Méthodes de sous-espaces de Krylov rationnelles pour le contrôle et la réduction de modèles

Oussama Abidi

THÈSE DE DOCTORAT

Présentée à

L'Université Littoral Côte d'Opale

Spécialité Mathématiques Appliquées

par

ABIDI OUSSAMA

Méthodes de sous-espaces de Krylov rationnelles pour le contrôle et la réduction de modèles

Directeur: **JBILOU KHALIDE**

Soutenue le 08 Décembre 2016

Membres du Jury:

Mme. ROSIER CAROLE,	Professeur, Université du Littoral Côte d'Opale	Présidente
M. VANDEBRIL RAF,	Professeur, Université KU Leuven, Belgique	Rapporteur
M. JEAN-PAUL CHEHAB,	Professeur, Université de Picardie Jules Verne	Rapporteur
M. JBILOU KHALIDE,	Professeur, Université du Littoral Côte d'Opale	Directeur

LMPA, Centre Universitaire de la Mi-Voix

50, rue Ferdinand Buisson

CS 80699 - 62228 Calais Cedex - France

Remerciements

Je tiens en premier lieu à remercier mon directeur de thèse M. Khalide JBILOU pour son encadrement, sa disponibilité et son aide considérable tout au long de mes années de thèse.

Je tiens à remercier M. Jean-Paul CHEHAB et M. Raf VANDEBRIL pour avoir accepté d'être rapporteurs de mes travaux de thèse et pour leurs observations qui m'ont permis d'améliorer la qualité de ce mémoire. Je tiens à leur exprimer mes remerciements pour l'honneur qu'ils me font en participant à ce jury.

Mes sincère remerciements et ma gratitude vont aussi à Mme Carole ROSIER pour avoir accepté de juger ce travail et d'en présider le jury.

Un grand merci à tous les membres du LMPA qui ont été toujours à mes côtés par leur aide.

Je remercie tous mes amis et proches, qui ont contribué de près ou de loin à l'accomplissement de cette thèse.

Mes remerciements les plus profonds vont naturellement à tous les membres de ma famille, qui m'ont soutenu constamment durant toutes ces longues années d'études.

Table des matières

Introduction **1**

0.0.1 Introduction 1

0.0.2 Motivation 5

1 Introduction aux systèmes dynamiques **9**

1.1 Systèmes dynamiques linéaires 9

1.2 Fonctions de transfert et Moments 10

1.2.1 Fonction de transfert 10

1.2.2 Moments d'une fonction de transfert 12

1.3 Concepts fondamentaux 13

1.3.1 Stabilité, contrôlabilité et observabilité 13

1.3.2 Les Gramians de contrôlabilité et d'observabilité 14

1.4 Différentes normes des systèmes dynamiques 17

1.4.1 La norme $\mathcal{H}_2$ 17

1.4.2 La norme $\mathcal{H}_\infty$ 18

1.4.3 La norme de Hankel 19

2 Rational block Arnoldi methods for model reduction in large-scale MIMO dynamical systems **21**

2.1 Introduction 21

2.2 The rational block Arnoldi method 23

2.3 An adaptive computation of the shifts 25

2.3.1 An *a priori* selection of the shifts 26

2.3.2 A new adaptive selection of the shifts 26

2.3.3 An error expression for the transfer function 28

2.4 A modified rational block Arnoldi algorithm 29
2.5 Numerical experiments 32
2.6 Conclusion 35

3 Global rational Arnoldi method for model reduction 37
3.1 Introduction 37
3.2 Preliminaries 39
3.3 The rational global Arnoldi algorithm 40
3.4 The modified adaptive rational global Arnoldi method 43
3.5 Model reduction, transfer functions and adaptive selection of the shifts 46
3.5.1 The reduced model 46
3.5.2 Criterion-selections of the shift parameters 48
3.6 A modified two-sided global rational Arnoldi method 49
3.7 Numerical experiments 53
3.8 Conclusion 55

4 On some properties of the extended block and global Arnoldi methods with applications to model reduction 57
4.1 Introduction 57
4.2 Some algebraic properties on the extended block and global Arnoldi processes 60
4.2.1 The block case 60
4.2.2 The global case 65
4.3 Application for model reduction techniques 70
4.4 Numerical tests 75
4.5 Conclusion 79

5 Balanced Truncation-Rational Krylov methods for model reduction in large scale dynamical systems 81
5.1 Introduction 81
5.2 Lyapunov-balanced truncation 83
5.2.1 The transfer function 83
5.2.2 Controllability and Observability Gramians 84
5.2.3 Lyapunov balanced truncation 85

5.3 The rational block Arnoldi method for solving large Sylvester matrix equations 88
5.4 The Riccati-balanced truncation method 93
5.4.1 The LQG-Riccati method for model reduction 93
5.4.2 The rational block Arnoldi for continuous-time algebraic Riccati equations 95
5.5 Numerical experiments 97
5.6 Conclusion 100

6 The Extended-Rational Krylov method **101**
6.1 A new Krylov subspace method 101
6.2 Model reduction, transfer function 107
6.3 Numerical examples 107
6.4 Conclusion 109

Conclusion and perspectives **109**

List of Algorithms

2.1 The Rational Block Arnoldi Algorithm 23
2.2 Modified Rational Arnoldi Algorithm 30
3.1 The Rational Global Arnoldi (RGA) Algorithm 41
3.2 The Modified Adaptive Rational Global Arnoldi (MARGA) Algorithm 44
4.1 The extended block Arnoldi algorithm 60
4.2 The extended global Arnoldi algorithm 65
5.1 The Rational block Arnoldi Algorithm (RBA) 89
5.2 The rational block Arnoldi algorithm for Sylvester equations (RBAS) 94
5.3 The rational block Arnoldi algorithm for CAREs (RBA-CARE) . . 97
6.1 The extended-rational block Arnoldi algorithm 102

Résumé

Beaucoup de phénomènes physiques sont modélisés par des équations aux dérivées partielles, la discrétisation de ces équations conduit souvent à des systèmes dynamiques (continus ou discrets) dépendant d'un vecteur de contrôle dont le choix permet de stabiliser le système dynamique. Comme ces problèmes sont, dans la pratique, de grandes tailles, il est intéressant de les étudier via un autre problème dérivé réduit et plus proche du modèle initial.

Dans cette thèse, on introduit et on étudie de nouvelles méthodes basées sur les processus de type Krylov rationnel afin d'extraire un modèle réduit proche du modèle original. Des applications numériques seront faites à partir de problèmes pratiques.

Après un premier chapitre consacré au rappel de quelques outils mathématiques, on s'intéresse aux méthodes basées sur le processus d'Arnoldi rationnel par blocs pour réduire la taille d'un système dynamique de type Multi-Input/Multi-Output (MIMO). On propose une sélection adaptative de choix de certains paramètres qui sont cruciaux pour l'efficacité de la méthode. On introduit aussi un nouvel algorithme adaptatif de type Arnoldi rationnel par blocs afin de fournir une nouvelle relation de type Arnoldi.

Dans la deuxième partie de ce travail, on introduit la méthode d'Arnoldi rationnelle globale, comme alternative de la méthode d'Arnoldi rationnel par blocs. On définit la projection au sens global, et on applique cette méthode pour approcher les fonctions de transfert.

Dans la troisième partie, on s'intéresse à la méthode d'Arnoldi étendue (qui est un cas particulier de la méthode d'Arnoldi rationnelle) dans les deux cas (global et par blocs), on donnera quelques nouvelles propriétés algébriques qui sont appliquées aux problèmes des moments.

On considère dans la quatrième partie la méthode de troncature balancée pour la réduction de modèle. Ce procédé consiste à résoudre deux grandes équations algébriques de Lyapunov lorsque le système est stable ou à résoudre deux équations de Riccati lorsque le système est instable. Comme ces équations sont de grandes tailles, on va appliquer la méthode de Krylov rationnel par blocs pour approcher la solution de ces équations.

Le travail de cette thèse sera clôturé par une nouvelle idée, dans laquelle on définit un nouvel espace sous le nom de sous espace de Krylov rationnel étendu. On introduit la méthode de Krylov rationnelle étendue qui sera utilisée pour la réduction du modèle.

Abstract

Many physical phenomena are modeled by PDEs. The discretization of these equations often leads to dynamical systems (continuous or discrete) depending on a control vector whose choice can stabilize the dynamical system. As these problems are, in practice, of a large size, it is interesting to study the problem through another one which is reduced and close to the original model.

In this thesis, we develop and study new methods based on rational Krylov-based processes for model reduction techniques in large-scale Multi-Input Multi-Output (MIMO) linear time invariant dynamical systems.

In Chapter 2 the methods are based on the rational block Arnoldi process to reduce the size of a dynamical system through its transfer function. We provide an adaptive selection choice of shifts that are crucial for the effectiveness of the method. We also introduce a new adaptive Arnoldi-like rational block algorithm to provide a new type of Arnoldi's relationship.

In Chapter 3, we develop the new rational global Arnoldi method which is considered as an alternative to the rational block Arnoldi process. We define the projection in the global sense, and apply this method to extract reduced order models that are close to the large original ones. Some new properties and applications are also presented.

In Chapter 4 of this thesis, we consider the extended block and global Arnoldi methods. We give some new algebraic properties and use them for approaching the firt moments and Markov parameters in moment matching methods for model reduction techniques.

In Chapter 5, we consider the method of balanced truncation for model reduction. This process is based on the solutions of two major algebraic equations : Lyapunov equations when the system is stable or Riccati equations when the system is

unstable. Since these equations are of large sizes, we will apply the rational block Arnoldi method for solving these equations.

In Chapter 6, we introduce a new method based on a new subspace called the extended-rational Krylov subspace. We introduce the extended-rational Krylov method which will be used for model reduction in large-scale dynamical systems.

Introduction

0.0.1 Introduction

De nombreux phénomènes physiques, souvent décrits par des systèmes d'équations aux dérivées partielles, sont modélisés par des systèmes dynamiques linéaires invariants dans le temps ("*Linear time Invariant*" **LTI**). Ces derniers peuvent prendre la forme d'une équation différentielle algébrique du type

$$\begin{cases} \dot{x}(t) &= A\,x(t) + B\,u(t) \\ y(t) &= C\,x(t), \end{cases} \tag{1}$$

où $A \in \mathbb{R}^{n\times n}$, $B,\ C^T \in \mathbb{R}^{n\times p}$, $x(t) \in \mathbb{R}^n$ le vecteur d'état, $u(t) \in \mathbb{R}^p$ le vecteur de sortie et $y(t) \in \mathbb{R}^p$ le vecteur d'entré du système (1).

Ces modèles sont obtenus aussi soit par discrétisation (éléments finis, différences finies...) des équations aux dérivées partielles en variables spatiales, soit à partir d'une linéarisation d'un système non-linéaire.

Le système dynamique linéaire LTI est beaucoup utilisé dans le domaine du contrôle, de la simulation et de l'ingénierie. Cependant, ces types de systèmes qui sont dérivés en général des problèmes réels sont souvent trop compliqués à étudier et à traiter même avec la grande puissance des machines dont on dispose actuellement en raison du grand nombre de variables d'état.

Les stratégies habituellement mises en place visent à remplacer le système originel par un système de taille réduite. Ce modèle réduit doit conserver autant que possible les propriétés du modèle dont il dérive. Le but donc est de donner un modèle sous cette forme

$$\begin{cases} \dot{x}_m(t) &= A_m\,x_m(t) + B_m\,u(t) \\ y_m(t) &= C_m\,x_m(t), \end{cases} \tag{2}$$

tels que $A_m \in \mathbb{R}^{m\times m}$, $B_m,\ C_m^T \in \mathbb{R}^{m\times p}$ et $x_m(t), y_m(t) \in \mathbb{R}^m$, avec $m \ll n$, tout en respectant ces conditions

1. L'erreur (et/ou le résidu) entre le modèle approché et celui à partir duquel il est construit doit être petite.
2. Le modèle approché doit conserver les propriétés les plus pertinentes du système original.
3. L'algorithme de calcul doit être rapide et robuste.

Plusieurs approches ont été utilisées comme l'approximation de type Padé [30, 80], "balanced truncation" [70], "optimal Hankel norm" [35, 36] et les méthodes de sous espaces de Krylov [24, 25, 32, 52]. Ces dernières sont des méthodes de projection et ont joué un rôle central dans les réductions des modèles de grandes tailles ; voir [9, 21, 33].
Le sous espace de Krylov standard est défini par

$$\mathcal{K}_m(A, B) = \text{Range}\{B, AB, \ldots, A^{m-1}B\}.$$

En projetant les matrices du système sur cet espace, il est possible d'obtenir un système réduit suffisamment précis de taille inférieure.

D'autres variantes, jugées préférables [16, 27, 28, 32, 34] ont été analysées. La plus générale est donnée par les sous espaces de Krylov rationnels définis par

$$\mathbf{K}_m(A, B) = \text{Range}\{B, (A - s_2 I)^{-1}B, ..., \prod_{i=1}^{m}(A - s_i I)^{-1}B\}, \tag{3}$$

où $s_2, \ldots, s_m$ sont des nombres complexes choisis.

La méthode de sous-espaces de Krylov rationnels a été introduite par Ruhe [74] pour résoudre les problèmes de valeurs propres, puis utilisée pour la construction des techniques de réduction de modèles [37] dans laquelle le choix de certains paramètres 'shifts' est crucial pour la qualité de l'approximation. Dans la réduction de modèle, le rôle des sous espaces de Krylov rationnels est un peu différent, car ils sont particulièrement bien adaptés pour l'approximation de la fonction de transfert sur l'axe imaginaire.

En effet l'espace de Krylov rationnel est reconnu comme un outil puissant dans les techniques de réduction de l'ordre de modèle pour les systèmes dynamiques linéaires. Cependant, son succès a été entravé par quelques problèmes, comme le choix de shifts qui sont utilisés pour construire l'espace, ainsi que, des identités

connues sous le nom de relations d'Arnoldi ont été utilisées pour des majorations d'erreurs, des calculs de normes de résidus, des tests d'arrêt et pour effectuer une analyse des perturbations.

Les sous-espaces de Krylov rationnels ont été utilisés aussi dans des autres applications comme l'approximation de fonctions de matrices. Dans un tel contexte le but est d'approcher d'une manière efficace l'action de $f(M)$ sur un vecteur v où f est une fonction scalaire et $M \in \mathbb{C}^{n\times n}$. Ces types de problèmes apparaissent dans de nombreuse applications comme la résolution des équations aux dérivées partielles ainsi qu'en théorie de contrôle et en physique des particules [50]. Cette thèse peut être considérée comme une étude d'un cas particulier de fonctions de matrices où f est une fonction rationnelle. Des travaux sur l'utilisation des méthodes basées sur les sous-espaces de Krylov rationnels ont été publiés dans la thèse de Güttel [43] et son papier [44].

Une autre alternative qui permet d'approcher un système de grande taille est la méthode POD (Proper orthogonal decomposition). L'objectif principal de cette méthode est l'extraction d'informations de l'espace à partir d'un ensemble de données collectées sur un intervalle de temps et dans un domaine spatial. Ce qui revient donc à obtenir une base optimale de faible dimension pour représenter des données expérimentales ou de simulation ayant initialement une très grande dimension. Cette base peut être obtenue en calculant un ensemble des espaces propres. La base de dimension réduite peut donc être utilisée pour formuler des modèles de taille réduite, par exemple des modèles décrivant des fluides complexes. Plus précisément POD décompose un champ fluctuant donné $u'(x,t)$ en un système orthogonal de mode spacials $u_i(x)$ aux quels correspondent des coefficients temporels $a_i(t)$

$$u'(x,t) = \sum_{i=1}^{N} a_i(t)u_i(x).$$

La méthode POD fournit non seulement des bases orthogonales de vecteurs propres mais elle permet également une mesure quantitative de l'importance relative de chacun de ces vecteurs. Cet aspect de la méthode la rend très utile dans l'analyse, l'identification et la réduction des systèmes dynamiques. La version discrète de la méthode POD est la méthode de décomposition en valeurs singulières SVD. Cette méthode est beaucoup utilisée en mécanique de fluides et en turbulence et en générale à des systèmes non linéaires. Par contre l'implémentation de cette méthode peut être très coûteuse en termes de temps et de place mémoire. Pour plus des détails sur

cette méthode le lecteur peut se référer aux articles suivants [14, 20, 22, 86] et [8] pour une comparaison entre cette méthode à celles de type sous-espaces de Krylov.

Le but de cette thèse est d'exploiter les méthodes de projection sur les sous espaces de Krylov rationnels afin de produire des modèles d'ordre réduit en respectant les conditions mentionnées auparavant.

Cette thèse comporte six chapitres. Le premier chapitre est une introduction générale aux systèmes dynamiques où l'on présentera les notions fondamentales utiles.

Dans le deuxième chapitre, on considère le processus d'Arnoldi par blocs pour approcher les systèmes dynamiques linéaires de types MIMO (multi input multi output). On présentera aussi un algorithme d'Arnoldi rationnel par blocs modifié et on établira des nouvelles relations de type Arnoldi rationnel par blocs qui seront utiles pour la majoration en norme de l'erreur de la fonction de transfert. Un choix approprié de shifts sera aussi proposé dans ce chapitre.

Une autre alternative au processus d'Arnoldi par blocs à été beaucoup utilisée sous le nom d'algorithme d'Arnoldi global pour résoudre les systèmes linéaires multiples et ensuite pour les équations du Lyapunov, Sylvester ou Riccati.
Dans le troisième chapitre, on proposera un algorithme adaptatif dit Arnoldi global rationnel et cela dans deux cas classique et modifié, afin de l'appliquer pour la réduction de modèles. On s'intéressera aussi aux questions relatives aux choix des shifts et aux relations de types Arnoldi. Ce travail sera étendu au cas bi-Arnoldi rationnel dont le but d'approcher le système dynamique d'une manière plus efficace.

