb{V}_{2m}\mathbb{L}_{2m}^j\mathbb{E}_1 + \sum_{i=1}^j A^{-(i-1)}V_{m+1}L_{m+1,m}\mathbb{E}_m^T\mathbb{L}_{2m}^{j-i}\mathbb{E}_1.$$

As \mathbb{L}_{2m} is an upper block Hessenberg matrix, it follows that $\mathbb{E}_m^T\mathbb{L}_{2m}^{j-i}\mathbb{E}_1 = 0$, for $j = 1, \dots, m-1$, and then equation (6.37) is verified. In a similar way, we can use equation (6.23) of proposition (6.1.2) to show (6.38).

Now to prove (6.39), we multiply (6.40) from the right by \mathbb{E}_j to get

$$A^{-1}\mathbb{V}_{2m}\mathbb{E}_j = \mathbb{V}_{2m}\mathbb{L}_{2m}\mathbb{E}_j, \text{ for } j = 1, \dots, m-1.$$

We pre-multiply the above equality by $\mathbb{W}_{2m}^T A$ from the left and we use the biorthogonality condition to have

$$\mathbb{E}_j = \mathbb{T}_{2m}\mathbb{L}_{2m}\mathbb{E}_j, \text{ for } j = 1, \dots, m-1.$$

Finally, equation (6.39) can be obtained if we assume that \mathbb{T}_{2m} is non-singular.

The following result is proven in [87], it gives a general property for two upper Hessenberg matrices.

Proposition 6.1.6 *Let $T = (T_{i,j})$ and $L = (L_{i,j})$ be two upper Hessenberg matrices with blocks $T_{i,j}, L_{i,j} \in \mathbb{R}^{p \times p}$ for $i, j = 1, \dots, m$, and suppose that*

$$T\mathbb{E}_j = L\mathbb{E}_j, \quad \text{for } j = 1, \dots, m-1.$$

Then

$$T^k\mathbb{E}_1 = L^k\mathbb{E}_1, \quad \text{for } k = 1, \dots, m-1.$$

6.2 Application to model reduction problem

We consider the Multi-Input Multi-Output LTI dynamical system Σ described in (2.1). Then the aim of this section is to present a new projection method that allows to compute the low-order dimensional system (2.3) by projecting the

original problem on to the extended Krylov subspace. The application of the extended block Lanczos algorithm, described in last section, to the pairs (A, B) and (A^T, C^T) gives two bi-orthogonal bases $\mathbb{V}_{2m} \in \mathbb{R}^{n \times 2mp}$ and $\mathbb{W}_{2m} \in \mathbb{R}^{n \times 2mp}$ and then the reduced order model can be defined as

$$A_{2m} = \mathbb{T}_{2m} = \mathbb{W}_{2m}^T A \mathbb{V}_{2m}, \quad B_{2m} = \mathbb{W}_{2m}^T B \quad \text{and} \quad C_{2m} = C \mathbb{V}_m. \quad (6.41)$$

As in the next chapters, we use the moment matching techniques to generate the reduced order model (2.3). We consider the Markov parameters

$$f_\infty^{(j)} = CA^j B, \quad j \geq 0,$$

and the j^{th} moment of $F(s)$ around $\sigma = 0$

$$f_0^{(j)} = CA^{-j} B, \quad j \geq 0.$$

Then, the aim of the moment matching problem using the extended block Lanczos algorithm is to produce a reduced order model such that $2m - 1$ moments are to be matched for the Markov parameters, and $2m - 1$ moments are also to be matched around zero, i.e.,

$$\hat{f}_\infty^{(j)} = f_\infty^{(j)}, \quad \text{for } j = 0, \dots, 2m - 2, \quad (6.42)$$

and

$$\hat{f}_0^{(j)} = f_0^{(j)}, \quad \text{for } j = 0, \dots, 2m - 2. \quad (6.43)$$

For the Markov parameters, the equality (6.42) is already proven in the literature; see [86] and the references therein. the following result shows that the first $2m - 1$ moments of the transfer function F around $\sigma = 0$ are also matched.

Proposition 6.2.1 *Let $\hat{f}_0^{(j)}$ and $f_0^{(j)}$ be the matrix moments given by the Laurent expansions of the transfer functions F_m and F around $\sigma = 0$, respectively. Then, the first $2m - 1$ moments of the original and the reduced models are the same, that is,*

$$\hat{f}_0^{(j)} = f_0^{(j)}, \quad \text{for } j = 0, \dots, 2(m - 1).$$

Proof. Let $j \in \{0, 1, \dots, 2(m - 1)\}$, let $j_1, j_2 \in \{0, 1, \dots, m - 1\}$ such that $j_1 + j_2 = j$. Then we have

$$f_0^{(j)} = CA^{-j} B = CA^{-(j_1+j_2)} B = CA^{-j_1} A^{-j_2} B. \quad (6.44)$$

Using the QR decomposition defined in (6.2) for $V = B$ and $W = C^T$ gives

$$B = V_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T W_1^T.$$

Substituting this result in equation (6.44) yields

$$\begin{aligned} f_0^{(j)} &= \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T W_1^T A^{-j_1} A^{-j_2} V_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T \mathbb{E}_1^T \mathbb{W}_{2m}^T A^{-j_1} A^{-j_2} \mathbb{V}_{2m} \mathbb{E}_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix}. \end{aligned} \quad (6.45)$$

Therefore, using the result of Proposition 6.1.5, we get

$$\begin{aligned} f_0^{(j)} &= \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T \mathbb{E}_1^T \mathbb{L}_{2m}^{j_1} \mathbb{W}_{2m}^T \mathbb{V}_{2m} \mathbb{L}_{2m}^{j_2} \mathbb{E}_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T \mathbb{E}_1^T \mathbb{L}_{2m}^j \mathbb{E}_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix}. \end{aligned} \quad (6.46)$$

On the other hand, since \mathbb{L}_{2m} and \mathbb{T}_{2m}^{-1} are both Hessenberg matrices that verify $\mathbb{T}_{2m}^{-1} \mathbb{E}_j = \mathbb{L}_{2m} \mathbb{E}_j$, then the application of Proposition 6.1.6 gives

$$\mathbb{L}_{2m}^j \mathbb{E}_1 = \mathbb{T}_{2m}^{-j} \mathbb{E}_1, \quad \text{for } j = 0, \dots, m-1,$$

and so

$$\begin{aligned} f_0^{(j)} &= \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T \mathbb{E}_1^T \mathbb{T}_{2m}^{-j} \mathbb{E}_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \Lambda_W^{(1,1)} \\ 0 \end{bmatrix}^T W_1^T \mathbb{V}_{2m} \mathbb{T}_{2m}^{-j} \mathbb{W}_{2m}^T V_1 \begin{bmatrix} \Lambda_V^{(1,1)} \\ 0 \end{bmatrix} \\ &= C \mathbb{V}_{2m} \mathbb{T}_{2m}^{-j} \mathbb{W}_{2m}^T B \\ &= C_r \mathbb{T}_{2m}^{-j} B_r \\ &= \hat{f}_0^{(j)}, \end{aligned} \quad (6.47)$$

which completes the proof of Proposition 6.2.1.