Dans le quatrième chapitre, on étudiera la méthode d'Arnoldi étendue. Théoriquement, les sous espaces de Krylov étendus sont des cas particuliers des sous espaces de Krylov rationnel en alternant les shifts entre zéro et l'infini. La méthode d'Arnoldi étendue est considérée comme un outil puissant pour la réduction d'ordre de modèles ; de plus elle présente l'avantage de ne pas nécessiter le calcul de shifts. Le but de ce travail est de donner de nouvelles propriétés algébriques de l'algorithme d'Arnoldi étendu globale et par blocs. Ces propriétés seront utilisées pour montrer qu'un certain nombre de moments et de paramètres de Markov de la fonction de transfert approchée coïncident avec ceux d'origine.

Dans le cinquième chapitre, on considèrera la méthode de troncature balancée pour la réduction de modèles dynamique LTI dans le cas de systèmes MIMO. La méthode nécessite la résolution de deux équations matricielles couplées de Lyapunov lorsque le système est stable ou de Riccati lorsque le système est instable. En

utilisant la méthode d'Arnoldi rationnelle par blocs, on montre comment approcher les solutions de ces équations tout en établissant une écriture simplifiée de résidu. Les solutions approchées sont obtenues sous une forme factorisée et seront utilisées pour construire un modèle d'ordre réduit.

Dans le dernier chapitre de cette thèse, on s'intéresse encore à la réduction de l'ordre de modèles par les méthodes de projection, en introduisant une nouvelle méthode qui sera appliquée à la réduction de la fonction de transfert. L'idée générale de cette méthode est de fournir un nouvel espace de Krylov plus riche que le sous espace de Krylov rationnel et le sous espace de Krylov étendu. Cette idée vient de l'absence d'informations sur la matrice A pour le sous espace de Krylov rationnel. À cette fin, on introduit le sous espace de Krylov rationnel étendu, dans lequel on projette le problème afin de réduire son ordre.

0.0.2 Motivation

Dans cette sous section on va donner quelques exemples d'applications dans lesquelles les systèmes dynamiques à grande échelle se posent. Ces types de systèmes peuvent être utilisés pour la simulation, la prédiction de futur comportement et pour le contrôle. Pour plus d'exemples voir [7, 78].

Réacteurs chimiques : Contrôle de la température des réactifs

L'exemple suivant est le 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 d'entrée de certains réactifs dans un réacteur chimique. Les équations du modèle sont :

$$\begin{aligned} \frac{\partial X}{\partial t} - \kappa \Delta X + v.\nabla X &= 0 && \text{sur} \quad \Omega \\ X &= X_0, && \text{sur} \quad \Gamma_{in} \\ \frac{\partial X}{\partial n} &= \sigma(u - X) && \text{sur} \quad \Gamma_{heat1} \cup \Gamma_{heat2} \\ \frac{\partial X}{\partial n} &= 0 && \text{sur} \quad \Gamma_{out} \end{aligned}$$

Ici Ω désigne le domaine rectangulaire représenté sur la figure 1.
Le flux entrant Γ_{in} est du coté gauche du domaine, et le flux sortant Γ_{out} est à la

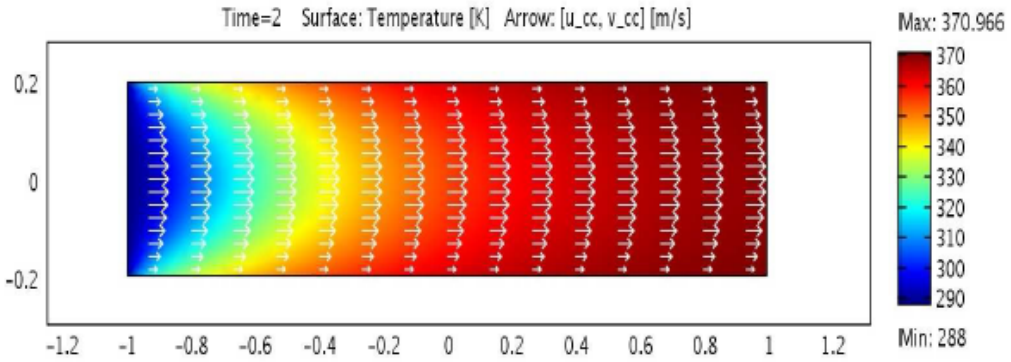

FIGURE 1 – Le domaine Ω : une coupure transversale en 2 dimensions d'un flux de liquide à l'intérieur d'un tube rond.

frontière droite. Nous pouvons nous limiter à ce domaine à 2 dimensions en assumant une symétrie rotationnelle ce qui est équivalent à assumer un écoulement non-turbulent. Les matrices tests ont été crées 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 du l'écoulement du flux à la sortie. Notons que pour ce cas nous avons un domaine convexe nous permettant d'évaluer en des points les sorties.

Puisque une discrétisation par éléments finis de l'espace a été utilisée ici, le modèle semi-discret s'écrit sous la forme :

$$\begin{aligned} M\dot{x} &= \tilde{A}x + \tilde{B}u \\ y &= \tilde{C}x. \end{aligned}$$

En décomposant M en $M = M_L M_U$ ce système peut être transformé sous la forme standard comme (1).

Système vibrationnel/système acoustique

Considérons un pare-prise (d'une voiture) soumis 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-prise constitué d'un matériau donné. La discrétisation par éléments finis donne 7564 nœuds (3 couches de $60 * 30$ éléments), pour un matériau

constitué de verre avec un module de Young égale à 7.10^7 N/m^2, une densité de 2490 kg/m^3 et un facteur de poisson de 0.23. Ces paramètres aident à déterminer expérimentalement les coefficients du modèle élément fini. Enfin le pare-prise subit une force en un point donné et l'objectif est donc de calculer le déplacement de ce point. Le problème discrétisé a une dimension de 22 692. Notons que cet exemple mène à une équation du second ordre de type

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

où X est la position, $\frac{d}{dt}X$ est la vitesse du pare-prise au point choisi. M, C et K sont respectivement la masse, l'amortissement et les matrices de raideur. Comme ce système est de second ordre, sa complexité est supérieure (45 384 états).

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

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

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

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

On suppose que ce système est influencé par les perturbations, en particulier on suppose que (i) les entrées aléatoires affectent toutes les variables $\hat{\phi}_i$ et (ii) toutes ces variables sont observables. Le système discrétisé est donc un système linéaire ayant même nombre d'entrée m, des vecteurs d'état n, et des sorties p. i.e,

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

Des modèles ainsi décrits sont utilisés pour traquer les tempêtes dans les moyennes altitudes de l'océan pacifique [29].

CHAPITRE 1

INTRODUCTION AUX SYSTÈMES DYNAMIQUES

Dans ce chapitre on exposera quelques définitions et propriétés importantes des systèmes dynamiques linéaires. Plus précisément on s'intéressera aux bases de la théorie du système dynamique invariant par le temps. On rappelle la fonction de transfert et les moments qui jouent un rôle important dans la réduction de modèle. Ensuite on définira les Gramains de contrôlabilité et d'observabilité en rappelant les résultats les plus importants. La dernière section sera consacrée aux différentes normes utilisées. Pour plus de détails voir les références [7, 79].

1.1 Systèmes dynamiques linéaires

Un système dynamique linéaire continu peut être exprimé sous forme d'une équation différentielle algébrique :

$$\begin{cases} \dot{x}(t) &= A(t)\,x(t) + B(t)\,u(t) \\ y(t) &= C(t)\,x(t) + D(t)\,u(t). \end{cases} \tag{1.1}$$

Le système (1.1) s'obtient à partir d'un modèle linéaire ou d'une linéarisation d'un système non-linéaire. Lorsque les coefficients matriciels $(A(t), B(t), C(t), D(t))$ dans (1.1) ne dépendent pas du temps ou ne varient pas beaucoup sur des périodes du temps, alors on peut les remplacer par des coefficients constants, ce qui donne lieu à un système dynamique invariant en temps ("*Linear time Invariant*" LTI)

$$\begin{cases} \dot{x}(t) &= A\,x(t) + B\,u(t) \\ y(t) &= C\,x(t) + D\,u(t), \end{cases} \tag{1.2}$$

où $A \in \mathbb{R}^{n\times n}$, $B,\ C^T \in \mathbb{R}^{n\times p}$, $x(t) \in \mathbb{R}^n$ le vecteur d'état, $u(t) \in \mathbb{R}^p$ le vecteur de sortie et $y(t) \in \mathbb{R}^p$ le vecteur d'entrée du système (1.2). Le système est dit mono-entrée/mono-sortie si $p = 1$, et multi-entrées/multi-sorties ou multi-variable sinon. On utilisera les abréviations anglaises SISO (single-input/single-output) pour le cas mono-entrée/mono-sortie et MIMO (multi-input/multi-output) pour le cas multi-entrées/multi-sorties. Un système dynamique LTI peut être noté aussi comme

$$\mathbf{\Sigma} \equiv \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]. \tag{1.3}$$

Tout au long de cette thèse, on ne considère que les systèmes LTI. Cette dernière description du système linéaire est appelée description interne, elle utilise à la fois le vecteur d'entrée $u(t)$ et le vecteur d'état $x(t)$ pour le vecteur de sortie $y(t)$. Une autre caractérisation du système linéaire sous le nom de description externe peut être écrite comme suit

$$y(t) = h \star u := \int_{-\infty}^{+\infty} h(t-\tau)u(\tau)d\tau, \tag{1.4}$$

où $h(t)$ est noté "Kernel" ou "weighting pattern" du système $\mathbf{\Sigma}$. La fonction $h(t)$ est appelée "impulse response" quand $u(t) = \delta(t)$ la fonction Delta de Dirac, et dans ce cas là $y(t) = h(t)$. Notons que tout système LTI peut être représenté par une convolution avec un choix convenable de $h(t)$. Cette dernière description ne fait intervenir que le vecteur d'entrée $u(t)$ pour le vecteur de sortie $y(t)$ et cela via la fonction $h(t)$ qui dépend bien évidemment des coefficients matriciels (A, B, C, D). Par exemple la fonction "impulse response" du système stable LTI (1.2) est

$$h(t) = \begin{cases} C\exp(At)B + D\delta(t), & t \geq 0 \\ 0, & t < 0. \end{cases} \tag{1.5}$$

1.2 Fonctions de transfert et Moments

1.2.1 Fonction de transfert

On considère le système dynamique LTI (1.2). Une façon classique de relier l'entrée et la sortie est d'utiliser une fonction de transfert du système précédent LTI. Pour cela on aura besoin de la transformée de Laplace

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

Si on l'applique à (1.2), on obtient

$$\begin{cases} s\,X(s) &= A\,X(s) + B\,U(s) \\ Y(s) &= C\,X(s) + DU(s), \end{cases}$$

oú $X(s)$, $Y(s)$ et $U(s)$ sont les transformées de Laplace des $x(t)$, $y(t)$ et $u(t)$ respectivement. Si on élimine $X(s)$ dans les deux équations précédentes, on obtient l'un des concepts les plus importants de la théorie des systèmes linéaires :

$$F(s) = C\,(s\,I_n - A)^{-1}\,B + D. \tag{1.6}$$

La fonction $F(s)$ est appelée la fonction de transfert du système (1.2). Cette fonction de transfert relie l'entrée et la sortie par $Y(s) = F(s)\,U(s)$ dans le domaine des fréquences. On rappelle que la plupart des techniques de réduction de modèles sont basées sur cette fonction de transfert ; [10, 30, 32, 38].

Dans la suite, on rappelle la notion de deux systèmes (LTI) équivalents.

Définition 1.1 *Deux systèmes LTI :* $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ *et* $\left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right]$ *sont dits équivalents s'ils ont la même fonction de transfert, i.e.,*

$$\tilde{F}(s) = \tilde{C}\,(s\,I_n - \tilde{A})^{-1}\,\tilde{B} + \tilde{D} = C\,(s\,I_n - A)^{-1}\,B + D = F(s).$$

Il est facile de voir que pour toute matrice non singulière $T \in \mathbb{R}^{n\times n}$, les deux systèmes $\left[\begin{array}{c|c} T^{-1}AT & T^{-1}B \\ \hline CT & D \end{array}\right]$ et $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ sont équivalents. L'intérêt de définir le système dans le domaine de fréquences est d'obtenir plusieurs systèmes équivalents qui nous donnent la liberté de choisir la version la plus stable. Sous cette transformation, la relation qui relie les vecteurs d'états est définie par $x(t) = T\tilde{x}(t)$.
La résolution de (1.2) avec une condition initiale $x_0 = x(t_0)$ donne

$$x(t) = \exp(A(t-t_0))x_0 + \int_{t_0}^{t} \exp(A(t-\tau))Bu(\tau)d\tau.$$

Ainsi, le vecteur de sortie $y(t)$ dans le domaine de temps s'écrit

$$y(t) = C\exp(A(t-t_0))x_0 + C\int_{t_0}^{t} \exp(A(t-\tau))Bu(\tau)d\tau. \tag{1.7}$$

En comparant cette dernière avec celle dans le domaine de fréquences

$$Y(s) = F(s)\,U(s) = (C\,(s\,I_n - A)^{-1}\,B)\,U(s),$$

on voit que l'écriture dans le domaine de fréquences est bien plus simple.

1.2.2 Moments d'une fonction de transfert

Dans la réduction de modèles de grande échelle, les méthodes basées sur les moments et/ou les paramètres de Markov sont parmi les questions développées dans ce manuscrit.
La fonction de transfert peut être développée en série de Taylor au voisinage d'un point $s_0 \in \mathbb{C}$, ce qui donne

$$F(s) = \eta_0(s_0) - \eta_1(s_0)(s - s_0) + \ldots + (-1)^j \eta_j(s_0)(s - s_0)^j + \ldots$$

tels que

$$\eta_j(s_0) = C\,(s_0\, I_n - A)^{-(j+1)}\, B, \quad \forall j \geq 0.$$

Le coefficient matriciel $\eta_j(s_0)$ s'appelle le j-ème moment du système (1.2) en s_0, pour $j \geq 0$.
Dans le cas particulier où $s_0 = \infty$, la fonction de transfert F peut se développer en série de Laurent comme suit

$$F(s) = \frac{1}{s}\, C\,(I_n - \frac{A}{s})^{-1}\, B = \frac{1}{s} \sum_{i=0}^{\infty} \eta_i(\infty)\, s^{-i}, \text{ avec } \eta_i(\infty) = C\, A^i\, B.$$

Rappelons que dans ce cas, les coefficients matriciels $\eta_i(\infty)$ s'appellent : paramètres de Markov de F. Plusieurs méthodes de réduction de modèles ont pour but de traiter le problème des moments et plus particulièrement les méthodes de type projection sur les espaces de Krylov [34, 48, 49] utilisent largement cette approche.
La question principale dans le problème des moments est de construire un modèle d'ordre réduit (A_m, B_m, C_m) tel que les l premiers moments $\left(\widehat{\eta}_j(s_0)\right)_{j=0}^{l-1}$ de la fonction de transfert réduite

$$F_m(s) = \widehat{\eta}_0(s_0) - \widehat{\eta}_1(s_0)(s - s_0) + \ldots + (-1)^j \widehat{\eta}_j(s_0)(s - s_0)^j + \ldots$$

et ceux de F coïncident i.e.

$$\widehat{\eta}_j(s_0) = \eta_j(s_0), \quad j = 0, \ldots, l-1.$$

Même question pour les paramètres de Markov, i.e. $s_0 = \infty$.

1.3 Concepts fondamentaux

1.3.1 Stabilité, contrôlabilité et observabilité

On remarque que dans la formule définissant la fonction de transfert, la matrice D ne joue pas un rôle important. Dans la suite on prendra $D = 0$. Ainsi le système LTI s'écrit

$$\begin{cases} \dot{x}(t) &= A\,x(t) + B\,u(t) \\ y(t) &= C\,x(t). \end{cases} \tag{1.8}$$

Définition 1.2 *(Stabilité)*
Une matrice A est dite stable si $\Lambda(A) \subset \mathbb{C}_-$ i.e., les parties réelles des valeurs propres de A sont toutes négatives. Un système LTI= $\left[\begin{array}{c|c} A & B \\ \hline C & \end{array}\right]$ est dit stable si la matrice A est stable.

On note $\mathbb{C}_- := \{s \in \mathbb{C}; \text{réel}(s) < 0\}$.

Définition 1.3 *Une fonction à valeur matricielle $F(s) \in \mathbb{C}^{n\times n}$ $(s \in \mathbb{C})$ est dite réelle positive si ces trois conditions sont satisfaites :*
1. *Tous les éléments de $F(s)$ sont analytiques pour $s \in \mathbb{C}_+$,*
2. *$F(s) \in \mathbb{R}^{n\times n}$ pour $(s > 0)$,*
3. *$F(s) + F^\star(s) \geq 0$ pour $s \in \mathbb{C}_+$.*

On note $\mathbb{C}_+ := \{s \in \mathbb{C}; \text{réel}(s) > 0\}$.

Définition 1.4 *(Passivité)*
Un système LTI stable est dit passive si sa fonction de transfert est réelle positive, i.e., $F(s) + F^\star(s) \geq 0$ pour $s \in \mathbb{C}_+$.

Définition 1.5 *(Contrôlabilité)*
Un système est dit contrôlable si à partir d'un état initial nul, tout état peut être atteint via un contrôle convenable, i.e., étant donné $z \in \mathbb{R}^n$, si $x(t_0) = 0$ alors il existe $u(t)$ tel que $x(t) = z$.

Une condition nécessaire et suffisante pour la contrôlabilité est donnée par

Proposition 1.1 *Un système LTI définie par* $\dot{x}(t) = A\,x(t) + B\,u(t)$ *est contrôlable si et seulement si*

$$rank([B, AB, A^2B, \ldots, A^{n-1}B]) = n.$$

Définition 1.6 *(Observabilité)*
Un système LTI = $\left[\begin{array}{c|c} A & B \\ \hline C & \end{array}\right]$ *est dit observable lorsque* $u(t) = 0$, $y(t)$ *est alors déterminé uniquement par* $x(t_0)$.

D'une manière équivalente, la proposition suivante donne une condition nécessaire et suffisante pour l'observabilité

Proposition 1.2 *Un système LTI* $\left[\begin{array}{c|c} A & B \\ \hline C & \end{array}\right]$ *est observable si et seulement si*

$$rank\begin{pmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{pmatrix} = n.$$

1.3.2 Les Gramians de contrôlabilité et d'observabilité

On suppose que le système dynamique LTI est stable.

Définition 1.7 *Le Gramian de contrôlabilité associé au système LTI* (1.8) *est définit par*

$$P = \int_0^\infty e^{tA} B B^\top e^{tA^\top} dt, \tag{1.9}$$

et le Gramian d'observabilité est définit par

$$Q = \int_0^\infty e^{tA^\top} C^\top C e^{tA} dt. \tag{1.10}$$

En appliquant la formule de Parseval sur ces dernières relations, on obtient les nouvelles écritures des Gramians :

$$\begin{aligned} P &= \int_0^\infty (j\omega I - A)^{-1} B B^\top (j\omega I - A^\top)^{-1} d\omega, \\ Q &= \int_0^\infty (j\omega I - A^\top)^{-1} C^\top C (j\omega I - A)^{-1} d\omega. \end{aligned}$$

Les fonctions de contrôlabilité et d'observabilité sont des fonctions d'état du système. La fonction de contrôlabilité traduit l'énergie nécessaire pour atteindre un certain état à partir des conditions initiales définies alors que la fonction observabilité traduit l'énergie de sortie pour un état initial donné et une entrée nulle. Elles permettent de caractériser l'importance relative de différents états. Sous les hypothèses que A est asymptotiquement stable, i.e $\lambda_i(A) \in \mathbb{C}_-$ (le demi-plan gauche ouvert) et que $\mathbf{\Sigma}$ est minimal (les paires (A,B) et (C,A) soient, respectivement, contrôlables et observables), les P et Q sont définis positifs et forment les solutions des deux équations de Lyapunov

$$\begin{aligned} AP + PA^\top + BB^\top &= 0, \qquad (1.11) \\ A^\top Q + QA + C^\top C &= 0. \qquad (1.12) \end{aligned}$$

On considère le nouveau système dynamique LTI

$$\tilde{\mathbf{\Sigma}} \equiv \left[\begin{array}{c|c} T^{-1}AT & T^{-1}B \\ \hline CT & D \end{array}\right], \qquad (1.13)$$

où T est une matrice non singulière. Les deux Gramians de contrôlabilité et observabilité $\tilde{P}$ et $\tilde{Q}$ s'écrivent

$$\begin{aligned} \tilde{P} &= \int_0^\infty e^{t\tilde{A}}\tilde{B}\tilde{B}^\top e^{t\tilde{A}^\top}, \\ \tilde{Q} &= \int_0^\infty e^{t\tilde{A}^\top}\tilde{C}^\top\tilde{C}e^{t\tilde{A}}, \end{aligned}$$

où $\tilde{A} = T^{-1}AT$, $\tilde{B} = T^{-1}B$ et $\tilde{C} = CT$. Il est facile de voir que

$$\tilde{P} = T^{-1}PT^{-\top}, \qquad \text{et} \qquad \tilde{Q} = T^\top QT.$$

On remarque que les Gramians correspondant aux deux systèmes équivalents LTI ne sont pas similaires, mais la similarité est préservée pour le produit de deux Gramians

$$\tilde{P}\tilde{Q} = T^{-1}PQT, \qquad \tilde{Q}\tilde{P} = T^\top QPT^\top.$$

Pour discuter des résultats généraux sur les équations de Lyapunov, on définit l'équation de Sylvester

$$FX + XG + H = 0. \qquad (1.14)$$

L'équation matricielle de Sylvester et de Lyapunov ont joué un rôle clé dans de nombreuses applications telles que le contrôle et la théorie de la communication et les problèmes de réduction de modèles [25, 83] comme la méthode de "Balanced truncation" qui nécessite la résolution de (1.11) et (1.12) pour obtenir P et Q, et dans d'autres applications comme le filtrage et la restauration d'images.
L'équation matricielle (1.14) peut être reformulée comme le système linéaire

$$\big(I_s \otimes F + G^\top \otimes I_n\big)\vec{X} = \vec{H},$$

où $\otimes$ dénote le produit de Kronecker matriciel (pour deux matrices M et N, $[M \otimes N]_{i,j} = [M_{i,j} \otimes N]_{i,j}$) et $\vec{M}$ le vecteur obtenu en stockant les vecteurs colonnes de M dans une unique colonne.
Il est facile d'établir que les valeurs propres de $I_s \otimes F + G^\top \otimes I_n$ sont $\lambda_i + \mu_j$, où $\lambda_i \in \Lambda(F)$, $\mu_j \in \Lambda(G)$, et $\Lambda(.)$ est le spectre d'une matrice carrée.
Cette dernière écriture nous donne une condition nécessaire et suffisante pour que l'équation (1.14) ait une solution unique ; cette condition est $\lambda_i + \mu_j \neq 0$ pour tout $\lambda_i \in \Lambda(A)$ et $\mu_j \in \Lambda(B)$, $i = 1, \ldots, n$ $j = 1, \ldots, s$.

Théorème 1.1 *L'équation de Sylvester* (1.14) *admet une unique solution si et seulement si les spectres des matrices G et $-F$ sont disjoint, i.e., $\Lambda(F) \cap \Lambda(-G) = \emptyset$.*

Cette dernière condition nous assure l'existence et l'unicité de la solution de l'équation (1.14) sans en donner une écriture explicite. Sous une hypothèse plus forte, la proposition suivante nous donne une écriture explicite de la solution

Proposition 1.3 *Si les matrices F et G sont stables, alors l'équation* (1.14) *a pour unique solution la matrice*

$$X = \int_0^\infty e^{tF} H e^{tG}.$$

Lorsque la taille du problème est petite ou moyenne, on peut utiliser la méthode de Bartels-Stewart [11]. L'idée est de calculer la décomposition de Schur des deux coefficients matriciels F et G et de transformer l'équation (1.14) en une équation équivalente en utilisant la structure triangulaire des matrices de Schur.
L'algorithme de Bartels-Stewart [11] est décrit comme ce qui suit :

1. Calculer les formes de Schur : $F = U^\top R U$ et $G = V S V^\top$ avec R et S sont des matrices triangulaires supérieures ;
2. Résoudre $RY + YS + U^\top C V = 0$;
3. Calculer $X = UYV^\top$.

1.4 Différentes normes des systèmes dynamiques

Le but de ce paragraphe est de rappeler les différentes normes pour les systèmes dynamiques et la fonction de transfert. Ces normes seront utilisées pour déterminer l'erreur entre le modèle original et le modèle approché.

1.4.1 La norme $\mathcal{H}_2$

Définition 1.8 *La norme $\mathcal{H}_2$ de la fonction de transfert $F(s)$ est définie par :*

$$\|F(.)\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} trace[F(i\omega)^\top F(i\omega)]d\omega,$$

où i est le nombre complexe $i^2 = -1$.

On considère la réponse impulsive $g(t) = \mathcal{L}^{-1}[F(s)] = Ce^{tA}B$ où $\mathcal{L}$ est la transformée de Laplace. On a alors

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

En utilisant la relation de Parseval, on obtient

$$\int_0^{\infty} trace[g(t)^\top g(t)]dt = \frac{1}{2\pi}\int_{-\infty}^{\infty} trace[F(i\omega)^\top F(i\omega)]d\omega.$$

La norme $\mathcal{H}_2$ peut être exprimée comme

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

Si on remplace $g(t)$ par son écriture, on obtient

$$\|F(.)\|_{\mathcal{H}_2}^2 = trace\Big[B^\top\Big(\int_0^{\infty} e^{tA^\top}C^\top Ce^{tA}dt\Big)B\Big].$$

Par conséquent, la norme $\mathcal{H}_2$ peut être calculée comme suit

$$\|F(.)\|_{\mathcal{H}_2}^2 = trace(B^\top QB),$$

où Q est le Gramian d'observabilité défini en (1.10).
En supposant que la matrice A est stable, le Gramian d'observabilité peut être calculé en résolvant la seconde équation matricielle de Lyapunov (1.12).

On note que de manière similaire, la norme $\mathcal{H}_2$ peut être calculée en utilisant le Gramian de contrôlabilité défini par (1.9). Dans ce cas, la norme $\mathcal{H}_2$ peut s'écrire comme

$$\|F(.)\|^2_{\mathcal{H}_2} = trace(CPC^\top).$$

Le but de la réduction de modèle est de produire un modèle réduit de petite dimension m tel que l'erreur entre les sorties $y(t) - y_m(t)$ soit petite. Différentes mesures des choix d'approximation et de différentes classes d'entrées conduisent aux différents modèles de réduction.
Si on veut minimiser

$$\max_{t>0}|y(t) - y_m(t)|,$$

pour toutes les entrées u avec une énergie bornée, i.e,

$$\int_0^\infty |u(t)|^2 dt \leq 1,$$

alors on a

$$\begin{aligned}\max_{t>0}|y(t) - y_m(t)| &= \max_{t>0}\left|\frac{1}{2\pi}\int_0^\infty \big(Y(i\omega) - Y_m(i\omega)\big)e^{i\omega t}d\omega\right| \\ &\leq \frac{1}{2\pi}\int_{-\infty}^\infty |Y(i\omega) - Y_m(i\omega)|d\omega.\end{aligned}$$

Comme $Y(s) - Y_m(s) = \big(F(s) - F_m(s)\big)U(s)$, il en résulte que

$$\begin{aligned}\frac{1}{2\pi}\int_{-\infty}^\infty |Y(i\omega) - Y_m(i\omega)|d\omega &= \frac{1}{2\pi}\int_{-\infty}^\infty |F(i\omega) - F_m(i\omega)||U(i\omega)|d\omega \\ &\leq \left(\frac{1}{2\pi}\int_{-\infty}^\infty |F(i\omega) - F_m(i\omega)|^2 d\omega\right)^{\frac{1}{2}}\left(\frac{1}{2\pi}\int_{-\infty}^\infty |u(t)|^2 dt\right)^{\frac{1}{2}} \\ &\leq \left(\frac{1}{2\pi}\int_{-\infty}^\infty |F(i\omega) - F_m(i\omega)|^2 d\omega\right)^{\frac{1}{2}} = \|F - F_m\|_{\mathcal{H}_2}.\end{aligned}$$

1.4.2 La norme $\mathcal{H}_\infty$

Dans cette partie, on rappelle la norme $\mathcal{H}_\infty$ pour une fonction de transfert.

Définition 1.9 *La norme $\mathcal{H}_\infty$ de la fonction de transfert F est définie comme*

$$\|F(.)\|_{\mathcal{H}_\infty} = \sup_{\omega\in\mathbb{R}} \sigma_{max}(F(i\omega)),$$

où σ_{max} désigne la plus grande valeur singulière.

Pour approcher la norme $\mathcal{H}_\infty$ dans la pratique, on choisit un ensemble de fréquences $\Omega_N = \{\omega_1, \omega_2, \ldots, \omega_N\}$ et on cherche

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

1.4.3 La norme de Hankel

Les valeurs singulières de Hankel pour un système dynamique stable LTI sont les racines carrées de produit de Gramians de contrôlabilité et d'observabilité :

$$\sigma_i(F) = \sigma_i(\mathbf{\Sigma}) = \sqrt{\lambda_i(\mathbf{PQ})},$$

où P et Q sont les Gramians de système dynamique LTI dénoté par $\mathbf{\Sigma}$.

Définition 1.10 *La norme de Hankel pour un système dynamique stable LTI est donnée par*

$$\|F(.)\|_{\mathcal{H}} = \max_i \sigma_i(F).$$

CHAPITRE 2

RATIONAL BLOCK ARNOLDI METHODS FOR MODEL REDUCTION IN LARGE-SCALE MIMO DYNAMICAL SYSTEMS

2.1 Introduction

Let us consider a linear time-invariant (LTI) multi-input and multi-output (MIMO) system described by the state-space equations

$$\begin{cases} \dot{x}(t) &= A\,x(t) + B\,u(t) \\ y(t) &= C\,x(t), \end{cases} \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$ denotes the state vector and $u(t)$, $y(t) \in \mathbb{R}^p$ are the input and output vectors respectively of the (LTI) system (2.1). The matrix $A \in \mathbb{R}^{n\times n}$ is assumed to be large and sparse, and $B,\ C^T \in \mathbb{R}^{n\times p}$ are tall matrices with $p \ll n$. For single-input single-output (SISO) systems, the matrices B and C are vectors (i.e $p = 1$).

The linear time invariant system (2.1) arises in simulations of dynamical systems where partial differential equations are involved and the matrices A and B which are generated by the discretization of these equations are often very large. In many cases, the large state-space dimension (or order) n of the system (2.1) makes the simulations very difficult. Therefore, it is necessary to seek for a lower order model

whose behaviour is close to the original :

$$\begin{cases} \dot{x}_m(t) &= A_m\, x_m(t) + B_m\, u(t) \\ y_m(t) &= C_m\, x_m(t), \end{cases} \tag{2.2}$$

such that $A_m \in \mathbb{R}^{m\times m}$, $B_m,\ C_m^T \in \mathbb{R}^{m\times p}$, $x_m(t),\ y_m(t) \in \mathbb{R}^m$, and $m \ll n$, while maintaining the most relevant properties of the original system (2.1).
Many existing model order reduction methods such as Padé approximation [37,80], balanced truncation [70], optimal Hankel norm [35,36] and Krylov subspace based methods In particular the Arnoldi algorithm [24, 25, 32, 52] take advantage of the sparsity of the large-scale model and have been extensively used for large problems ; see [9, 32, 48].
When using block Krylov subspaces, one projects the system matrices of the original problem onto the subspace $\mathcal{K}_m(A,B) = Range\{B, AB, \ldots, A^{m-1}B\}$ generated by the columns of the matrices $B, AB, \ldots, A^{m-1}B$ and try to get a sufficiently accurate reduced system with a moderate space dimension.

In this work, we will consider the rational block Krylov subspace which is a subspace of $\mathbb{R}^n$ generated by the columns of the matrices $B, (A-s_2I)^{-1}B, \ldots, \prod_{i=2}^{m}(A-s_iI)^{-1}B$, where $s_2, \ldots, s_m$ are some selected complex shifts. The original large problem is projected onto this block Krylov subspace to get a new low order dynamical system close in some sense to the initial one. The rational Krylov subspace procedure was originally proposed by Ruhe [74] in the context of approximating interior eigenvalues and has been used during the last years for model order reduction ; see [37]. The selection of good shifts is a crucial issue for the quality of the approximation. The use of rational Krylov spaces is recognized as a powerful tool within model order reduction techniques for linear dynamical systems, however, its success has been hindered by the lack of a parameter-free procedure, which would effectively generate the sequence of shifts used to build the space. Major efforts have been devoted to this question in the recent years ; see for example [27, 28, 32, 34, 45, 64]. In the context of $\mathcal{H}_2$-optimality reduction, an interesting attempt to provide an automatic selection has been proposed recently in [41]. However, the computational and memory costs of this approach have not been fully assessed. We also mention the early contribution due to Grimme [37] for determining a sequence of shifts. Another approach has been recently developed in [28] to generate these parameters. In this work, we propose an adaptive computation of the shifts for building the rational space by minimizing, at each iteration of the process, some matrix norms. We will

derive some theoretical results such as upper bounds for the norm of the error on the transfer function. Some numerical tests will be provided in order to compare our approach with other existing methods.

This chapter is organized as follow : In Section 2, we introduce the rational block Arnoldi and give some new algebraic relations. Section 3 is devoted to the selection of the shifts that are used in the construction of rational Krylov subspaces and we give an error bound for the norm of the error on the transfer function. A new modified rational block Arnoldi is proposed in Section 4 and some new Arnoldi-like relations are proposed. The last section is devoted to some numerical tests and comparisons to some well known model order reduction methods.
We will use the following notations : the 2-norm of a vector or of a matrix will be denoted by $\| \, . \, \|$ and I_p is the identity matrix of dimension $p \times p$.

2.2 The rational block Arnoldi method

In this section we will describe the rational block Arnoldi algorithm for computing an orthonormal basis of the rational block Krylov subspace defined for a given matrix $B \in \mathbb{R}^{n\times p}$ as

$$\mathbf{K}_m(A, B) = \text{Range}\{B, (A - s_2 I)^{-1}B, \ldots, \prod_{i=2}^{m}(A - s_i I)^{-1}B\}. \tag{2.3}$$

The rational block Arnoldi algorithm generates a sequence of $n \times p$ blocks $\{V_1, \ldots, V_m\}$ whose columns form an orthonormal basis of the rational block Krylov subspace $\mathbf{K}_m(A, B)$. The algorithm is described as follows

Algorithm 2.1 The Rational Block Arnoldi Algorithm

— *Input :* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$ and a fixed integer m.

— Compute $V_1 = QR(B)$, $\mathcal{V}_1 = [V_1]$.