6.3 Numerical experiments

In this section, we give some experimental results to show the effectiveness of the extended block Lanczos algorithm proposed when applied to reduce the order of large scale dynamical systems. All the experiments were performed on a computer of Intel Core i5 at 1.3GHz and 8GB of RAM. The algorithms were coded in Matlab 8.0.

In the experiments of this chapter, we used some matrices from LYAPACK and different known benchmark models listed in Table 6.1. The subspace dimension will be fixed a-priori for all examples.

Table 6.1: The matrix tests.

Matrices	sizes
CD-Player	$n = 120, p = 2$
add32	$n = 4960, p = 4$
Modified FOM	$n = 1006, p = 5$
ISS	$n = 270, p = 3$
fdm	$n = 10.000, p = 6$
Flow-Meter	$n = 9669, p = 5$

Example 1. In this example, we used the extended block Lanczos algorithm to reduce the order of ISS and CD player models. They are small dimension systems but are generally difficult and are always considered as a benchmark test . The top curves of Figure 6.1 (ISS) and Figure 6.2 (CD player) show the frequency responses of the original system (circles) compared with the frequency responses of its approximations for $m = 5$. The bottom curves of these figures represent the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for different frequencies $\omega \in [10^{-6}, 10^6]$.

As a third test model of example 1, we considered the Flow model which is obtained from the discretization of a 2D convective thermal flow problem (flow meter model v0.5) from the Oberwolfach model reduction benchmark collection,

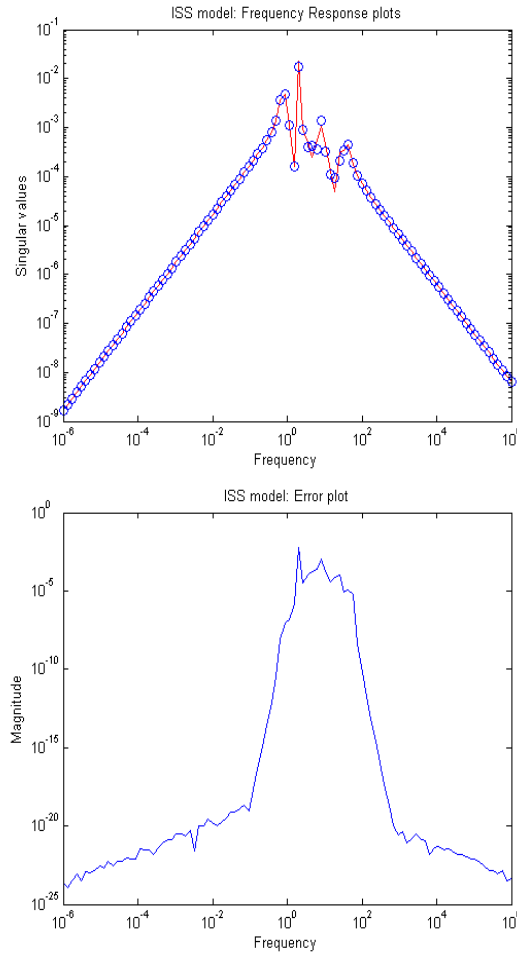


Figure 6.1: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the ISS model.

2003, with 5 inputs and 5 outputs. The obtained plots of this experiment are given in Figure 6.3.

Example 2. In this example, We plotted the \mathcal{H}_∞ error norm $\|F - F_m\|_\infty$ versus the number m of iterations for two different models. The first one is the modified FOM model, while the second is the `fdm` system. For the `fdm` model we have

$$f(x, y) = e^{xy}, \quad g(x, y) = \sin(xy) \quad \text{and} \quad h(x, y) = y^2 - x^2,$$

the dimension of the original system is $n = 10^4$ with 6 inputs and 6 outputs. As

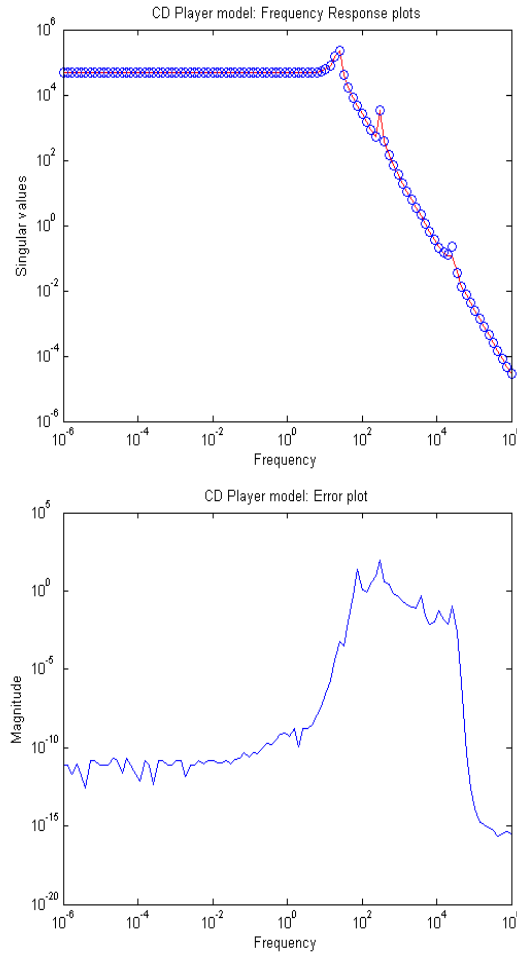


Figure 6.2: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the CD player model.

can be shown from Figure 6.4, the extended block Lanczos algorithm (EBLA) gives good result with small values of m .