— For $j = 1, \ldots, m-1$

1. $\widetilde{V}_{j+1} = (A - s_{j+1}I)^{-1}V_j$.
2. Orthogonalization step :

   For $i = 1, 2, \ldots, j$

   $H_{i,j} = V_i^\top \widetilde{V}_{j+1}$;

   $\widetilde{V}_{j+1} = \widetilde{V}_{j+1} - V_i H_{i,j}$;

   End For

The Rational Block Arnoldi Algorithm - Part 2

3. $QR(\widetilde{V}_{j+1}) = V_{j+1}H_{j+1,j}$.

4. $\mathcal{V}_{j+1} = [\mathcal{V}_j,\ V_{j+1}]$.

— End For.

The shifts $s_2, \ldots, s_m$ will be chosen a priori or a posteriori during the process and this will be explained later. After m steps, the rational block Arnoldi algorithm generates a block matrix $\mathcal{V}_m = [V_1, \ldots, V_m] \in \mathbb{R}^{n\times mp}$ whose columns form an orthonormal basis of the rational block Krylov subspace $\mathbf{K}_m(A, B)$ and an upper $(m+1)p \times mp$ block Hessenberg matrix $\overline{\mathcal{H}}_m$ whose blocks $H_{i,j}$ are defined by Algorithm 2.1. The $mp\times mp$ upper block Hessenberg matrix $\mathcal{H}_m$ is obtained from $\overline{\mathcal{H}}_m$ by deleting its last p-rows. In the sequel we will also use the restriction matrix $\mathcal{T}_m$ defined by $\mathcal{T}_m := \mathcal{V}_m^* A\mathcal{V}_m$. We first give some new algebraic relations generalising the well known Arnoldi-like relation given for the classical case.

Proposition 2.1 *Let $\mathcal{V}_m$, $\overline{\mathcal{H}}_m$ and $\mathcal{H}_m$ be the matrices generated by the rational block Arnoldi algorithm and let $\mathcal{S}_m$ be the diagonal matrix $diag(s_2 I_p, \ldots, s_{m+1} I_p)$ where $\{s_2, ..., s_{m+1}\}$ denotes the set of shifts used in the algorithm. Then we have the following relation*

$$\mathcal{T}_m := \mathcal{V}_m^* A\mathcal{V}_m = (I_{mp} + \mathcal{H}_m S_m - \mathcal{V}_m^* A V_{m+1} H_{m+1,m} E_m^*)\mathcal{H}_m^{-1},$$

where $E_m^* = [0_p, \ldots, 0_p, I_p] = (e_m^* \otimes I_p)$.

Proof After m steps of the rational block Arnoldi algorithm, we have

$$(A - s_{j+1}I_n)^{-1}V_j = \sum_{i=1}^{j+1} V_i H_{i,j} \qquad for\ j = 1, \ldots, m$$

then

$$V_j = A(\sum_{i=1}^{j+1} V_i H_{i,j}) - s_{j+1}(\sum_{i=1}^{j+1} V_i H_{i,j}) \qquad for\ j = 1, \ldots, m.$$

This gives the following relation

$$\mathcal{V}_m = A(\mathcal{V}_{m+1}\overline{\mathcal{H}}_m) - (\mathcal{V}_{m+1}\overline{\mathcal{H}}_m)\mathcal{S}_m,$$

which can also be written as

$$\mathcal{V}_m = A(\mathcal{V}_m\mathcal{H}_m + V_{m+1}H_{m+1,m}E_m^*) - (\mathcal{V}_m\mathcal{H}_m + V_{m+1}H_{m+1,m}E_m^*)\mathcal{S}_m.$$

Multiplying the last equality on the left by $\mathcal{V}_m^*$ and using the fact that the blocks $V_1, \ldots, V_{m+1}$ are orthonormal, we get the identity

$$I_{mp} = \mathcal{T}_m \mathcal{H}_m + \mathcal{V}_m^* A V_{m+1} H_{m+1,m} E_m^* - \mathcal{H}_m \mathcal{S}_m.$$

Finally, we can deduce the relation

$$\mathcal{T}_m = (I_{mp} + \mathcal{H}_m \mathcal{S}_m - \mathcal{V}_m^* A V_{m+1} H_{m+1,m} E_m^*) \mathcal{H}_m^{-1},$$

which ends the proof.
We can also state the following result

Proposition 2.2 *Under the same assumptions as in Proposition 2.1, we have*

$$A\mathcal{V}_m = \mathcal{V}_m \mathcal{T}_m - (I_n - \mathcal{V}_m \mathcal{V}_m^*) A V_{m+1} H_{m+1,m} E_m^* \mathcal{H}_m^{-1} + V_{m+1} H_{m+1,m} E_m^* S_m \mathcal{H}_m^{-1}.$$

Proof As stated in the previous proposition, we have

$$\mathcal{V}_m = A(\mathcal{V}_m \mathcal{H}_m + V_{m+1} H_{m+1,m} E_m^*) - (\mathcal{V}_m \mathcal{H}_m + V_{m+1} H_{m+1,m} E_m^*) \mathcal{S}_m.$$

Hence, we can write

$$\begin{aligned} A\mathcal{V}_m \mathcal{H}_m &= \mathcal{V}_m - A V_{m+1} H_{m+1,m} E_m^* + \mathcal{V}_m \mathcal{H}_m \mathcal{S}_m + V_{m+1} H_{m+1,m} E_m^* \mathcal{S}_m \\ &= \mathcal{V}_m (I_{mp} + \mathcal{H}_m \mathcal{S}_m) - A V_{m+1} H_{m+1,m} E_m^* + V_{m+1} H_{m+1,m} E_m^* \mathcal{S}_m. \end{aligned}$$

Using Proposition 2.1, we obtain the following relation

$$\begin{aligned} A\mathcal{V}_m \mathcal{H}_m &= \mathcal{V}_m (\mathcal{T}_m \mathcal{H}_m + \mathcal{V}_m^* A V_{m+1} H_{m+1,m} E_m^*) - A V_{m+1} H_{m+1,m} E_m^* \\ &+ V_{m+1} H_{m+1,m} E_m^* \mathcal{S}_m \\ &= \mathcal{V}_m \mathcal{T}_m \mathcal{H}_m - (I_n - \mathcal{V}_m \mathcal{V}_m^*) A V_{m+1} H_{m+1,m} E_m^* + V_{m+1} H_{m+1,m} E_m^* \mathcal{S}_m. \end{aligned}$$

Therefore

$$A\mathcal{V}_m = \mathcal{V}_m \mathcal{T}_m - (I_n - \mathcal{V}_m \mathcal{V}_m^*) A V_{m+1} H_{m+1,m} E_m^* \mathcal{H}_m^{-1} + V_{m+1} H_{m+1,m} E_m^* \mathcal{S}_m \mathcal{H}_m^{-1}.$$

2.3 An adaptive computation of the shifts

In this section, we will see some *a posteriori* and *a priori* procedures for selecting good shifts used during the construction of the rational block Arnoldi basis. This is a crucial problem when using rational Krylov subspace methods.

2.3.1 An *a priori* selection of the shifts

We briefly describe an *a priori* way for selecting the complex shifts. This technique was introduced by Penzl [73] and implemented in the routine `lp_para` of the library LYAPACK [72]. The parameters are selected by solving the following min-max problem ; see [73, 84, 85] for more details.

$$\{s_1, s_2, \ldots, s_l\} = arg \min_{\{\mu_1,\mu_2,\ldots,\mu_l\}\in\mathbb{C}_-} \left(\max_{\lambda\in\sigma(A)} \frac{|\,(\lambda-\mu_1)\ldots(\lambda-\mu_l)\,|}{|\,(\lambda+\mu_1)\ldots(\lambda+\mu_l)\,|} \right), \quad (2.4)$$

where $\sigma(A)$ denotes the spectrum of the matrix A.
As we generally are unable to compute the spectrum of the matrix A, the classical approach is to cover it by a domain $\Omega \subset \mathbb{C}_-$ and then to solve the minimax problem with respect to Ω. In [72, 73], a heuristic procedure was proposed to find "sub-optimal" parameters. This technique first generates a discrete set which approximates the spectrum $\sigma(A)$ using a pair of Arnoldi processes. The first one acts on the matrix A and generates k_+ Ritz values which tend to approximate the eigenvalues far from the origin. The second process, acting on the matrix A^{-1}, generates k_- Ritz values whose inverses are close to the origin. The set of shift parameters is then chosen as a subset of these Ritz values. This procedure is widely used in the ADI-type methods for solving large scale matrix equations such as Lyapunov or Sylvester matrix equations ; see for example [12, 17]

2.3.2 A new adaptive selection of the shifts

In this subsection we propose an adaptive technique for computing the shifts that are used to build the rational Krylov subspace. This procedure automatically generates the sequence of shifts during the construction of the rational Arnoldi subspaces.
A classical way of relating the input to the output is to use the transfer function (or impulse response in the time domain) of the LTI system (2.1). Indeed, applying the Laplace transform

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

to the dynamical system (2.1), we obtain

$$\begin{cases} s\,X(s) &= A\,X(s) + B\,U(s) \\ Y(s) &= C\,X(s) \end{cases},$$

where $X(s)$, $Y(s)$ and $U(s)$ are the Laplace transforms of $x(t)$, $y(t)$ and $u(t)$, respectively. Eliminating $X(s)$ in the previous two equations, we get

$$Y(s) = H(s)\,U(s),$$

where

$$H(s) = C\,(s\,I_n - A)^{-1}\,B. \tag{2.5}$$

The rational function $H(s)$ is called the transfer function of the system (2.1). We recall that most model order reduction techniques, for example the moment-matching approaches, are based on the approximation of this transfer function; see [10, 32, 38]. If the number of state variables is very large, it would be very difficult to use the full system for simulation or run-on-time control. So it is reasonable to look for lower order models that approximate the behavior of the original models. This will be done by approximating the transfer function (2.5).

Let us write $H(s) = C\,X$ where $X \in \mathbb{R}^{n\times p}$ is the solution of the matrix linear system

$$(sI_n - A)X = B. \tag{2.6}$$

In order to approximate the transfer function H, we will look for approximations of the solution X of the multiple linear system (2.6). Let X_m denotes the approximate solution obtained by the Galerkin projection method onto the rational Krylov subspace $\mathbf{K}_m(A, B)$. This approximate solution is given by

$$X_m = \mathcal{V}_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^* B,$$

where $\mathcal{T}_m = \mathcal{V}_m^* A \mathcal{V}_m$, hence the transfer function H is approximated by the low order transfer function corresponding to the projected low order dynamical system and given by

$$H_m(s) = C\mathcal{V}_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^* B,$$

which can be written as

$$H_m(s) = C_m(sI_{mp} - \mathcal{T}_m)^{-1}B_m, \tag{2.7}$$

where $C_m = C\mathcal{V}_m$ and $B_m = \mathcal{V}_m^* B$.

In the sequel, we will give an expression for the norm of the error $H(s) - H_m(s)$, which will be used for the selection of our shift parameters. First, we recall the norm $\mathcal{H}_\infty$ for a matrix-valued function

$$\|H\|_\infty = \sup_{y\in\mathbb{R}} \sigma_{max}(H(iy)).$$

Indeed we have :

$$\begin{aligned} H(s) - H_m(s) &= CX - CX_m \\ &= C(sI_n - A)^{-1}B - CX_m \\ &= C(sI_n - A)^{-1}[B - (sI_n - A)X_m]. \end{aligned}$$

By applying the norm described above, we obtain

$$\|H(s) - H_m(s)\| \leq \|C(sI_n - A)^{-1}\|\|\Gamma_m\|_\infty$$

where $\Gamma_m = B - (sI_n - A)X_m$. So, one way for selecting a new shift, is to choose those that allows us to reach $\|\Gamma_m\|_\infty$. Hence, our new shift s_{m+1} will be chosen as

$$s_{m+1} = \{s \in \mathbb{R} : \sigma_{max}(\Gamma_m(is)) = \|\Gamma_m\|_\infty\}. \tag{2.8}$$

As we will see in the numerical tests, this simple procedure gives good results.

2.3.3 An error expression for the transfer function

In the following proposition we give an upper bound for the 2-norm of the error $H(s) - H_m(s)$.

Proposition 2.3 *Let H be the transfer function defined in (2.5) and let H_m be its approximation. Then, under the conditions $\|A\| < |s|$, we have the following upper bound :*

$$\begin{aligned} H(s) - H_m(s) &= C(sI_n - A)^{-1}\big[- (I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1} \\ &+ V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\big](sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B. \end{aligned}$$

And

$$\|H(s) - H_m(s)\| \leq \frac{\|C\|\|H_{m+1,m}\|(\|A\| + \|S_m\|)\|\mathcal{H}_m^{-1}\|}{(|s| - \|A\|)}\|(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\|.$$

Proof We have :

$$\begin{aligned} H(s) - H_m(s) &= C(sI_n - A)^{-1}B - C_m(sI_{mp} - \mathcal{T}_m)^{-1}B_m \\ &= C(sI_n - A)^{-1}B - C\mathcal{V}_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B \\ &= C(sI_n - A)^{-1}\big[B - (sI_n - A)\mathcal{V}_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big] \\ &= C(sI_n - A)^{-1}\big[B - (s\mathcal{V}_m - A\mathcal{V}_m)(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big]. \end{aligned}$$

Using Proposition 2.2, we obtain

$$\begin{aligned} H(s) &- H_m(s) = C(sI_n - A)^{-1}\big[B - \big(s\mathcal{V}_m - \mathcal{V}_m\mathcal{T}_m \\ &+ (I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1} \\ &- V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\big)(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big] \\ &= C(sI_n - A)^{-1}\big[B - (s\mathcal{V}_m - \mathcal{V}_m\mathcal{T}_m)(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B \\ &- \big((I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1} - V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\big) \\ &\times (sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big] \\ &= C(sI_n - A)^{-1}\big[B - \mathcal{V}_m\mathcal{V}_m^*B - \big((I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1} \\ &- V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\big)(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big]. \end{aligned}$$

As B is in the rationa Krylov subspace (2.3), then we have $\mathcal{V}_m\mathcal{V}_m^*B = B$. This gives the following expression

$$\begin{aligned} H(s) - H_m(s) &= C(sI_n - A)^{-1}\big[- (I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1} \\ &+ V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\big](sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B. \end{aligned}$$

By applying the 2-norm we obtain

$$\begin{aligned} \|H(s) - H_m(s)\| &\leq \|C(sI_n - A)^{-1}\|\Big[\|(I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1}\| \\ &+ \|V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\|\Big] \times \|(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\|. \end{aligned}$$

Therefore , as $\|A\| < |s|$ we obtain

$$\begin{aligned} \|H(s) - H_m(s)\| &\leq \frac{\|C\|}{(|s| - \|A\|)}\Big[\|(I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}H_{m+1,m}E_m^*\mathcal{H}_m^{-1}\| \\ &+ \|V_{m+1}H_{m+1,m}E_m^*S_m\mathcal{H}_m^{-1}\|\Big]\,\|(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\|, \end{aligned}$$

furthermore as $I_n - \mathcal{V}_m\mathcal{V}_m^*$ is an orthogonal projection and $\|V_{m+1}\| = 1$, we get

$$\|H(s) - H_m(s)\| \leq \frac{\|C\|\|H_{m+1,m}\|(\|A\| + \|S_m\|)\|\mathcal{H}_m^{-1}\|}{(|s| - \|A\|)}\|(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\|.$$

2.4 A modified rational block Arnoldi algorithm

In this section, we describe a generalization of the rational Krylov subspace, allowing some shifts to be equal to infinity. At each step $j+1$, the algorithm computes

a new block $\widetilde{V}_{j+1} = (A - s_{j+1}I)^{-1}V_j$ if s_{j+1} is finite and $\widetilde{V}_{j+1} = AV_j$ if $s_{j+1} = \infty$. The modified rational Arnoldi algorithm is summarized as follows

Algorithm 2.2 Modified Rational Arnoldi Algorithm

— *Input :* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, m.

— Compute $V_1 = QR(B)$, $\mathcal{V}_1 = [V_1]$.

— For $j = 1, \ldots, m-1$

1. Set $\widetilde{V}_{j+1} = \begin{cases} (A - s_{j+1}I)^{-1}V_j, & if \quad s_{j+1} \neq \infty;\ s_{j+1} \text{ by using (2.8).} \\ AV_j, & if \quad s_{j+1} = \infty \end{cases}$

2. Orthogonalization step :

   For $i = 1, 2, \ldots, j$

   $H_{i,j} = V_i^\top \widetilde{V}_{j+1}$;

   $\widetilde{V}_{j+1} = \widetilde{V}_{j+1} - V_i H_{i,j}$;

   End For

3. $QR\big(\widetilde{V}_{j+1}\big) = V_{j+1}H_{j+1,j}$.

4. $\mathcal{V}_{j+1} = [\mathcal{V}_j,\ V_{j+1}]$.

— End For.

The idea of including infinity as a possible interpolation point could be considered as a generalization of the extended block Arnoldi algorithm [46, 81]. This new version also allows one to obtain new simple Arnoldi-like relations that could be used when deriving for example error bounds or residual error expressions and perturbation analysis. Using the modified rational Arnoldi algorithm, we can state the following simple Arnoldi-like relations

Proposition 2.4 *Let* $\mathcal{S} = \{s_2, \ldots, s_m\} \subset \mathbb{C}$ *and* $\mathcal{V}_m = [V_1, \ldots, V_{m+1}] \in \mathbb{R}^{n\times(m+1)p}$ *as generated by running Algorithm 2.2 for one extra interpolation point at* $s_{m+1} = \infty$. *Then the following Arnoldi-like equations are satisfied*

$$\begin{aligned} A\mathcal{V}_m &= \mathcal{V}_{m+1}\overline{\mathcal{T}}_m \\ &= \mathcal{V}_m\mathcal{T}_m + V_{m+1}N_m, \end{aligned}$$

where $\overline{\mathcal{T}}_m = \mathcal{V}_{m+1}^* A\mathcal{V}_m$, $\mathcal{T}_m = \mathcal{V}_m^* A\mathcal{V}_m$ *and* $N_m = V_{m+1}^* A\mathcal{V}_m$.

Proof From Algorithm 2.2, it easy to see that the following relations are satisfied

$$Range([\mathcal{V}_m \;\; AV_m]) = Range(\mathcal{V}_{m+1}), \qquad \text{and} \;\; \mathcal{V}_{m+1}^* \mathcal{V}_{m+1} = I_{(m+1)p}.$$

Let us now prove that

$$Range(A\mathcal{V}_m) \subset Range(\mathcal{V}_{m+1}).$$

Indeed, after $m-1$ iterations of the rational Arnoldi algorithm, the proof of Proposition 2.1 gives us

$$\mathcal{V}_{m-1} = A(\mathcal{V}_{m-1}\mathcal{H}_{m-1} + V_m H_{m,m-1} E_{m-1}^*) - (\mathcal{V}_{m-1}\mathcal{H}_{m-1} + V_m H_{m,m-1} E_{m-1}^*)\mathcal{S}_{m-1},$$

then

$$\begin{aligned} A\mathcal{V}_{m-1} &= \mathcal{V}_{m-1}\mathcal{H}_{m-1}^{-1} - AV_m H_{m,m-1} E_{m-1}^* \mathcal{H}_{m-1}^{-1} \\ &+ (\mathcal{V}_{m-1}\mathcal{H}_{m-1} + V_m H_{m,m-1} E_{m-1}^*)\mathcal{S}_{m-1}\mathcal{H}_{m-1}^{-1}. \end{aligned}$$

Using the fact that $Range(AV_m) \subset Range(\mathcal{V}_{m+1})$, it is clear that $Range(A\mathcal{V}_{m-1}) \subset Range(\mathcal{V}_{m+1})$, and $Range(A\mathcal{V}_m) \subset Range(\mathcal{V}_{m+1})$. Therefore we have

$$A\mathcal{V}_m = \mathcal{V}_{m+1}\overline{\mathcal{T}}_m, \tag{2.9}$$

for some matrix $\overline{\mathcal{T}}_m$. Since $\mathcal{V}_{m+1}$ is orthonormal, multiplying (2.9) on the left by $\mathcal{V}_{m+1}^*$, we get $\overline{\mathcal{T}}_m = \mathcal{V}_{m+1}^* A\mathcal{V}_m$. We can also see that

$$A\mathcal{V}_m = \mathcal{V}_m\mathcal{L}_m + V_{m+1}N_m \tag{2.10}$$

for some matrices $\mathcal{L}_m$ and N_m. Therefore, multiplying (2.10) on the left by $\mathcal{V}_m^*$ gives

$$\mathcal{L}_m = \mathcal{T}_m = \mathcal{V}_m^* A\mathcal{V}_m,$$

and multiplying (2.10) by V_{m+1}^* we get

$$N_m = V_{m+1}^* A\mathcal{V}_m.$$

This completes the proof.

In the next proposition, we give a new expression of the error $H(s) - H_m(s)$ which could be used to compute a new upper bound for the norm of the error on the transfer function.

Proposition 2.5 *Under the hypothesis of Proposition 2.4 , we have the following relation*

$$H(s) - H_m(s) = C(sI_n - A)^{-1}V_{m+1}N_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B, \tag{2.11}$$

and we also have the upper bounds for the norm of the error given by

$$\begin{aligned} \|H(s) - H_m(s)\| &\leq \frac{\|C\|}{|s| - \|A\|}\|N_m\|\|(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\| && (2.12)\\ &\leq \frac{\|C\|\|B\|\|A\|}{|s| - \|A\|}\|(sI_{mp} - \mathcal{T}_m)^{-1}\|. && (2.13) \end{aligned}$$

Proof

$$\begin{aligned} H(s) - H_m(s) &= C(sI_n - A)^{-1}B - C_m(sI_{mp} - \mathcal{T}_m)^{-1}B_m\\ &= C(sI_n - A)^{-1}B - C\mathcal{V}_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\\ &= C(sI_n - A)^{-1}\big[B - (sI_n - A)\mathcal{V}_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big]\\ &= C(sI_n - A)^{-1}\big[B - (s\mathcal{V}_m - A\mathcal{V}_m)(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big]. \end{aligned}$$

We use the result of Proposition 2.4 and we obtain

$$\begin{aligned} H(s) - H_m(s) &= C(sI_n - A)^{-1}\\ &\times \big[B - (s\mathcal{V}_m - \mathcal{V}_m\mathcal{T}_m - V_{m+1}N_m)(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big]\\ &= C(sI_n - A)^{-1}\big[B - \mathcal{V}_m\mathcal{V}_m^*B + V_{m+1}N_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B\big]. \end{aligned}$$

Using the fact that B is in the rational block Krylov subspace $\mathbf{K}_m(A, B)$, it follows that

$$H(s) - H_m(s) = C(sI_n - A)^{-1}V_{m+1}N_m(sI_{mp} - \mathcal{T}_m)^{-1}\mathcal{V}_m^*B.$$

The relations (2.12) are easily derived from the preceding relation.

2.5 Numerical experiments

In this section, we give some numerical examples to show the effectiveness of rational block Arnoldi method with our adaptive choice of shifts denoted by `ARAM`. We compared to the rational block Arnoldi method `RAM` with *a priori* choice of shifts using the Matlab function `lp_para` [72], the iterative rational Krylov algorithm `IRKA` method proposed in [41] and with the method `RKSM` introduced in [28]. All

TABLE 2.1 – Information for the test problems.

Matrix A	Size n	$\|A\|_F$	cond(A)
fdm	2500	2.9996e+005	1.0235e+003
fom	1006	1.8283e+04	1000
beam	348	5.6430e+003	3.7420e+007
CDplayer	120	2.3095e+05	1.8149e+04

the experiments were performed on a 1.3GHz Intel Core i5 laptop with 8Gb of RAM. The algorithms were coded in Matlab R2010a. For all the tests, we set $B = C^\top = rand(n, p)$. We used various benchmark matrices as reported in Table 2.1.

We first compared our proposed `ARAM` method , with the Rational Arnoldi Method `RAM` for wich we used *an priori* choice of shifts calculated by the routine `lp_para` from [72]. In the first experiment, we considered the `fom` model and we compared `ARAM` and the rational block Arnoldi when using the shifts computed via `lp_para` with $m = 8$ and 16. In the second experiment, we compared

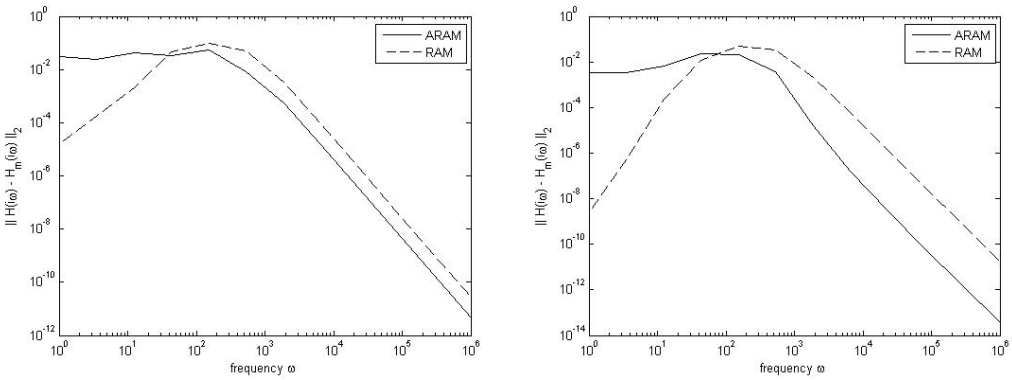

FIGURE 2.1 – The `fom` model : Comparison of `ARAM` and `RAM` with `lp_para`. The error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ for $\omega \in [1, 10^6]$ with $m = 8$ (left) and $m = 16$ (right) (p=3).

the performances of `ARAM` and `IRKA` for the `fdm` model. In Figure 2.2, we plotted the curve corresponding to the errors for the norm of the transfer functions for the method `IRKA` and `ARAM`. For this experiment, we considered the `fdm` model from Table 2.1 where the matrix A is of dimension $n = 2500$. The algorithm `IRKA` starts with a set of parameters chosen randomly as suggested in [41]. We also compared the performance of `ARAM` to the recent rational Krylov subspace method

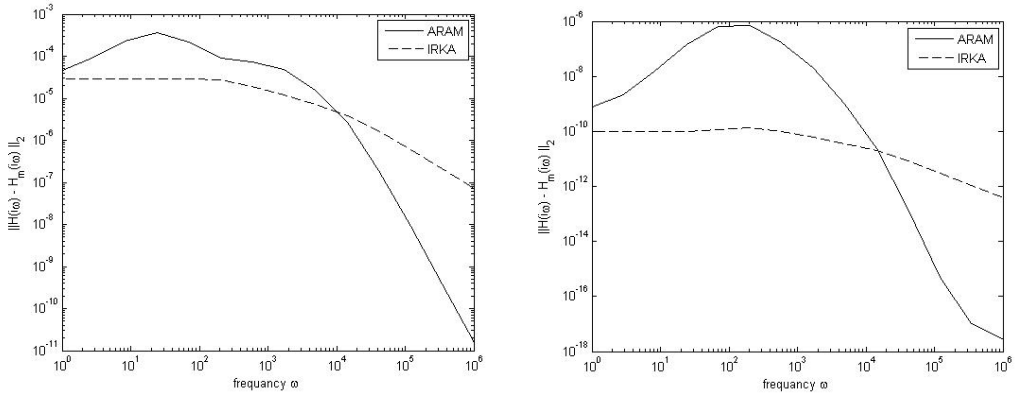

FIGURE 2.2 – The `fdm` model : Comparison of ARAM and IRKA. The error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ for $\omega \in [1, 10^6]$ with $m = 8$ (left) and $m = 16$ (right).

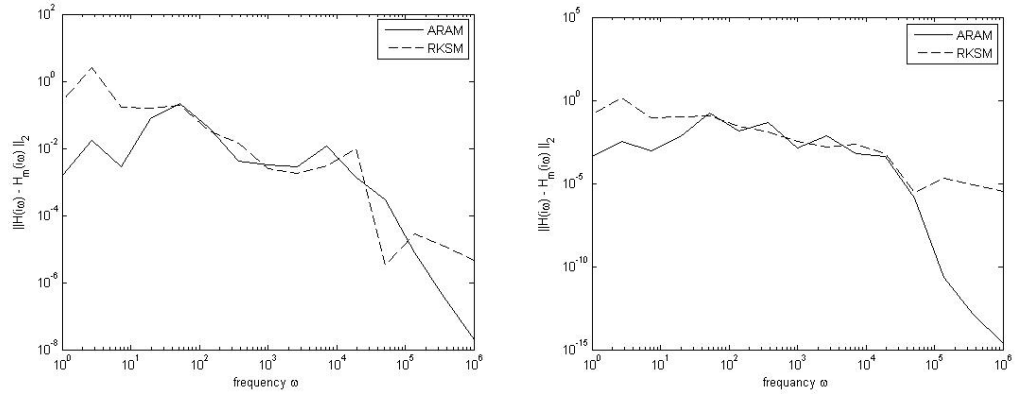

FIGURE 2.3 – The `CDplayer` model. Comparison of `ARAM` and `RKSM`. The error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ for $\omega \in [1, 10^6]$ with $m = 8$ (left) and $m = 16$ (right).

`RKSM` developed in [28] for SISO systems ($p = 1$). In this example we consider the `CDplayer` model. The method `RKSM` starts with the two input shifts : $s_0^{(0)} = 10^{-1}$ and $s_0^{(1)} = 800 + i5.10^4$ as suggested in [28] and the obtained results are shown in Figure 2.3.

For our last experiment, we considered the adaptive rational Arnoldi algorithm with the modified version as described in Algorithm 2. This algorithm will be named Modified Adaptive Rational Block Arnoldi Method `MARAM`. As a test model, we used the `beam` model from Table 2.1 and we set $m = 5$ and $p = 3$. The plots in Figure 2.4 show the original system $\sigma_{max}(H(i\omega))$ and its approximation $\sigma_{max}(H_m(i\omega))$ (left plot), and the associated exact error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ for $\omega \in [1, 10^6]$.

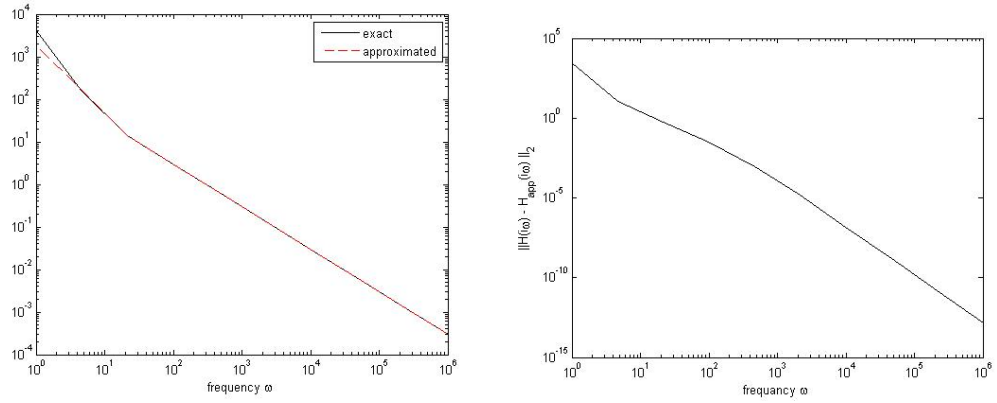

FIGURE 2.4 – The `beam` model : Left : $\|H(i\omega)\|_2$ and it's approximation $\|H_m(i\omega)\|_2$. Right : the exact error $\|H(i\omega) - H_m(i\omega)\|_2$ for $\omega \in [1, 10^6]$ with $m = 5$ and $p = 3$.

2.6 Conclusion

In the present chapter, we considered new projection methods for model reduction in large scale linear dynamical systems. The proposed methods are Krylov subspace type methods based on the rational block Arnoldi algorithm. We proposed a new procedure for selecting good parameter shifts needed in the proposed rational algorithm and we also give some new algebraic relations. A modified version of the rational block Arnoldi algorithm was also proposed and new simple Arnoldi-like relations were developed. The numerical results show that the method is very attractive for sparse problems.

CHAPITRE 3

GLOBAL RATIONAL ARNOLDI METHOD FOR MODEL REDUCTION

3.1 Introduction

Consider the multi-input multi-output (MIMO) linear time-invariant (LTI) system described by the state-space equations

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

where $x(t) \in \mathbb{R}^n$ denotes the state vector and $u(t)$, $y(t) \in \mathbb{R}^p$ respectively denote the input and output vectors of the system (3.1), where the dimension n of the state-space is called the order of the system (3.1). The matrix $A \in \mathbb{R}^{n\times n}$ is assumed to be large and sparse, and $B,\ C^T \in \mathbb{R}^{n\times p}$.

This class of systems arise in many modeling or control design of linear problems (power grids, heat transfer etc) or as a linearization of a nonlinear model. In many applications, the order n of such systems is too large to allow a simulation of the modeled physical problem. It is then necessary to derive an approximate model from the original system

$$\begin{cases} \dot{x}_m(t) &= A_m\,x_m(t) + B_m\,u(t) \\ y_m(t) &= C_m\,x_m(t), \end{cases} \tag{3.2}$$

such as $A_m \in \mathbb{R}^{m\times m}$, $B_m,\ C_m^T \in \mathbb{R}^{m\times p}$, $x_m(t),\ y_m(t) \in \mathbb{R}^m$, and $m \ll n$, while preserving the most relevant properties and the structure of the original system (3.1) such as stability, passivity, moments matching etc.

Many existing model reduction methods such as Padé approximation [30, 80], balanced truncation [69, 70], optimal Hankel norm [35, 36], Krylov projection me-

thods, and in particular the Arnoldi algorithm [24, 25, 32, 48, 49, 52, 67], exploit the sparsity of the large-scale model and have been extensively used for model reduction of large-scale systems; see [9, 21, 33]. Amongst all listed methods, let us focus on the standard global Krylov subspace projection method. The projection space is defined as the matrix Krylov subspace $\mathcal{K}_m(A, B) = \text{span}\{B, AB, \ldots, A^{m-1}B\}$ generated by the the matrices $B, AB, \ldots, A^{m-1}B$. By projecting the initial model (3.1) onto the global Krylov subspace $\mathcal{K}_m(A, B)$, it is possible to obtain a sufficiently accurate reduced system with a moderate space dimension. Unfortunately, this method tends to create reduced order models that poorly approximate some frequency dynamics.

In this chapter, we will consider the rational global Krylov subspace defined by

$$\mathbf{K}_m(A, B) = \text{span}\{B, (A - s_2 I)^{-1}B, \ldots, \prod_{i=2}^{m}(A - s_i I)^{-1}B\} \subset \mathbb{R}^{n\times p}, \qquad (3.3)$$

where $s_2, \ldots, s_m$ are some selected complex parameters. The rational Krylov method (with $s = 1$) was originally proposed by Ruhe [74] in the context of approximating interior eigenvalues of large matrices and is now widely used in model reduction. Indeed, the projection onto rational Krylov subspaces is particularly well suited for approximating the behavior of the transfer function on the imaginary axis [37]. The accuracy of reduced order models can be greatly improved by using this class of subspaces and it is now recognized as a powerful tool within model reduction of linear dynamical systems. Many efforts have been done to compute good shift parameters, see [27, 28, 32, 34, 41, 42, 45, 64]. Nevertheless, rational Krylov methods still suffer from some issues that are summarized below.