Example 3. In the last example we compared the extended block Lanczos algorithm (EBLA) with IRKA method. We used four models: the ISS, the add32, the Modified fom and the fdm models. In Table 6.2, we listed the obtained \mathcal{H}_∞ norm of the error transfer function $\|F - F_m\|_\infty$, the corresponding cpu-time, and the used space dimension. A maximum number of $m_{max} = 100$ iterations was

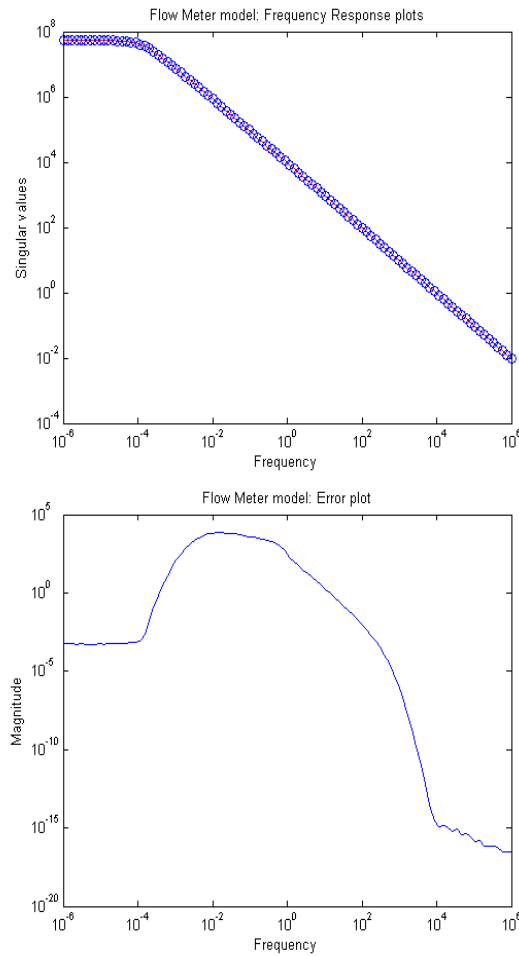


Figure 6.3: Top: $\|F(j\omega)\|_2$ and its approximations $\|F_m(j\omega)\|_2$. Bottom: the exact error $\|F(j\omega) - F_m(j\omega)\|_2$ for the flow-meter model.

allowed to the IRKA method. As observed from Table 6.2, IRKA and EBLA return similar results for the \mathcal{H}_∞ norm, with an important advantage of the cpu-time for the EBLA.

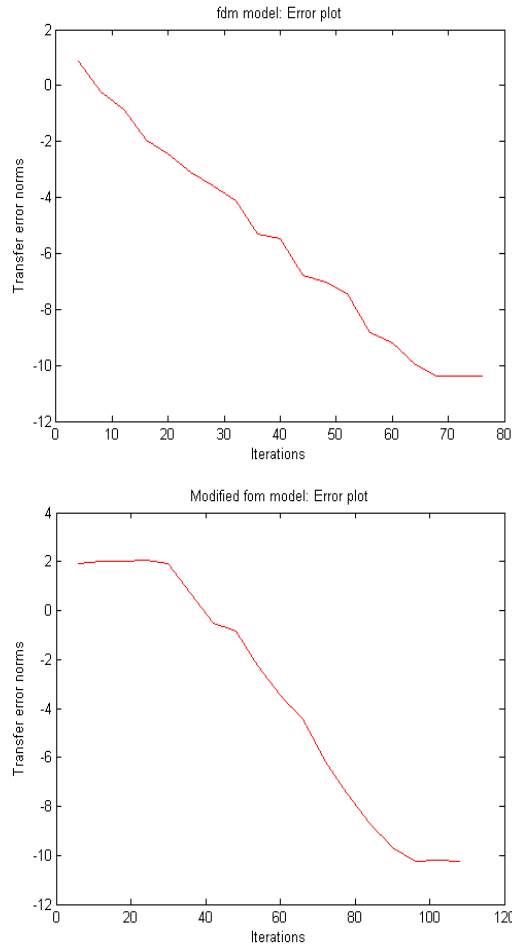


Figure 6.4: The \mathcal{H}_∞ error $\|F - F_m\|_\infty$ versus the number of iterations for the `fdm` model (top curve) and the `modified fom` model (bottom curve).

6.4 Conclusion

In this chapter, we proposed an extended block Lanczos algorithm with applications in model order reduction of MIMO first-order stable linear dynamical systems. Moreover, we derived new theoretical results and new properties for this process. We presented some numerical results to confirm the good performance of the extended block Lanczos subspace method compared with other known method. The proposed procedure is tested on well known benchmark problems of medium

Table 6.2: Comparison between IRKA and EBLA for ISS, add32, Modified fom and fdm models.

	EBLA			IRKA		
	\mathcal{H}_∞ error	# dim($2mp$)	time	\mathcal{H}_∞ error	# dim(mp)	time
ISS	8.41e-04	60	0.25s	1.61e-04	75	12.71s
add32	1.96e-08	120	4.72s	8.65e-09	120	313.77s
Modified fom	3.89e-11	216	1.61s	3.60e-10	150	78.66s
fdm	6.07e-11	180	20.93s	3.75e-11	150	1000.60s

and large dimensions, and the numerical results show that the application of the extended algorithm on model reduction problem allows one to obtain reduced order models of small dimension.

7

Conclusions and Future Directions

This chapter provides a summary of the results on model order reduction established in the previous chapters. Moreover, suggestions for future directions on the improvements of the aforementioned techniques and extension of these results will be stated.

7.1 Summary of results

This dissertation has focused on projection methods to efficiently generated reduced order models for large scale linear dynamical systems with Multiple-Input Multiple-Output, especially, the moment matching techniques based on multi-point rational interpolation. Rational Krylov has been shown to be very effective for large-scale systems and produce better approximations over a broad frequency

range by matching the moments of the original transfer function $F(s)$ around different frequencies. Unfortunately, these methods have the drawback that the selection of interpolation points is a difficult task since it is an ad-hoc process.

The main contribution of all the chapters of this thesis may be summarized in four parts:

- Proposing a new variant of rational Lanczos algorithm.
- Deriving some algebraic properties that describe the proposed process.
- Introducing new techniques for choosing the interpolation points.
- Giving new expressions for the transfer function error.

Then in this thesis, four rational Lanczos algorithms are proposed and applied to model order reduction problem. The first one is named the rational block Lanczos-type algorithm and it is related to a set of rational equations that describe the relation between the two bi-orthogonal bases generated by this process and the matrices of the original system. The combination of the first algorithm and an adaptive approach for choosing the interpolation points gives an Adaptive Order Rational Block Lanczos-type (AORBL) algorithm. The numerical results confirm the good performance of the rational block Lanczos subspace method, especially when compared with the well known approach IRKA.

The second algorithm, named the modified rational block Lanczos algorithm, can be considered as a generalization of the first one where different multiplicities are consider for each interpolation point. The advantage of this procedure is that the standard Lanczos equations remain valid also in the rational block case.

An other extension of the standard Krylov subspace method for MIMO systems is the global Krylov subspace. This algorithm compute F -biorthogonal bases of the rational matrix Krylov subspaces. In this part, two versions of the rational Global Lanczos algorithm are given. The first one represents the general form of this process which allows to obtain the global Lanczos equations but in the

rational form. In the second version, we modify the first rational global Lanczos algorithm in such a way the standard Lanczos equations remain valid also in the rational global case. Next, we combine the modified rational global Lanczos process and one of the proposed method to select the interpolation points to get an Adaptive Modified Rational Global Lanczos (AMRGL) procedure.