• The selection of good shifts is crucial for the quality of the approximation. The success of rational Krylov methods has been hindered by the lack of a parameter-free procedure, which would effectively generate the sequence of shifts used to build the space. Therefore a key issue is the construction of a set of shifts.

• As for any Krylov subspace method, a set of identities known as the Arnoldi relations are satisfied and are used to compute error bounds, residuals, stop tests and to perform perturbation analysis. In the case of rational Krylov subspaces, some relations have been established in the literature; see [27, 76]. However these identities are much more complex in the rational case when compared to the standard Arnoldi equations.

The contributions of this chapter are the design of a modified rational global Arnoldi algorithm, its applications to the model order reduction of large scale sys-

tems and some results regarding the issues discussed above. We will develop a new modified version of the rational global Arnoldi method and its two-sided version allowing us to obtain good low order reduced system and develop new Arnoldi like relations. We also give some adaptive shift selections in order to approach efficiently the initial transfer function by a low order one.

The present chapter is organized as follows. Section 2 is a review of some background results on the linear algebra tools that will be used throughout this work. In Section 3, we propose a modified version of the global rational Arnoldi algorithm and propose some new Arnoldi relations. These results are then applied to model order reduction, addressing in particular the issue of shifts selection and error analysis on the transfer function. In Section 4, we propose a two-sided projection method based on the bi-orthonormal modified global Arnoldi algorithm. Some numerical tests are performed in Section 5 in order to illustrate the performances of those new approaches.

3.2 Preliminaries

In this section, we review some notations and definitions which are used throughout this chapter. For two matrices X and Y in $\mathbb{R}^{n\times p}$, we define the Frobenius inner product $\langle X, Y\rangle_F = \text{Tr}(X^\top Y)$ where $\text{Tr}(X^\top Y)$ denotes the trace of the square matrix $X^\top Y$. The associated Frobenius norm is given by $\|Y\|_F = \text{Tr}(Y^T Y)^{\frac{1}{2}}$. A sequence $V_1, V_2, \ldots, V_m$ of elements of $\mathbb{R}^{n\times p}$ is said to be $F-$orthonormal if it is orthonormal with respect to the inner product $\langle .\ , .\rangle_F$, 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 stocking the columns of Y. For two matrices X and Y, $X \otimes Y = [x_{i,j}Y]$ denotes the Kronecker product of the matrices X and Y. In the sequel, we give some properties of the Kronecker product.

1. $(A \otimes B)^\top = A^\top \otimes B^\top$.
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. $\text{vec}(A)^\top \text{vec}(B) = \text{Tr}(A^\top B)$.

Definition 3.1 *[18] Let $A = [A_1, \ldots, A_s]$ and $B = [B_1, \ldots, B_l]$ be matrices of dimension $n \times sp$ and $n \times lp$, respectively, where A_i and B_j are in $\mathbb{R}^{n\times p}$. Then the*

$s \times l$ matrix $A^\top \diamond B$ is defined by :

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

Remark 3.1 *The following relations were established in [18].*

1. *The matrix $A = [A_1, \ldots, A_s]$ 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. $A^\top \diamond \big(B(L \otimes I_p)\big) = (A^\top \diamond B)L$.
5. $\|A^\top \diamond B\|_F \le \|A\|_F \|B\|_F$.

3.3 The rational global Arnoldi algorithm

The global Krylov method was first proposed in [54] for solving linear equations with multiple right hand sides and Lyapunov equations; see also [48, 60]. Applications to model order reduction are studied in [16, 21, 48]. It is known to be more efficient than block Arnoldi algorithm for model reduction [16]. Unfortunately, up to the authors' knowledge, the existing literature does not provide simple Arnoldi-type relations and as a consequence there are no results on the error of the approximation by a reduced-order model obtained by projecting a dynamical linear system onto a global rational Krylov space.

Let $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$, m be a fixed integer and $s_2, \ldots, s_m$ are chosen complex numbers. We define the global rational Krylov subspace as follows

$$\mathcal{K}_m(A, B) = span\{B, (A - s_2 I)^{-1} B, \ldots, \prod_{i=2}^{m} (A - s_i I)^{-1} B\}. \tag{3.4}$$

It is the subspace spanned by the matrices $B, (A - s_2 I)^{-1} B, \ldots, \prod_{i=2}^{m} (A - s_i I)^{-1} B$. Let the $n \times mp$ block matrix $K_m = [B \ (A - s_2 I)^{-1} B \ \ldots \ \prod_{i=1}^{m} (A - s_i I)^{-1} B]$, then

$$\begin{aligned} Z \in \mathcal{K}_m(A, B) \quad &\Leftrightarrow \quad Z = \alpha_1 B + \sum_{j=2}^{m} \alpha_j \prod_{i=2}^{j} (A - s_j I)^{-1} B, \qquad \alpha_j,\ j = 1, \ldots, m, \\ &\Leftrightarrow \quad Z = K_m(\alpha \otimes I_p), \qquad \alpha = (\alpha_1, \ldots, \alpha_m)^\top \in \mathbb{R}^m. \end{aligned}$$

We now describe the rational global Arnoldi algorithm for computing an F-orthonormal basis of the rational global Krylov subspace (3.4).

Algorithm 3.1 The Rational Global Arnoldi (RGA) Algorithm

— *Input :* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$ and a fixed integer m.

— Compute $V_1 = B/\|B\|_F$, $\mathcal{V}_1 = [V_1]$.

— For $j = 1, \ldots, m-1$

1. $\widetilde{V}_{j+1} = (A - s_{j+1}I)^{-1}V_j$.
2. Orthogonalization step :
   For $i = 1, 2, \ldots, j$
   $h_{i,j} = \langle V_i, \widetilde{V}_{j+1}\rangle$;
   $\widetilde{V}_{j+1} = \widetilde{V}_{j+1} - h_{i,j}V_i$;
   End For
3. $h_{j+1,j} = \|V_{j+1}\|_F$.
4. $V_{j+1} = \widetilde{V}_{j+1}/h_{j+1,j}$.
5. $\mathcal{V}_{j+1} = [\mathcal{V}_j,\ V_{j+1}]$.

— End For.

The shifts $s_2, \ldots, s_m$ are chosen a priori or adaptively during the process. The selection of shifts will be addressed later. In Algorithm 3.1, we compute a set of $F-$orthonormal block vectors $\{V_1, \ldots, V_m\}$ i.e.

$$\mathcal{V}_m^\top \diamond \mathcal{V}_m = I_m,$$

where $\mathcal{V}_m$ is the $n\times mp$ matrix $\mathcal{V}_m = [V_1, \ldots, V_m]$ and an $(m+1)\times m$ upper Hessenberg matrix $\overline{H}_m$ whose nonzero entries are the $h_{i,j}$ elements defined in Algorithm 3.1. The $m \times m$ upper Hessenberg matrix H_m is obtained from $\overline{H}_m$ by deleting its last row. We now state some new algebraic relations derived from Algorithm 3.1.

Proposition 3.1 *Let us consider the matrices $\mathcal{V}_m$, $\overline{H}_m$ and H_m as defined by the rational global Arnoldi algorithm assuming that H_m is nonsingular. Let $\mathcal{D}_m$ be the $m \times m$ diagonal matrix $diag(s_2, \ldots, s_{m+1})$ where $\{s_2, ..., s_{m+1}\}$ is the set of shifts used in the algorithm. Then we have the following relations*

$$\mathcal{T}_m = \left[I_m + H_m\mathcal{D}_m - h_{m+1,m}(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top\right]H_m^{-1}. \tag{3.5}$$

And

$$A\mathcal{V}_m = \mathcal{V}_m(\mathcal{T}_m \otimes I_p) + h_{m+1,m}\mathbb{J}_m(H_m^{-1} \otimes I_p), \tag{3.6}$$

where $\mathbb{J}_m = \left[\mathcal{V}_m(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top) \otimes I_p + V_{m+1}E_m^T\mathcal{S}_m - AV_{m+1}E_m^T\right]$
and $\mathcal{S}_m = \mathcal{D}_m \otimes I_p$.

Proof After m steps of the rational global Arnoldi algorithm, we have

$$(A - s_{j+1}I_n)^{-1}V_j = \sum_{i=1}^{j+1} h_{i,j}V_i \qquad for\ j = 1, \ldots, m$$

then

$$V_j = A(\sum_{i=1}^{j+1} h_{i,j}V_i) - s_{j+1}(\sum_{i=1}^{j+1} h_{i,j}V_i) \qquad for\ j = 1, \ldots, m.$$

We can then notice that

$$\mathcal{V}_m = A\big(\mathcal{V}_{m+1}(\overline{H}_m \otimes I_p)\big) - \big(\mathcal{V}_{m+1}(\overline{H}_m \otimes I_p)\big)\mathcal{S}_m, \tag{3.7}$$

which can also be written as

$$\mathcal{V}_m = A\big(\mathcal{V}_m(H_m \otimes I_p) + h_{m+1,m}V_{m+1}E_m^T\big) - \big(\mathcal{V}_m(H_m \otimes I_p) + h_{m+1,m}V_{m+1}E_m^T\big)\mathcal{S}_m,$$

where $E_m^T = [0_p, \ldots, 0_p, I_p] = (e_m^T \otimes I_p)$.

Applying the $\diamond$ product on the left by $\mathcal{V}_m^T$ to the last equality and using the diamond and Kronecker properties described in Section 1, we get the following identity as a consequence of the fact that the blocks $V_1, \ldots, V_{m+1}$ are F$-$orthonormal

$$I_m = (\mathcal{V}_m^\top \diamond A\mathcal{V}_m)H_m + h_{m+1,m}(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top - H_m\mathcal{D}_m.$$

Denoting $\mathcal{T}_m = \mathcal{V}_m^\top \diamond A\mathcal{V}_m$, we can deduce the relation

$$\mathcal{T}_m = (I_m + H_m\mathcal{D}_m - h_{m+1,m}(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top)H_m^{-1},$$

which proves the relation (3.5).
From the relation (3.7), we get

$$\begin{aligned} A\mathcal{V}_m(H_m \otimes I_p) &= \mathcal{V}_m - h_{m+1,m}AV_{m+1}E_m^T + \mathcal{V}_m(H_m \otimes I_p)\mathcal{S}_m \\ &+ h_{m+1,m}V_{m+1}E_m^T\mathcal{S}_m \qquad (3.8) \\ &= \mathcal{V}_m(I_{mp} + (H_m \otimes I_p)\mathcal{S}_m) - h_{m+1,m}AV_{m+1}E_m^T \\ &+ h_{m+1,m}V_{m+1}E_m^T\mathcal{S}_m. \qquad (3.9) \end{aligned}$$

Then, using (3.5) we obtain

$$I_{mp} + (H_m \otimes I_p)\mathcal{S}_m = \left(\mathcal{T}_m H_m + h_{m+1,m}(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top\right) \otimes I_p.$$

Replacing the latter in (3.9), we have

$$\begin{aligned} A\mathcal{V}_m(H_m \otimes I_p) &= \mathcal{V}_m\Big((\mathcal{T}_m H_m + h_{m+1,m}(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top) \otimes I_p\Big) \\ &- h_{m+1,m}AV_{m+1}E_m^T + h_{m+1,m}V_{m+1}E_m^T\mathcal{S}_m. \end{aligned}$$

Therefore

$$\begin{aligned} A\mathcal{V}_m(H_m \otimes I_p) &= \mathcal{V}_m(\mathcal{T}_m \otimes I_p)(H_m \otimes I_p) + h_{m+1,m}\big[\mathcal{V}_m(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top \otimes I_p \\ &+ V_{m+1}E_m^T\mathcal{S}_m - AV_{m+1}E_m^T\big]. \end{aligned}$$

Finally, we have

$$\begin{aligned} A\mathcal{V}_m = \mathcal{V}_m(\mathcal{T}_m \otimes I_p) &+ h_{m+1,m}\Big[(\mathcal{V}_m(\mathcal{V}_m^T \diamond AV_{m+1})e_m^\top) \otimes I_p + V_{m+1}E_m^T\mathcal{S}_m \\ &- AV_{m+1}E_m^T\Big] \times (H_m \otimes I_p)^{-1}, \end{aligned}$$

which proves (3.6) and ends the proof.

3.4 The modified adaptive rational global Arnoldi method

The relations of Proposition 3.1 known as the Arnoldi relations are useful for establishing error or residual bounds and perturbation analysis. In the SISO case, similar identities have been already stated in the literature in the rational case [27, 28, 76], however they are far from simple compared to the standard Arnoldi case. The main contribution of this section is to propose some simpler rational global Arnoldi relations. For this purpose, we need to define a generalization of the rational global Krylov subspace, allowing some shifts to be equal to infinity. At each step $j+1$, The Modified Rational Global Arnoldi Algorithm 3.2 generates a new block $\widetilde{V}_{j+1} = (A - s_{j+1}I)^{-1}V_j$ if s_{j+1} is finite and $\widetilde{V}_{j+1} = AV_j$ if $s_{j+1} = \infty$.
The adaptive modified rational global Arnoldi algorithm is summarized as follows

Algorithm 3.2 The Modified Adaptive Rational Global Arnoldi (`MARGA`) Algorithm

— *Input :* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$ and a fixed integer m.

— Compute $V_1 = B/\|B\|_F$, $\mathcal{V}_1 = [V_1]$.

— For $j = 1, \ldots, m$

1. Choose the next shift s_{j+1} by using (3.20) .
2. Set $\widetilde{V}_{j+1} = \begin{cases} (A - s_{j+1}I)^{-1}V_j, & if \quad s_{j+1} \neq \infty \\ AV_j, & if \quad s_{j+1} = \infty \end{cases}$
3. Orthogonalization step :
   For $i = 1, 2, \ldots, j$
   $h_{i,j} = \langle V_i, \widetilde{V}_{j+1}\rangle_F$;
   $\widetilde{V}_{j+1} = \widetilde{V}_{j+1} - h_{i,j}V_i$;
   End For
4. $h_{j+1,j} = \|V_{j+1}\|_F$.
5. $V_{j+1} = \widetilde{V}_{j+1}/h_{j+1,j}$.
6. $\mathcal{V}_{j+1} = [\mathcal{V}_j,\ V_{j+1}]$.

— End For.

The idea of including infinity as a possible interpolation point could also be considered as a generalization of the extended block Arnoldi algorithm [46, 81]. Using this modified version of the global rational Arnoldi algorithm, we can state the following Arnoldi relations.

Proposition 3.2 *Let* $\mathcal{S} = \{s_2, \ldots, s_m\} \subset \mathbb{C}$ *and* $\mathcal{V}_{m+1} = [V_1, \ldots, V_{m+1}] \in \mathbb{R}^{n\times(m+1)p}$ *be generated by Algorithm 3.2 for one extra interpolation point at* $s_{m+1} = \infty$.
Then the following Arnoldi-like relations are satisfied

$$A\mathcal{V}_m = \mathcal{V}_{m+1}(\overline{\mathcal{T}}_m \otimes I_p) \tag{3.10}$$

$$= \mathcal{V}_m(\mathcal{T}_m \otimes I_p) + V_{m+1}(N_m \otimes I_p), \tag{3.11}$$

where $\overline{\mathcal{T}}_m = \mathcal{V}_{m+1}^\top \diamond A\mathcal{V}_m$, $\mathcal{T}_m = \mathcal{V}_m^\top \diamond A\mathcal{V}_m$ *and* $N_m = V_{m+1}^\top \diamond A\mathcal{V}_m$.

Proof From Algorithm 3.2, we have

$$Span\{V_1, \ldots, V_m, AV_m\} = Span\{V_1, \ldots, V_{m+1}\}, \text{ and } \mathcal{V}_{m+1}^{\top} \diamond \mathcal{V}_{m+1} = I_{(m+1)}.$$

First we need to prove that

$$Span\{AV_1, \ldots, AV_m\} \subset Span\{V_1, \ldots, V_{m+1}\}.$$

Indeed, after $m-1$ iterations of the rational Arnoldi algorithm, the proof of Proposition 3.1 gives us

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

Therefore

$$\begin{aligned} A\mathcal{V}_{m-1} &= \mathcal{V}_{m-1}(H_{m-1}^{-1} \otimes I_p) - AV_m\big((h_{m,m-1} e_{m-1}^T H_{m-1}^{-1}) \otimes I_p\big) \\ &+ \mathcal{V}_{m-1}(\mathcal{S}_{m-1} \otimes I_p) + V_m\big((h_{m,m-1} s_m e_{m-1}^T H_{m-1}^{-1}) \otimes I_p\big). \end{aligned}$$

Using the fact that $AV_m \in Span\{V_1, \ldots, V_{m+1}\}$,
it is clear that $Span\{AV_1, \ldots, AV_{m-1}\} \subset Span\{V_1, \ldots, V_{m+1}\}$,
and $Span\{AV_1, \ldots, AV_m\} \subset Span\{V_1, \ldots, V_{m+1}\}$.
Therefore we have

$$A\mathcal{V}_m = \mathcal{V}_{m+1}(\overline{\Gamma}_m \otimes I_p), \tag{3.12}$$

for some matrix $\overline{\Gamma}_m$. Since $\mathcal{V}_{m+1}^{\top} \diamond \mathcal{V}_{m+1} = I_{(m+1)}$, applying the left $\diamond$ product by $\mathcal{V}_{m+1}^T$ to (3.12), we get

$$\overline{\Gamma}_m = \mathcal{V}_{m+1}^{\top} \diamond A\mathcal{V}_m.$$

We can also notice that

$$A\mathcal{V}_m = \mathcal{V}_m(\Gamma_m \otimes I_p) + V_{m+1}(N_m \otimes I_p), \tag{3.13}$$

for some matrices Γ_m and N_m Hence, applying the left $\diamond$ product by $\mathcal{V}_{m+1}^T$ to (3.13), we get

$$\Gamma_m = \mathcal{T}_m = \mathcal{V}_m^{\top} \diamond A\mathcal{V}_m,$$

and applying the left $\diamond$ product by $V_{m+1}^{\top}$ to (3.13), we have

$$N_m = V_{m+1}^{\top} \diamond A\mathcal{V}_m, \tag{3.14}$$

which completes the proof.