The last procedure proposed in this thesis is the extended block Lanczos algorithm (EBLA). The advantage of this process compared with other rational Krylov procedures is that we don't need to construct the poles of the rational Krylov subspace to compute the F -biorthogonal bases.

7.2 Future directions

The goal of this dissertation was to provide efficient rational Krylov algorithms for model order reduction problem. Some recommendations for future studies related to this work are presented below:

- Proposing other rational Krylov algorithms.
- Applying the rational Lanczos algorithms proposed to approximate the problem of $e^{tA}B$.
- Developing efficient algorithms based on balanced truncation method and rational Krylov algorithms.
- Developing new and efficient techniques for choosing the interpolation points.

Bibliography

- [1] O. ABIDI, M. HACHED, AND K. JBILOU, Adaptive rational block Arnoldi methods for model reductions in large-scale MIMO dynamical systems, *New Trends in Mathematical Sciences*, 4(2)(2016), pp. 227–239.
- [2] O. ABIDI, M. HEYOUNI, AND K. JBILOU, On some properties of the extended block and global Arnoldi methods with applications to model reduction, *Numer. Algorithms.*, 2016 to appear.
- [3] A.C. ANTOULAS, *Approximation of Large-Scale Dynamical Systems*, *Advances in Design and Control*, SIAM, Philadelphia, 2005.
- [4] A.C. ANTOULAS, AND J.C. WILLEMS, A behavioral approach to linear exact modeling, *IEEE Trans. Automat. Contr.*, AC-38(1993), pp. 1776–1802.
- [5] A.C. ANTOULAS, On recursivness and related topics in linear systems, *IEEE Trans. Automat. Contr.*, AC-31(1986), pp. 1121–1135.
- [6] A.C. ANTOULAS, Recursive modeling of discrete-time time series, in *IMA volume on Linear Algebra for Control*, P. van Dooren and B. W. Wyman Editors, Springer Verlag, 62(1993), pp. 1–20.
- [7] W.E. ARNOLDI, the principle of minimized iterations in the solution of the matrix eigenvalue problem, *Quart. Appl. Math.*, 9(1951), pp. 17–29.

- [8] Z. BAI, D. DAY, AND Q. YE, ABLE: an Adaptive Block Lanczos Method for Non-Hermitian Eigenvalue Problems, *SIAM J. Mat. Anal. Appl.*, 20(1999), pp. 1060–1082.
- [9] Z. BAI, J. DEMMEL, J. DONGARRA, A. RUHE, AND H. VAN DER VORST, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, Philadelphia, 2000.
- [10] Z. BAI, AND R. FREUND, A Partial Padé-via-Lanczos Method for Reduced-Order Modeling, *Lin. Alg. Appl.*, 332-334(2001), pp. 139–164.
- [11] Z. BAI, R.D. SLONE, W.T. SMITH, AND Q. YE, Error bounds for reduced system model by Padé approximation via the Lanczos process, *IEEE Trans. Comput. Aided Design.*, 18(2)(1999), pp. 133–141.
- [12] J. BASTIAN, AND J. HAASE, Order Reduction of Second Order Systems, In *Proc. 4th Mathmod.*, 2003, pp. 418–424, Vienna.
- [13] H. BARKOUKI, A.H. BENTBIB, AND K. JBILOU, An adaptive rational block Lanczos-type algorithm for model reduction of large scale dynamical systems, *J. Sci. Comput.*, 67(2016), pp. 221–236.
- [14] H. BARKOUKI, A.H. BENTBIB, AND K. JBILOU, A modified nonsymmetric rational block Lanczos method for model reduction in large scale LTI dynamical systems, *Applicationes Mathematicae*, 2016 to appear.
- [15] H. BARKOUKI, A. H. BENTBIB, AND K. JBILOU, A matrix rational Lanczos method for model reduction in large scale first and second order dynamical systems. *Numer. Linear Algebra Appl.*, 2016 to appear.
- [16] C.A. BEATTIE, AND S. GUGERCIN, Krylov-based model reduction of second-order systems with proportional damping, *Proceedings of the 44th IEEE Conference on Decision and Control.*, (2005), pp. 2278–2283.
- [17] M.A. BOTCHEV, V. GRIMM, AND M. HOCHBRUCK, Residual, restarting, and Richardson iteration for the matrix exponential, *SIAM J. Sci. Comput.*, 35(2013), pp. A1376–A1397.

- [18] C.A. BEATTIE, AND S. GUGERCIN, Krylov-based minimization for optimal \mathcal{H}_2 model reduction, in Proceedings of the 46th IEEE Conference on Decision and Control., (2007), pp. 4385-4390.
- [19] R. BHATIA, A note on Lyapunov equations, Linear Algebra App., 259(1997), pp. 71–76.
- [20] A. BODENDIEK, AND M. BOLLHÖFER, Adaptive-order rational Arnoldi-type methods in computational electromagnetism, BIT Numer. Math., 54(2014), pp. 357–380.
- [21] R. BOUYOULI, K. JBILOU, R. SADAKA, AND H. SADOK, Convergence properties of some block Krylov subspace methods for multiple linear systems, J. Comput. Appl. Math., (2006), pp. 498–511.
- [22] C. BREZINSKI, The block Lanczos and Vorobyev methods, C.R. Acad. Sci. Paris, Série I, 331(2000), pp. 137–142.
- [23] C. BREZINSKI, Model reduction by state space projection, Commun. Appl. Anal., 8(2004), pp. 323–333.
- [24] C. BREZINSKI, Padé-Type Approximation and General Orthogonal Polynomials, ISNM Vol. 50, Birkhauser- Verlag, Basel, 1980.
- [25] C. BREZINSKI, Computational Aspects of Linear Control, Kluwer, Dordrecht, 2002.
- [26] A. BULTHEEL, AND B. DE MOOR, Rational approximation in linear systems and control, J. Comput. Appl. Math., 121(2000), pp. 355–378.
- [27] Y. CHAHLAOUI, K.A. GALLIVAN, A. VANDENDORPE, AND P. VAN DOOREN, Model Reduction of Second-Order System. P. Benner, V. Mehrmann, and D. Sorensen, eds., Vol. 45 of Lecture Notes in Computational Science and Engineering Springer-Verlag, Berlin/Heidelberg, Germany, (2005), 149–172.