Next, we will show how to apply these algorithms to obtain reduced model order dynamical systems. This will be done by approximating the transfer function corresponding to the original dynamical system.

3.5 Model reduction, transfer functions and adaptive selection of the shifts

3.5.1 The reduced model

The main purpose of this section is the computation of a reduced order model (3.2) approaching the original model (3.1) by using rational global Krylov subspace techniques. We also propose an adaptive technique for computing the shifts that are used to build the rational global Krylov subspace. This procedure automatically generates the sequence of shifts during the construction of the rational global Arnoldi subspaces. We recall the linear time-invariant (LTI) multi-input and multi-output (MIMO) system (3.1)

$$\left\{ \begin{array}{rcl} \dot{x}(t) & = & A\,x(t) + B\,u(t) \\ y(t) & = & C\,x(t). \end{array} \right. \tag{3.15}$$

A classical way of relating the input to output is to use the transfer function (or impulse response in the time domain) of the LTI system (3.1). Indeed, applying the Laplace transform

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

to the dynamical system (3.1), we obtain the relation

$$\left\{ \begin{array}{rcl} s\,X(s) & = & A\,X(s) + B\,U(s) \\ Y(s) & = & C\,X(s) \end{array} \right. ,$$

where $X(s)$, $Y(s)$ and $U(s)$ are the Laplace transforms of $x(t)$, $y(t)$ and $u(t)$, respectively. Eliminating $X(s)$ in the previous two equations, we get

$$Y(s) = H(s)\,U(s),$$

where the function H is given by

$$H(s) = C\,(s\,I_n - A)^{-1}\,B. \tag{3.16}$$

The rational function $H(s)$ is called the transfer function of the system (3.1). We recall that most model order reduction techniques, for example the moment-matching approaches, are based on the approximation of this transfer function; for more details, see [10, 32, 38] and the references therein. If the number of state variables is

very large, it would be very difficult to use the full system for simulation or run-on-time control. So it is necessary to look for lower order models that approximate the behavior of the original models. This will be done by approximating the transfer function (3.16).

Let us write $H(s) = C\,X$ where $X \in \mathbb{R}^{n\times p}$ is the solution of the matrix linear system (assumed to be nonsingular)

$$(sI_n - A)X = B. \tag{3.17}$$

In order to approximate the transfer function H, we will look for approximations of the solution X of the multiple linear system (3.17). Let X_m denote the approximate solution obtained by the projection method onto the rational Krylov subspace $\mathcal{K}_m(A, B)$. This approximate solution is written as the product

$$X_m = \mathcal{V}_m(y_m \otimes I_p). \tag{3.18}$$

Enforcing the orthogonality constraint $R_m \perp_F \mathcal{K}_m(A, B)$, where R_m is the residual $R_m = B - (sI - A)X_m$ corresponding to the approximate solution X_m, we get

$$\mathcal{V}_m^\top \diamond [B - (sI - A)X_m] = 0,$$

and then

$$\mathcal{V}_m^\top \diamond B - \mathcal{V}_m^\top \diamond [(sI - A)\mathcal{V}_m(y_m \otimes I_p)] = 0.$$

Using a property of the $\diamond$-product (Remark 3.1) and the fact that $B = \|B\|_F V_1$, we obtain

$$[\mathcal{V}_m^\top \diamond (sI - A)\mathcal{V}_m]y_m = \|B\|_F e_1,$$

where $e_1 = [1, 0, \ldots, 0]^\top \in \mathbb{R}^m$. Denoting $\mathcal{T}_m = \mathcal{V}_m^\top \diamond A\mathcal{V}_m$, we obtain

$$(sI_m - \mathcal{T}_m)y_m = \|B\|_F e_1,$$

which yields

$$y_m = (sI_m - \mathcal{T}_m)^{-1}\|B\|_F e_1.$$

Replacing the last identity in (3.18), we obtain

$$X_m = \mathcal{V}_m[((sI_m - \mathcal{T}_m)^{-1}\|B\|_F e_1) \otimes I_p],$$

and using the Kronecker product's properties, we get

$$X_m = \mathcal{V}_m[((sI_{mp} - \mathcal{T}_m \otimes I_p)^{-1}\|B\|_F E_1)],$$

where $E_1 = e_1 \otimes I_p$. We can therefore express the approximate transfer function as

$$H_m(s) = C\,X_m = C\mathcal{V}_m[((sI_{mp} - \mathcal{T}_m \otimes I_p)^{-1}\|B\|_F E_1)].$$

This suggests that the reduced order model to (3.1) can be expressed as

$$\begin{cases} \dot{x}_m(t) &= A_m\, x_m(t) + B_m\, u(t) \\ y_m(t) &= C_m\, x_m(t). \end{cases}$$

where $A_m = \mathcal{T}_m \otimes I_p \in \mathbb{R}^{mp\times mp}$, $B_m = \|B\|_F E_1 \in \mathbb{R}^{mp\times p}$ and $C_m = C\,\mathcal{V}_m \in \mathbb{R}^{p\times mp}$. Moreover, the transfer function of the previous reduced order model is given by

$$H(s) = C_m\,(s\,I_{mp} - A_m)^{-1}\,B_m. \tag{3.19}$$

3.5.2 Criterion-selections of the shift parameters

In the sequel, we will give expressions for the norm of the error $H(s) - H_j(s)$, at the iteration j, that will be used for the selection of our shift parameters. First, let us recall the $\mathcal{H}_\infty$ norm of a matrix-valued function

$$\|H\|_\infty = \sup_{y\in\mathbb{R}} \sigma_{max}(H(iy)).$$

Noticing that

$$\begin{aligned} H(s) - H_j(s) &= CX - CX_j \\ &= C(sI_n - A)^{-1}B - CX_j \\ &= C(sI_n - A)^{-1}[B - (sI_n - A)X_j], \end{aligned}$$

we obtain

$$\|H(s) - H_j(s)\|_\infty \leq \|C(sI_n - A)^{-1}\|_\infty \|\Gamma_j\|_\infty,$$

where $\Gamma_j = B - (sI_n - A)X_j$. This suggests selecting each step a new shift in order to reach $\|\Gamma_j\|_\infty$. Hence, our new shift s_{j+1} will be chosen such that

$$\sigma_{max}(\Gamma_j(\mathbf{i}s_{j+1})) = \sup_{y\in\mathbb{R}} \sigma_{max}(\Gamma_j(\mathbf{i}y)) = \|\Gamma_j\|_\infty. \tag{3.20}$$

As we will see in the numerical tests, this simple procedure gives good results.

3.6 A modified two-sided global rational Arnoldi method

In this section, we will define a new two-sided global rational method which is based on an oblique projection considering the two rational global Krylov subspaces

$$\mathcal{K}_m(A,B) = Span\{B, (A-s_2I)^{-1}B, \ldots, \prod_{i=2}^{m}(A-s_iI)^{-1}B\}, \tag{3.21}$$

and

$$\mathcal{K}_m(A^T,C^T) = Span\{C^T, (A-s_2I)^{-\top}C^T, \ldots, \prod_{i=2}^{m}(A-s_iI)^{-\top}C^T\}. \tag{3.22}$$

Let $\{V_1,\ldots,V_m\}$ and $\{W_1,\ldots,W_m\}$ denote the $F-$orthonormal bases of $\mathcal{K}_m(A,B)$ and $\mathcal{K}_m(A^T,C^T)$ respectively, generated by the Modified Rational Global Arnoldi Algorithm 3.2 for one extra interpolation point at $s_{m+1}=\infty$. Let us define the two $n\times mp$ matrices $\mathcal{V}_m=[V_1,\ldots,V_m]$ and $\mathcal{W}_m=[W_1,\ldots,W_m]$. In the sequel, for both global Krylov subspace (3.21) and (3.22), we still consider the same bases and same matrices as defined above.

We set $\mathcal{Z}_m = \mathcal{W}_m\big((\mathcal{V}_m^\top \diamond \mathcal{W}_m)^{-1}\otimes I_p\big)$, and using the diamond product properties, we get

$$\mathcal{V}_m^\top \diamond \mathcal{Z}_m = \mathcal{V}_m^\top \diamond \Big[\mathcal{W}_m\big((\mathcal{V}_m^\top \diamond \mathcal{W}_m)^{-1}\otimes I_p\big)\Big] = (\mathcal{V}_m^\top \diamond \mathcal{W}_m)(\mathcal{V}_m^\top \diamond \mathcal{W}_m)^{-1} = I_m.$$

Therefore, we can state the following relation

$$\mathcal{Z}_m^\top \diamond \mathcal{V}_m = \mathcal{V}_m^\top \diamond \mathcal{Z}_m = I_m.$$

Projection operators play an important role in numerical linear algebra, particularly in iterative methods for solving various matrix problems. These projections have been defined for classical and block Krylov subspaces, however, in the global Krylov case, it is still an open question up to the authors' knowledge. In the sequel, we define the projection operator onto the global Krylov subspace.

Definition 3.2 *Let $\mathcal{V}_m$ be the $n\times mp$ matrix $\mathcal{V}_m=[V_1,\ldots,V_m]$ whose blocks form an F-orthonormal basis of the rational global Krylov subspace $\mathcal{K}_m(A,B)$, and let $\mathcal{L}$ be the subspace spanned by the columns of the $n\times mp$ matrix $\mathcal{Z}_m=[Z_1,\ldots,Z_m]$. We define the projector P onto $\mathcal{K}_m(A,B)$ and orthogonal to the subspace $\mathcal{L}$ by :*

$$P(M) = \mathcal{V}_m\big((\mathcal{Z}_m^\top \diamond M)\otimes I_p\big);\ \forall M\in\mathbb{R}^{n\times p}. \tag{3.23}$$

Our aim here is to use this projector to define a new reduced-order system with the corresponding transfer function that should approach the transfer function (3.16). Let X_m in (3.18) denote the approximate solution obtained by the projection method onto the rational Krylov subspace $\mathcal{K}_m(A, B)$. By following the same technique as in Section 3.5, where we impose that $R_m \perp_F \mathcal{L}$, where $R_m = B - (sI - A)X_m$ denotes the residue of the system (3.17), i.e.

$$\mathcal{Z}_m^\top \diamond [B - (sI - A)X_m] = 0.$$

We define the approximate transfer function by

$$H_m(s) = C\mathcal{V}_m(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big), \tag{3.24}$$

where $Y_m = \mathcal{Z}_m^\top \diamond A\mathcal{V}_m$.

Proposition 3.3 *Let us consider $\mathcal{Z}_m$ as described above, $D_m = \mathcal{W}_m^\top \diamond \mathcal{V}_m$ and M an $n \times p$ matrix, then we have*

$$\mathcal{Z}_m^\top \diamond M = D_m^{-1}(\mathcal{W}_m^\top \diamond M). \tag{3.25}$$

Proof We have

$$\begin{aligned}
\mathcal{Z}_m^\top \diamond M = (M^\top \diamond \mathcal{Z}_m)^\top &= \Big[M^\top \diamond \Big(\mathcal{W}_m\big((\mathcal{V}_m^\top \diamond \mathcal{W}_m)^{-1} \otimes I_p\big)\Big)\Big]^\top \\
&= \Big[(M^\top \diamond \mathcal{W}_m)(\mathcal{V}_m^\top \diamond \mathcal{W}_m)^{-1}\Big]^\top \\
&= (\mathcal{W}_m^\top \diamond \mathcal{V}_m)^{-1}(\mathcal{W}_m^\top \diamond M) \\
&= D_m^{-1}(\mathcal{W}_m^\top \diamond M).
\end{aligned}$$

Proposition 3.4 *Let $\{V_1, \ldots, V_{m+1}\}$ and $\{W_1, \ldots, W_{m+1}\}$ generated by running the modified rational global Arnoldi algorithm 3.2 (for one extra interpolation point at $s_{m+1} = \infty$) on the two subpaces $\mathcal{K}_m(A, B)$ and $\mathcal{K}_m(A^T, C^T)$, respectively. Then the following equation is satisfied*

$$A\mathcal{V}_m = \mathcal{V}_m\big((D_m^{-1}A_m) \otimes I_p\big) + (I - P)\big(V_{m+1}(N_m \otimes I_p)\big),$$

where $A_m = \mathcal{W}_m^\top \diamond A\mathcal{V}_m$ and P is the projector given by (3.23).

Proof From Proposition 3.2, we have

$$A\mathcal{V}_m = \mathcal{V}_m(\mathcal{T}_m \otimes I_p) + V_{m+1}(N_m \otimes I_p). \tag{3.26}$$

Multiplying on the left by $\mathcal{W}_m$ and using the diamond product property, we obtain

$$\underbrace{\mathcal{W}_m^\top \diamond A\mathcal{V}_m}_{A_m} = \underbrace{(\mathcal{W}_m^\top \diamond \mathcal{V}_m)}_{D_m}\mathcal{T}_m + (\mathcal{W}_m^\top \diamond V_{m+1})N_m,$$

and

$$\mathcal{T}_m = D_m^{-1}A_m - D_m^{-1}(\mathcal{W}_m^\top \diamond V_{m+1})N_m.$$

Replacing this last relation in (3.26), we have

$$\begin{aligned} A\mathcal{V}_m &= \mathcal{V}_m\Big[(D_m^{-1}A_m)\otimes I_p - \big(D_m^{-1}(\mathcal{W}_m^\top \diamond V_{m+1})N_m\big)\otimes I_p\Big] + V_{m+1}(N_m \otimes I_p) \\ &= \mathcal{V}_m\big((D_m^{-1}A_m)\otimes I_p\big) - \mathcal{V}_m\Big(\big(D_m^{-1}(\mathcal{W}_m^\top \diamond V_{m+1})N_m\big)\Big)\otimes I_p \\ &+ V_{m+1}(N_m \otimes I_p). \end{aligned}$$

From Proposition 3.3 and the diamond property, we get

$$\begin{aligned} D_m^{-1}(\mathcal{W}_m^\top \diamond V_{m+1})N_m &= (\mathcal{Z}_m^\top \diamond V_{m+1})N_m \\ &= \mathcal{Z}_m^\top \diamond \big(V_{m+1}(N_m \otimes I_p)\big), \end{aligned}$$

therefore

$$\begin{aligned} \mathcal{V}_m\big(D_m^{-1}(\mathcal{W}_m^\top \diamond V_{m+1})N_m\big)\otimes I_p &= \mathcal{V}_m\Big(\mathcal{Z}_m^\top \diamond \big(V_{m+1}(N_m \otimes I_p)\big)\Big)\otimes I_p \\ &= P\Big(V_{m+1}(N_m \otimes I_p)\Big), \end{aligned}$$

which yields

$$A\mathcal{V}_m = \mathcal{V}_m\big((D_m^{-1}A_m)\otimes I_p\big) + (I-P)\big(V_{m+1}(N_m \otimes I_p)\big).$$

In the next proposition, we give a new expression of the error $H(s) - H_m(s)$ which could be used to compute a new upper bound for the norm of the error on the transfer function.

Proposition 3.5 *Let H be the transfer function defined in (3.16) and let H_m be its approximation obtained by applying the oblique projection, as defined in (3.24). Then the following relation is satisfied*

$$\begin{aligned} H(s) - H_m(s) &= C(sI_n - A)^{-1}(I-P)\big(V_{m+1}(N_m \otimes I_p)\big) \\ &\times \ (sI_{mp} - Y_m \otimes I_p)^{-1}\|B\|_F e_1 \otimes I_p. \end{aligned} \tag{3.27}$$

Proof We have

$$\begin{aligned} H(s) &- H_m(s) = C(sI_n - A)^{-1}B - C\mathcal{V}_m(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big) \\ &= C(sI_n - A)^{-1}\big[B - (sI_n - A)\mathcal{V}_m(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big)\big] \\ &= C(sI_n - A)^{-1}R_B(s), \end{aligned} \tag{3.28}$$

where $R_B(s) = B - (sI_n - A)\mathcal{V}_m(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big)$.
Using Proposition 3.4, we obtain

$$\begin{aligned} R_B(s) &= B - (sI_n - A)\mathcal{V}_m(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big) \\ &= B - (s\mathcal{V}_m - A\mathcal{V}_m)(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big) \\ &= B - \big[s\mathcal{V}_m - \mathcal{V}_m\big((D_m^{-1}A_m) \otimes I_p\big) - (Id - P)\big(V_{m+1}(N_m \otimes I_p)\big)\big] \\ &\times \ (sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big). \end{aligned}$$

Proposition 3.3 gives us

$$D_m^{-1}A_m = D_m^{-1}(\mathcal{W}_m^\top \diamond A\mathcal{V}_m) = \mathcal{Z}_m^\top \diamond A\mathcal{V}_m = Y_m,$$

therefore

$$\begin{aligned} R_B(s) &= B - \big[\mathcal{V}_m\big(sI_{mp} - Y_m \otimes I_p\big) - (Id - P)\big(V_{m+1}(N_m \otimes I_p)\big)\big] \\ &\times \ (sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big) \\ &= B - \mathcal{V}_m\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big) + (Id - P)\big(V_{m+1}(N_m \otimes I_p)\big) \\ &\times \ (sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big). \end{aligned}$$

Notice that, since B belongs to the Krylov subspace $\mathcal{K}_m(A, B)$, we have

$$\mathcal{V}_m\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big) = P(B) = B$$

Therefore, we can write

$$R_B(s) = (I - P)\big(V_{m+1}(N_m \otimes I_p)\big)(sI_{mp} - Y_m \otimes I_p)^{-1}\big((\mathcal{Z}_m^\top \diamond B) \otimes I_p\big).$$

It can also be noticed that

$$(\mathcal{Z}_m^\top \diamond B) \otimes I_p = \|B\|_F(\mathcal{Z}_m^\top \diamond V_1) \otimes I_p = \|B\|_F e_1 \otimes I_p,$$

which gives us the following

$$R_B(s) = (I - P)\big(V_{m+1}(N_m \otimes I_p)\big)(sI_{mp} - Y_m \otimes I_p)^{-1}\|B\|_F e_1 \otimes I_p.$$

By replacing the last relationship in (3.28), we finish the proof.