- [28] Y. CHAHLAOU, D. LEMONNIER, A. VANDENDORPE, AND P. VAN DOOREN, Second-Order Balanced Truncation, *Lin. Alg. Appl.*, (2-3)(415)(2006), pp. 373–384.
- [29] C.C. CHU, M.H. LAI, AND W.S. FENG, The multiple point global Lanczos method for multiple-inputs multiple-outputs interconnect order reduction, *IECE Trans. Elect.*, E89-A (2006), pp. 2706–2716.
- [30] C.C. CHU, M.H. LAI, AND W.S. FENG, MIMO interconnects order reductions by using the multiple point adaptive-order rational global Arnoldi algorithm, *IECE Trans. Elect.*, E89-C(2006), pp. 792–802.
- [31] C.C. CHU, M. LAI, AND W.S. FENG, Lyapunov-based error estimations of MIMO interconnect reductions by using the global Arnoldi algorithm, *IEICE Trans. Electron.*, E90-A(2)(2007), pp. 415–418.
- [32] C.C. CHU, M. LAI, AND W.S. FENG, Model-order reductions for MIMO systems using global Krylov subspace methods, *Math. Comput. Simulation.*, 79(2008), pp. 1153–1164.
- [33] C.K. CHUI, AND G. CHEN, *Kalman filtering*, 3rd ed., Springer-Verlag, 1999.
- [34] T. CHIU, Model reduction by the low-frequency approximation balancing methods for unstable systems, *IEEE trans. Automat. Control.*, AC-41(7)(1996), pp. 995–997.
- [35] B. DATTA, M. HEYOUNI, AND K. JBILOU, The global Arnoldi process for solving the Sylvester-Observer equation, *Comput. Appl. Math.*, 29(3)(2010), pp. 527–544.
- [36] C. DE VILLEMAGNE, AND R. SKELTON, Model reduction using a projection formulation, *Int J. Control.*, 40(1987), pp. 2141–2169.
- [37] V. DRUSKIN, AND L. KNIZHNERMAN, Krylov subspace approximation of eigenpairs and matrix functions in exact and computer arithmetic, *Numer. Linear Algebra Appl.*, 2(1995), pp. 205–217.

- [38] V. DRUSKIN, AND L. KNIZHNERMAN, Two polynomial methods of calculating functions of symmetric matrices, *USSR Comput. Maths. Math. Phys.*, 29(1989), pp. 112–121.
- [39] V. DRUSKIN, AND L. KNIZHNERMAN, Extended Krylov subspaces: Approximation of the matrix square root and related functions, *SIAM J. Matrix Anal. Appl.*, 19(1998), pp. 755–771.
- [40] V. DRUSKIN, C. LIEBERMAN, AND M. ZASLAVSKY, On adaptive choice of shifts in rational Krylov subspace reduction of evolutionary problems, *SIAM J. Sci. Comput.*, 32(5)(2010), pp. 2485–2496.
- [41] V. DRUSKIN, V. SIMONCINI, AND M. ZASLAVSKY, Adaptive Tangential Interpolation in Rational Krylov Subspaces for MIMO Dynamical Systems, *SIAM J. Matrix Anal. Appl.*, 35(2)(2014), pp. 476–498.
- [42] V. DRUSKIN, V. SIMONCINI, AND M. ZASLAVSKY, Adaptive rational Krylov subspaces for large-scale dynamical systems, *Syst. Control Lett.*, 60(2011), pp. 546–560.
- [43] V. DRUSKIN, AND V. SIMONCINI, Adaptive rational Krylov subspaces for large-scale dynamical systems, *Syst. Contr. Lett.*, 60(8)(2011), pp. 546–560.
- [44] R. EID, B. SALIMBAHRAMI, B. LOHMANN, E. RUDNYI, AND J. KORVINK, Parametric order reduction of proportionally damped second-order systems, *Sensor Mater.*, 19(2007), pp. 149–164.
- [45] M. EIERMANN, AND O. ERNST, A restarted Krylov subspace method for the evaluation of matrix functions, *SIAM J. Numer. Anal.*, 44(2006), pp. 2481–2504.
- [46] D. ENNS, Model reduction with balanced realizations: An error bound and a frequency weighted generalization, in the *Proceedings of the 23d IEEE Conf. Decision and Control*, 1984.
- [47] B.F. FARRELL, AND P.J. LOANNOU, State estimation using a reduced order Kalman Filter, *Journal Atmospheric Sciences*, 58(2001), pp. 3666–3680.

- [48] P. FELDMAN, AND R.W. FREUND, Efficient linear circuit analysis by Padé approximation via a Lanczos method, *IEEE Trans. Comput. Aided Des.*, 14(1995), pp. 639–649.
- [49] L. FORTUNA, G. NUNNARI, AND A. GALLO, *Model Order Reduction Techniques with Applications in Electrical Engineering*, Springer-Verlag London, 1992.
- [50] M. FRANGOS, AND I.M. JAIMOUKHA, Adaptive rational interpolation: Arnoldi and Lanczos-like equations, *Eur. J. Control.*, 14(4)(2008), pp. 342–354.
- [51] M. FRANGOS, AND I.M. JAIMOUKHA, Adaptive rational Krylov algorithms for model reduction, In *Proc. European Control Conference*, (2007), pp. 4179–4186.
- [52] R.W. FREUND, Dimension Reduction of Higher Order Systems via Krylov Subspace Techniques. In *BIRS workshop on Model Reduction Problems and Matrix Methods*, Banff, Alberta, Canada, April 2004.
- [53] R.W. FREUND, SPRIM: Structure-Preserving Reduced-Order Interconnect Macromodeling, In *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design.*, (2004), pp. 80–87.
- [54] R. FRIESNER, L. TUCKERMAN, B. DORNBLASER, AND T. RUSSO, A method for exponential propagation of large systems of stiff nonlinear differential equations, *SIAM J. Sci. Comput.*, 4(1989), pp. 327–354.
- [55] A. FROMMER, K. KAHL, T. LIPPERT, AND H. RITTICH, 2-norm error bounds and estimates for Lanczos approximations to linear systems and rational matrix functions, *SIAM J. Matrix Anal. Appl.*, 34(2013), pp. 1046–1065.
- [56] A. FROMMER, AND V. SIMONCINI, Stopping criteria for rational matrix functions of Hermitian and symmetric matrices, *SIAM J. Sci. Comput.*, 30(2008), pp. 1387–1412.

- [57] K. GALLIVAN, A. VANDENDORPE, AND P. VANDOOREN, Sylvester equations and projection-based model reduction, *J. Comp. Appl. Math.*, 162(2004), pp. 213–229.
- [58] K. GALLIVAN, E. GRIMME, AND P. VAN DOOREN, Padé Approximation of Large-Scale Dynamic Systems with Lanczos Methods, In Proceedings of the 33rd IEEE Conference on Decision and Control, (1994), pp. 443–448.
- [59] K. GALLIVAN, E. GRIMME, AND P. VAN DOOREN, A rational Lanczos algorithm for model reduction, *Num. Algebra.*, 12(1996), pp. 33–63.
- [60] K. GALLIVAN, P. VAN DOOREN, AND E. GRIMME, On some recent developments in projection-based model reduction, in *ENUMATH 97* (Heidelberg), World Scientific, River Edge, NJ, 1998, pp. 98–113.
- [61] K. GALLIVAN, E. GRIMME, D. SORENSEN, AND P. VAN DOOREN, On some modifications of the Lanczos algorithm and the relation with Padé approximations, *Mathematical Research Series*, Akademie Verlag, Berlin, 7(1996), pp 87–116.
- [62] E. GALLOPOULOS, AND Y. SAAD, Efficient solution of parabolic equations by polynomial approximation methods, *SIAM J. Stat. Sci. Comput.*, 13(1992), pp. 1236–1264.
- [63] E. GALLOPOULOS, AND Y. SAAD, Parallel solution of parabolic equations using polynomial approximations to the exponential, Technical report, Research Institute for Advanced Computer Science, Technical report number 90–14. Submitted.
- [64] E. GALLOPOULOS, AND Y. SAAD, On the parallel solution of parabolic equations, In R. De Groot, editor, *Proceedings of the International Conference on Supercomputing 1989*, Heraklion, Crete, June 5–9, 1989. ACM press, 1989.
- [65] W. GAWRONSKI, AND J.-N. JUANG, Model reduction in limited time and frequency intervals, *Int. J. Systems Sci.*, 21(2)(1990), pp. 349–376.

- [66] K. GLOVER, All optimal Hankel-norm approximations of linear multivariable systems and their L_∞ -errors bounds, *Internat. J. Control.*, 39(1984), pp. 1115–1193.
- [67] K. GLOVER, D.J.N. LIMEBEER, J.C. DOYLE, E.M. KASENALLY, AND M.G. SAFONOV, A characterisation of all solutions to the four block general distance problem, *SIAM J. Control Optim.*, 29(1991), pp. 283–324.
- [68] G.H. GOLUB, AND R. UNDERWOOD, The block Lanczos method for computing eigenvalues, *Mathematical Software III*, J. R. Rice, ed., Academic Press, New York, (1977), pp.361–377.
- [69] W.B. GRAGG, AND A. LINDQUIST, On the partial realization problem, *Lin. Alg. Appl.*, 50(1983), pp. 277–319.
- [70] E. GRIMME, K. GALLIVAN, AND P. VAN DOOREN, Rational Lanczos algorithm for model reduction II: interpolation point selection, Technical report, University of Illinois at Urbana Champaign (1998).
- [71] E. GRIMME, Krylov Projection methods for model reduction, PhD thesis, ECE Department, University of Illinois, Urbana-Champaign (1997).
- [72] S. GUGERCIN, A.C. ANTOULAS, AND C.A. BEATTIE, A Rational Krylov iteration for optimal \mathcal{H}_2 model reduction, in *Proceedings of Mathematical Theory of Networks and Systems*, 2006.
- [73] S. GUTTEL, Rational Krylov Methods for Operator Functions, PhD thesis, Institut für Numerische Mathematik und Optimierung der Technischen Universität Bergakademie Freiberg, 2010.
- [74] S. GUGERCIN, AND A.C. ANTOULAS, Approximation of the International Space Station 1R and 12A models, in the *Proceedings of the 40th CDC*, December 2001.
- [75] S. GUGERCIN, Projection methods for model reduction of large-scale dynamical systems, Ph.D. Dissertation, ECE Dept., Rice University, December 2002.

- [76] M.H. GUTKNECHT, A completed theory of the unsymmetric Lanczos process and related algorithms, Part I. SIAM J. Matrix Anal. Appl., 13(1992), pp. 594–639.
- [77] M.H. GUTKNECHT, A completed theory of the unsymmetric Lanczos process and related algorithms, Part II. SIAM J. Matrix Anal. Appl., 15(1994), pp. 15–58.
- [78] M. HEYOUNI, K. JBILOU, A. MESSAOUDI, AND K. TABAA, Model reduction in large scale MIMO dynamical systems via the block Lanczos method, Comput. Appl. Math., 27(2)(2008), pp. 211–236.
- [79] M. HEYOUNI, The global Hessenberg and CMRH methods for linear systems with multiple right-hand sides, Numer. Algorithms., 26(4)(2001), pp. 317–332.
- [80] M. HEYOUNI, AND K. JBILOU, An extended block Arnoldi algorithm for large-scale solutions of the continuous-time algebraic Riccati equation, Elect. Trans. Num. Anal., 33(2009), pp. 53–62.
- [81] M. HEYOUNI, AND K. JBILOU, Matrix Krylov subspace methods for large scale model reduction problems, Appl. Math. Comput., 181(2006), pp. 1215–1228.
- [82] M. HOCHBRUCK, AND C. LUBICH, Exponential integrators for quantum-classical molecular dynamics, BIT, Numerical Mathematics, 39(1999), pp. 620–645.
- [83] M. HOCHBRUCK, C. LUBICH, AND H. SELHOFER, Exponential integrators for large systems of differential equations, SIAM J. Sci. Comput., 19 (1998), pp. 1552-1574.
- [84] I.M. JAIMOUKHA, AND E.M. KASENALLY, Oblique projection methods for large scale model reduction, SIAM J. Matrix Anal. Appl., 16(1995), pp. 602–627.

- [85] K. JBILOU, A survey of Krylov-based methods for model reduction in large-scale MIMO dynamical systems, *Applied and Computational Mathematics*, 15(2)(2016), pp. 24–42.
- [86] K. JBILOU, A. MESSAOUDI, AND H. SADOK, Global FOM and GMRES algorithms for matrix equations, *Appl. Numer. Math.*, 31(1999), pp. 49–63.
- [87] K. JBILOU, H. SADOK, AND A. TINZEFTE, Oblique projection methods for linear systems with multiple right-hand sides, *Electr. Trans. Numerical Analysis.*, 20(2005), pp. 119–138.
- [88] K. JBILOU, AND A. RIQUET, Projection methods for large Lyapunov matrix equations, *Linear Algebra Appl.*, 415(2-3)(2006), pp. 344–358.
- [89] R.E. KALMAN, Contributions to the theory of optimal control, *Bol. Soc. Mat. Mex.*, 5(1960), pp. 102–119.
- [90] R.E. KALMAN, On the general theory of control systems, *Proc. 1st International Congress on Automatic Control, Moscow, 1960*, Butterworths, London, 1(1961), pp. 481–492.
- [91] C. KENNY, AND G. HEWER, Necessary and sufficient conditions for balancing unstable systems, *IEEE Trans. Automat. Contr.*, AC-32(2)(1987), pp. 157–160.
- [92] H.M. KIM, AND R.R. CRAIG JR, Structural dynamics analysis using an unsymmetric block Lanczos algorithms, *Internat. J. Numer. Methods Engrg*, 26(1988), pp. 2305–2318.
- [93] H.M. KIM, AND R.R. CRAIG JR, Computational enhancement of an unsymmetric block Lanczos algorithms, *Internat. J. Numer. Methods Engrg*, 30(1990), pp. 1083–1089.
- [94] L. KNIZHNERMAN, V. DRUSKIN, AND M. ZASLAVSKY, On optimal convergence rate of the rational Krylov subspace reduction for electromagnetic problems in unbounded domains, *SIAM J. Numer. Anal.*, 47(2)(2009), pp. 953–971.

- [95] C. LANCZOS, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, *J. Res. Nat. Bur. Stand.*, 45(1950), pp. 255–282.
- [96] A.J. LAUB, M.T. HEATH, C.C. PAIGE, AND R.C. WARD, Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms, *IEEE Trans. Automat. Control.*, 32(1987), pp. 115–122.
- [97] H.J. LEE, C.C. CHU, AND W.S. FENG, An adaptive-order rational Arnoldi method for model-order reductions of linear time-invariant systems, *Linear Algebra Appl.*, 415(2006), pp. 235–261.
- [98] R.-C. LI, Structural Preserving Model Reductions, Tech. Rep. 04–02, University of Kentucky, Lexington, 2004.
- [99] C.-A LIN, AND T.-Y CHIU, Model reduction via frequency weighted balanced realization, *Control Theory and Advanced Technol.*, 8(1992), pp. 341–351.
- [100] B. LOHMANN, AND B. SALIMBAHRAMI, Ordnungsreduktion mittels Krylov-Unter-Raummethoden, *Automatisierungstechnik*, 52(1)(2004), pp. 30–38.
- [101] B. LOHMANN, AND B. SALIMBAHRAMI, Reduction of Second Order Systems using Second Order Krylov subspaces, In *IFAC world congress*, Prag, Czech Rep., 2005.
- [102] L. LOPEZ, AND V. SIMONCINI, Analysis of projection methods for rational function approximation to the matrix exponential, *SIAM J. Numer. Anal.*, 44(2006), pp. 613–635.
- [103] K. MEERBERGEN, Dangers in changing the poles in the rational Lanczos method for the Hermitian eigenvalue problem, ral-tr-1999-025, Rutherford Appleton Laboratory, Chilton, UK, 1999.

- [104] V. MEHRMANN, The autonomous Linear Quadratic Control Problem, theory and Numerical Solution, Number in Lecture Notes in Control and Information Sciences, Springer-Verlag, Heidelberg (1991).
- [105] D.G. MEYER, AND S. SRINIVASAN, Balancing and Model Reduction for Second-Order Form Linear Systems, IEEE Trans. Aut. Control., 41(11)(1996), pp. 1632–1644.
- [106] C. MOLER, AND C. VAN LOAN, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later, SIAM Review, 45(2003), pp. 3–49.
- [107] B.C. MOORE, Principal component analysis in linear systems: controllability, observability and model reduction, IEEE Trans. Automatic Contr., AC-26(1981), pp. 17–32.
- [108] I. MORET, AND P. NOVATI, RD-rational approximations of the matrix exponential, BIT, 44(2004), pp. 595–615.
- [109] C.T. MULLIS, AND R.A. ROBERTS, IEEE Trans. Acoust. Speec Signal Process., 24(1976).
- [110] V. MEHRMANN, AND T. PENZL, *Benchmark collections in SLICOT*. Technical Report SLWN1998- 5, SLICOT Working Note, ESAT, KU Leuven, K. Mercierlaan 94, Leuven-Heverlee 3100, Belgium, 1998. Available under <http://www.win.tue.nl/niconet/NIC2/reports.html>
- [111] T.V. NGUYEN, AND JR. LI, Multipoint Padé Approximation using a rational block Lanczos algorithm, in Proc. IEEE/ACM Intl. Conf. Computer-Aided Design, (1997), pp. 72–75.
- [112] G. OBINATA, AND B.D.O. ANDERSON, Model Reduction for Control System Design, Springer, 2000.
- [113] T. PENZL, *LYAPACK*-A MATLAB toolbox for Large Lyapunov and Riccati Equations, Model Reduction Problems, and Linear-quadratic Optimal Control Problems, <http://www.tu-chemintz.de/sfb393/lyapack>.

- [114] M. POPOLIZIO, AND V. SIMONCINI, Acceleration techniques for approximating the matrix exponential operator, *SIAM J. Matrix Anal. Appl.*, 30(2008), pp. 657–683.
- [115] V. POPOV, Hyperstability and optimality of automatic systems with several control functions, *Rev. Roum. Sci. Tech.*, 9(1964), pp. 629–690.
- [116] A. PREUMONT, Vibration control of active structures, An Introduction, second ed., Kluwer, Dordrecht, 2002.
- [117] D. RAMASWAMY, AND J. WHITE, Automatic Generation of Small-Signal Dynamic Macromodel from 3-D Simulation, In *Proc. Int. Conf. Modeling and Simulation of Microsystems*, South Carolina, U.S.A., 2001, pp. 27–30.
- [118] M.F. RUBINSTEIN, Structural systems-statics, dynamics and stability, Prentice Hall, Inc, 1970.
- [119] E.B. RUDNYI, J. LIENEMANN, A. GREINER, AND J.G. KORVINK, mor4ansys: Generating Compact Models Directly from ANSYS Models, In *Proc. NSTI-Nanotech*, 2(2004), pp. 279–282.
- [120] A. RUHE, Rational Krylov algorithms for nonsymmetric eigenvalue problems II: matrix pairs, *Linear Alg. Appl.*, 197(1994), pp. 283–295.
- [121] A. RUHE, Rational Krylov sequence methods for eigenvalue computations, *Linear Algebra and Appl.*, 58(1984), pp. 391–405.
- [122] A. RUHE, Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils, *SIAM J. Sci. Comput.*, 19(1998), pp. 1535–1551.
- [123] Y. SAAD, Analysis of some Krylov subspace approximations to the matrix exponential operator, *SIAM J. Numer. Anal.*, 29(1992), pp. 209–228.
- [124] J. SAAK, Efficient Numerical Solution of Large Scale Algebraic Matrix Equations in PDE Control and Model Order Reduction, PhD thesis, TU Chemnitz, July 2009.

- [125] M.G. SAFONOV, AND R.Y. CHIANG, A Schur method for balanced-truncation model reduction, *IEEE Trans. Automat. Control.*, 34(1989), pp. 729–733.
- [126] M.G. SAFONOV, AND R.Y. CHIANG, A Schur method for balanced-truncation model reduction, *IEEE Trans. Automat. Control.*, 34(1989), pp. 729–733.
- [127] B. SALIMBAHRAMI, AND B. LOHMANN, Structure Preserving Order Reduction of Large Scale Second Order Systems, In *IFAC Symp. Large Scale Systems: Theory and App.*, Osaka, Japan, (2004), pp. 245–250.
- [128] B. SALIMBAHRAMI, AND B. LOHMANN, Order Reduction of Large Scale Second Order Systems Using Krylov Subspace Methods, *Linear Algebra Appl.*, 415(2006), pp. 385–405.
- [129] B. SALIMBAHRAMI, Structure Preserving Order Reduction of Large Scale Second Order Models, Ph.D. Thesis, Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen Universität München.
- [130] B.N. SHEEHAN, ENOR: Model Order Reduction of RLC Circuits using Nodal Equations for Efficient Factorization, In *Proc. 36th ACM/IEEE conference on Design Automation*, 1999.
- [131] V. SIMONCINI, A new iterative method for solving large-scale Lyapunov matrix equations, *SIAM J. Sci. Comput.*, 29(2007), pp. 1268–1288.
- [132] V. SIMONCINI, The extended Krylov subspace for parameter dependent systems, *Appl. Numer. Math.*, 60(2010), pp. 550–560.
- [133] V. SIMONCINI, D.B. SZYLD, AND M. MONSALVE, On two numerical methods for the solution of large-scale algebraic Riccati equations, *J. Numer. Anal.*, 34(2014), pp. 904–920.
- [134] A. SOPPA, Krylov-Unterraum basierte Modellreduktion zur Simulation von Werkzeugmaschinen, PhD thesis, University Braunschweig, 2011.

- [135] T.J. SU, AND R.R. CRAIG JR, Model reduction and control of flexible structures using Krylov vectors, *J. Guid. Contr. Dyn.*, 14(1991), pp. 260–267.
- [136] C.P. THERAPOS, Balancing transformation for unstable nonminimal linear systems, *IEEE Trans. Automat. Contr.*, 34(4)(1989), pp. 455–457.
- [137] L. TREFETHEN, AND M. GUTKNECHT, The Carathéodory-Fejér method for real rational approximation, *SIAM J. Numer. Anal.*, 20(1983), pp. 420–436.
- [138] L. TREFETHEN, J. WEIDEMAN, AND T. SCHMELZER, Talbot quadratures and rational approximations, *BIT*, 46(2006), pp. 653–670.
- [139] E.E. TYRTYSHNIKOV, Mosaic-skeleton approximations, *Calcolo*, 33(1996), pp. 47–57.
- [140] P. VAN DOOREN, The Lanczos algorithm and Padé approximations, Short Course, Benelux Meeting on Systems and Control, 1995.
- [141] A. VANDENDORPE, AND P. VAN DOOREN, Krylov techniques for model reduction of second-order systems, Tech. Rep. 07-2004, CESAME, Université catholique de Louvain, 2004.
- [142] W. WEAVER JR, AND P.R. JOHNSTON, Structural dynamics by finite elements, Prentice Hall, Inc, 1987.
- [143] P.M.R. WORTELBOER, M. STEINBUCH, AND O.H. BOSGRA, Iterative Model and Controller Reduction Using Closed-Loop Balancing, with Application to a Compact Disc Mechanism, *Int. J. Robust and Nonlinear Control*, 9(1999), pp. 123–142.
- [144] G. WU, AND Y. WEI, On analysis of projection methods for rational function approximation to the matrix exponential, *SIAM J. Numer. Anal.*, 48(2010), pp. 191–197.
- [145] A. YOUSUFF, AND R.E. SKELTON, Linear system approximation via covariance equivalent realizations, *J. Math. Anal. App.*, 196(1985), pp. 91–115.

Bibliography

- [146] A. YOUSUFF, AND R.E. SKELTON, Covariance equivalent realizations with applications to model reduction of large-scale systems, In: Leondes, C.T. (ed.) Control and Dynamic Systems, Academic Press., 22(1985), pp. 273–348.
- [147] K. ZHOU, AND J.C. DOYLE, Essentials of robust control, Prentice Hall, 1998.
- [148] K. ZHOU, Frequency-weighted L_1 norm and optimal Hankel norm model reduction, IEEE Trans. Automat. Contr., 40(10)(1995), pp. 1687–1699.
- [149] K. ZHOU, G, SALOMON, AND E. WU, Balanced realization and model reduction for unstable systems, Int. J. Robust Nonlinear Control, 9(1999), pp. 183–198.

Publications

List of publications

- H. BARKOUKI, A. H. BENTBIB, AND K. JBILOU, An adaptive rational block Lanczos-type algorithm for model reduction of large scale dynamical systems, *J. Sci. Comput.*, 67(2016), pp. 221–236.
- H. BARKOUKI, A. H. BENTBIB, AND K. JBILOU, A modified nonsymmetric rational block Lanczos method for model reduction in large scale LTI dynamical systems, *Applicationes Mathematicae*, (2016) to appear.
- H. BARKOUKI, A. H. BENTBIB, AND K. JBILOU, A matrix rational Lanczos method for model reduction in large scale first and second order dynamical systems. *Numerical Linear Algebra with Applications*, (2016) to appear.

Submitted papers

- H. BARKOUKI, A. H. BENTBIB, AND K. JBILOU, An extended block Lanczos algorithm for model reduction of large scale dynamical MIMO systems, submitted to publication.

List of talks

- H. Barkouki, A. H. Bentbib, and K. Jbilou, "An adaptive rational block Lanczos-type algorithm for large scale model reduction problems". Congès international Modélisation et Calcul Scientifique pour l'Ingénierie Mathématiques (MOCASIM), 19-22 Novembre 2014, Marrakech Morocco.
- H. Barkouki, A. H. Bentbib, and K. Jbilou, "A modified nonsymmetric rational block Lanczos method for model reduction in large scale LTI dynamical systems". 4 ème conférence international de la SM2A, 12-14 Février 2015, Oujda Morocco.
- H. Barkouki, A. H. Bentbib, and K. Jbilou, "A matrix rational Lanczos method for model reduction in large scale MIMO systems". Modélisation Mathématique et Informatique et Simulation Numérique, 18-19 Mai 2015, Errachidia Morocco.
- H. Barkouki, A. H. Bentbib, and K. Jbilou, "A rational Krylov subspace method for the computation of the matrix exponential $e^{tA}B$ ". 1 ère Journée Doctoriale MOCASIM, 11-12 Février 2016, Marrakech Morocco.
- H. Barkouki, A. H. Bentbib, and K. Jbilou, "A matrix rational Lanczos method for model reduction in large scale first and second order dynamical systems". Journées d'Analyse Numérique et d'Optimisation (Jano'11), 27-29 Avriile 2016, Beni Mellal Morocco.