3.7 Numerical experiments

In this section, we report some numerical examples on some benchmark models. We considered the Modified Adaptive Rational Global Arnoldi (`MARGA`) method and and its two-sided version. This latter will be compared with the Iterative Rational Krylov Algorithm (`IRKA`) [41].
All the experiments were performed on a 1.3GHz Intel Core i5 laptop with 8Gb of RAM. The algorithms were coded in Matlab R2010a and we used various benchmark models reported in Table 3.1.

TABLE 3.1 – Test problems.

Matrix A	Size n	$\|A\|_F$	cond(A)
`fdm`	40.000	3.6118e+007	2.4608e+004
`CDplayer`	120	2.3095e+05	1.8149e+04
`iss`	270	2.0594e+004	9.6800e+003

Example 1. For the first experiment, we considered the `CDplayer` model. This model has a small dimension but, as it is a difficult model, it is always taken as a benchmark example. The plots in Figure 3.1 show the σ-plots : $\sigma_{max}(H(i\omega))$ and its approximation $\sigma_{max}(H_m(i\omega))$ (left plot). In the right-part of this figure, we also plotted the error-norm $\sigma_{max}(H(i\omega) - H_m(i\omega))$ for $\omega \in [10^{-3}, 10^3]$ with $m = 10$ and $p = 2$. The entries of the matrices B and C were random values uniformly distributed on $[0\,,\,1]$.

Example 2. In this example, we considered the well known `iss` model. As in the preceding example, we plotted in the left-side of Figure 3.2, the singular values $\sigma_{max}(H(i\omega))$ for the original transfer function and its approximation $\sigma_{max}(H_m(i\omega))$. The error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ is plotted on the right side of Figure 3.2 versus the frequencies $\omega \in [10^{-3}, 10^3]$ with $m = 10$ and $p = 2$. The matix B and C are those of the model.

Example 3. For this experiment, we considered the `fdm` model from Table 3.1 with a large dimension $n = 40.000$ and $p = 5$. We plotted the $\mathcal{H}_\infty$ norm of the error $\|H - H_m\|_\infty$ versus the number m of iterations. The entries of the matrices B and C were random values uniformly distributed on $[0\,,\,1]$. The results are given in Figure 3.3.

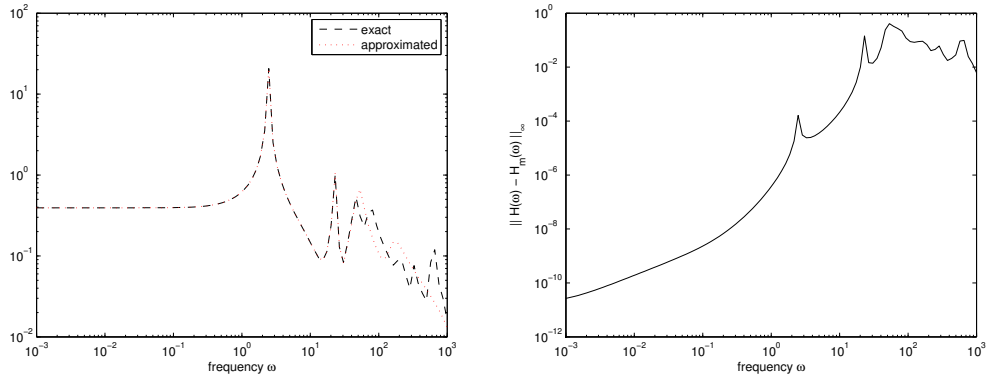

FIGURE 3.1 – The `CDplayer` model : The original system $\|H(i\omega)\|_2$ and it's approximation $\|H_m(i\omega)\|_2$ (left plot) and the error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ (right plot) for $\omega \in [10^{-3}, 10^3]$ with $m = 10$ and p=2.

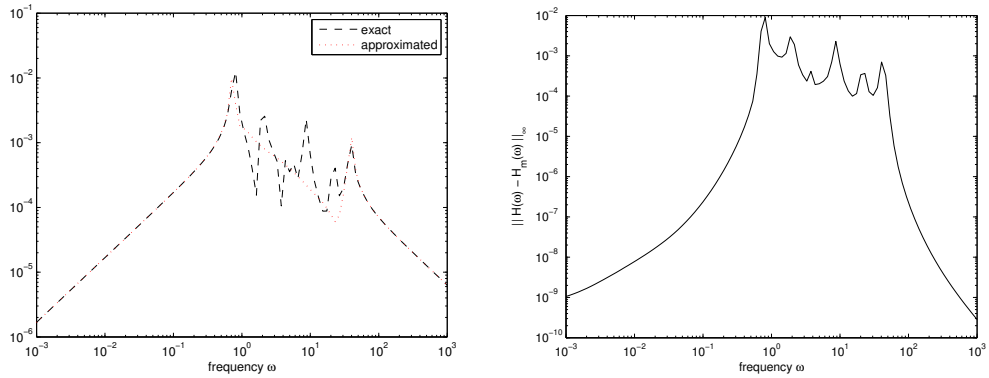

FIGURE 3.2 – The `iss` model : The original system $\|H(i\omega)\|_2$ and it's approximation $\|H_m(i\omega)\|_2$ (left plot) and the error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ (right plot) for $\omega \in [10^{-3}, 10^3]$ with $m = 10$ and $p = 2$.

Example 4. In the last example, we compared the two side `MARGAM` with the block iterative rational Krylov algorithm `IRKA` method [41]. For this experiment, we considered the `CDplayer` model. In Figure 4, we plotted the error norms of the transfer functions with two values of m : $m = 10$ (left) and $m = 20$ (right), with $p = 2$. In this figure, we plotted the errors $\sigma_{max}(H(\omega) - H_m(\omega)$ for `MARGA` (solid line) and `IRKA` (dotted line). As observed from Figure 4, `MARGA` returns good results.

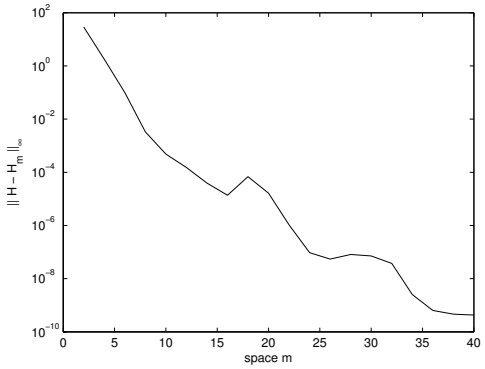

FIGURE 3.3 – The $\mathcal{H}_\infty$ error norms $\|H - H_m\|_\infty$ versus the number m of iterations for the `fdm` model with $p = 5$.

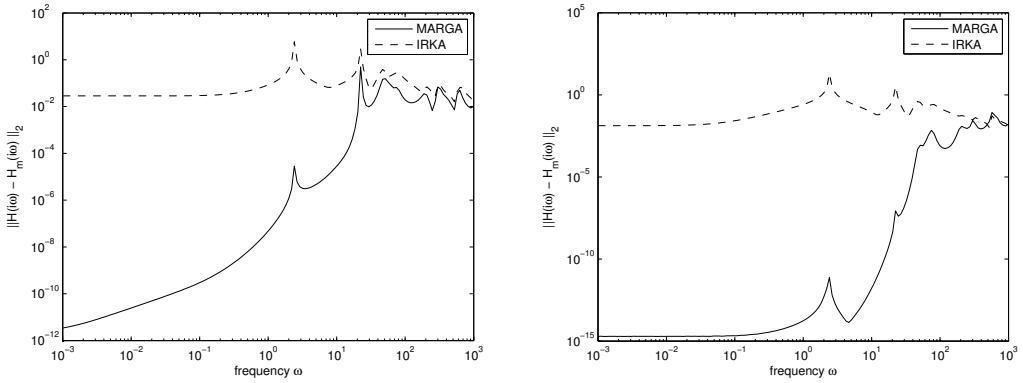

FIGURE 3.4 – The `CDplayer` model : Comparison of (two side) `MARGA` and `IRKA`. The error $\sigma_{max}(H(i\omega) - H_m(i\omega))$ for $\omega \in [10^{-3}, 10^3]$ with $m = 10$ (left) and $m = 20$ (right) (p=2).

3.8 Conclusion

In this chapter, we considered MIMO dynamical systems and we developed new global methods to get low order dynamical systems by projecting the initial large-scale dynamical systems onto a low dimensional space. We develop new modified adaptive rational global Arnoldi method and its two side version to get such a projectors. New Arnoldi like relations were also given generalizing the classical well known Arnoldi relations. Using upper bounds for the norm of error between the initial transfer function and its approximation, obtained from the proposed projections, we proposed some shifts selection techniques to get good approximations on a large frequency domain. The numerical experiments on some benchmark models

show the effectiveness of the proposed approaches.

CHAPITRE 4

ON SOME PROPERTIES OF THE EXTENDED BLOCK AND GLOBAL ARNOLDI METHODS WITH APPLICATIONS TO MODEL REDUCTION

4.1 Introduction

The extended Arnoldi method was first proposed by Druskin and Knizhnerman in [26] for functions of matrices in the symmetric, large and sparse case. The method was then generalized to the nonsymmetric case by Simoncini in [81] and applied for solving large-scale Lyapunov matrix equations [62, 81] with low rank right-hand sides. In [46], the extended block Arnoldi method was used for computing approximate solutions to large scale continuous-time algebraic Riccati equations while in [47] the extended global Arnoldi method was defined and used for solving large Sylvester matrix equations. If $A \in \mathbb{R}^{n\times n}$ is nonsingular, $v \in \mathbb{R}^n$ and m is a fixed integer, the classical extended Arnoldi Krylov subspace $K_m(A, v)$, is the subspace of $\mathbb{R}^n$ spanned by the vectors $A^{-m}\, v, \ldots, A^{-2}\, v, A^{-1}\, v, v, A\, v, A^2\, v, \ldots, A^{m-1}\, v$.
A convergence analysis of the extended Krylov subspace was recently developed in [62] where new general estimates for the convergence rate were obtained with real nonsymmetric and nonsingular matrices A.
For $V \in \mathbb{R}^{n\times r}$, the extended block Krylov subspace $\mathbb{K}_m(A, V)$ is the subspace of $\mathbb{R}^n$ spanned by the columns of the matrices $A^k\, V$, $k = -m, \ldots, m-1$. This

subspace is denoted by

$$\mathbb{K}_m(A, V) = \text{Range}\{A^{-m}\,V, \ldots, A^{-2}\,V, A^{-1}\,V, V, A\,V, A^2\,V, \ldots, A^{m-1}\,V\}.$$

The subspace $\mathbb{K}_m(A, V)$ is the sum of the simple extended Krylov subspaces $K_m(A, V^{(i)})$, $i = 1, \ldots, r$ where $V^{(i)}$ is the i-th column of the matrix V. Notice that $Z \in \mathbb{K}_m(A, V)$ means that

$$Z = \sum_{i=-m}^{m-1} A^i\,V\,\Omega_i, \text{ where } \Omega_i \in \mathbb{R}^{r\times r},\ i = -m, \ldots m-1.$$

On the other hand, the extended matrix or global Krylov subspace $\mathcal{K}_m(A, V) \subset \mathbb{R}^{n\times r}$ is the subspace of matrices in $\mathbb{R}^{n\times r}$ spanned by $A^k\,V$, $k = -m, \ldots, m-1$, i.e.,

$$\mathcal{K}_m(A, V) = span\{A^{-m}\,V, \ldots, A^{-2}\,V, A^{-1}\,V, V, A\,V, A^2\,V, \ldots, A^{m-1}\,V\},$$

and hence $Z \in \mathcal{K}_m(A, V)$ iff $Z = \sum_{i=-m}^{m-1} \alpha_i A^i\,V,\ \alpha_i \in \mathbb{R}$.

In this work, we give some new algebraic properties of the extended block and the extended global Arnoldi algorithms. The new properties use algebraic relations with the matrix A^{-1}. These new relations could be used in moment matching techniques for model reduction in large scale Multiple Input Multiple Output (MIMO) dynamical systems. In particular, we will show that some moments of the original transfer function are matched when using the approximated transfer function. The advantage of the extended block or global methods is the fact that they allow to approximate the original transfer functions in the low frequency range as well as in the hight frequency one.

This chapter is organized as follows. In Section 2, we recall the extended block and global Arnoldi algorithms and give some new algebraic properties. The application of these methods to model order reduction is considered in Section 3. We show how to apply the extended block and global Arnoldi processes to dynamical MIMO systems to obtain low order models such that the Markov parameters and the moments of the original transfer function are approximated by the ones of the projected transfer function. The last section is devoted to some numerical experiments.

Preliminaries and notations We review some notations and definitions that will be used throughout this chapter. For two matrices Y and Z in $\mathbb{R}^{n\times r}$, we define the Frobenius inner product $\langle Y, Z\rangle_F = \text{Tr}(Y^T\,Z)$ where Tr($Y^T\,Z$) denotes the trace of the square matrix $Y^T\,Z$. The associated Frobenius norm is given by $\|Y\|_F =$

$\mathrm{Tr}(Y^T Y)^{\frac{1}{2}}$. A system $\{V_1, V_2, \ldots, V_m\}$ of elements of $\mathbb{R}^{n\times r}$ is said to be $F-$orthonormal if it is orthonormal with respect to the inner product $\langle . \ , . \rangle_F$, i.e., $\langle V_i, V_j \rangle_F = \delta_{i,j}$. For $Y \in \mathbb{R}^{n\times r}$, we denote by vec($Y$) the vector of $\mathbb{R}^{nr}$ 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 assuming that all the sizes are in agreement.

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. $\mathrm{vec}(A)^T \,\mathrm{vec}(B) = \mathrm{Tr}(A^T B)$.

Definition 4.1 *Let $A = [A_1, \ldots, A_q]$ and $B = [B_1, \ldots, B_l]$ be matrices of dimension $n \times qp$ and $n \times lp$, respectively, where A_i and B_j $(i = 1, \ldots, q;\ j = 1, \ldots, l)$ are $n \times p$. Then the $q \times l$ matrix $A^T \diamond B$ is defined by :*

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

Remark 4.1 *The following relations were established in [18].*

1. *The matrix $A = [A_1, \ldots, A_q]$ is $F-$orthonormal if and only if $A^T \diamond A = I_q$.*
2. *For all $X \in \mathbb{R}^{n\times p}$, we have $X^T \diamond X = \|X\|_F^2$.*
3. *$(D\, A)^T \diamond B = A^T \diamond (D^T B)$.*
4. *$A^T \diamond (B\,(L \otimes I_p)) = (A^T \diamond B)\, L$.*
5. *$\|A^T \diamond B\|_F \le \|A\|_F \|B\|_F$.*

In the next proposition, we recall the global QR (gQR) factorisation of an $n \times kr$ matrix Z.

Proposition 4.1 *[18] Let $\mathcal{Z} = \left[\ Z_1,\ \ Z_2,\ \ \ldots,\ \ Z_k\ \right]$ be an $n \times kr$ matrix with $Z_i \in \mathbb{R}^{n\times r}$, $i = 1, \ldots, k$. Then, the matrix $\mathcal{Z}$ can be factored as*

$$\mathcal{Z} = \mathcal{Q}\,(R \otimes I_r),$$

where $\mathcal{Q} = [Q_1, \ldots, Q_k]$ is an $n \times kr$ F-orthonormal matrix satisfying $\mathcal{Q}^T \diamond \mathcal{Q} = I_k$ and R is an upper triangular matrix of dimension $k \times k$.

4.2 Some algebraic properties on the extended block and global Arnoldi processes

4.2.1 The block case

The extended block Arnoldi algorithm generates a sequence of blocks $\{V_1^b, \ldots, V_m^b\}$ of size $n \times 2r$ such that their columns form an orthonormal basis of the extended block Krylov subspace $\mathbb{K}_m(A, V)$. The algorithm is defined as follows [46, 81].

Algorithm 4.1 The extended block Arnoldi algorithm

- Inputs : $A \in \mathbb{R}^{n\times n}$, $V \in \mathbb{R}^{n\times r}$, m.
- Compute $[V_1^b, \Lambda] = QR([V, A^{-1}\,V])$, $\mathbb{V}_1 = [V_1^b]$.
- For $j = 1, \ldots, m$
  1. Set $V_j^{b(1)}$: first r columns of V_j^b; $V_j^{b(2)}$: second r columns of V_j^b.
  2. $\widetilde{V}_{j+1}^b = [A\,V_j^{b(1)}, A^{-1}\,V_j^{b(2)}]$.
  3. Orthogonalize $\widetilde{V}_{j+1}^b$ with respect to to $\mathbb{V}_1^b, \ldots, \mathbb{V}_j^b$ to get V_{j+1}^b, i.e.,
     for $i = 1, 2, \ldots, j$
     $H_{i,j}^b = (V_i)^{bT}\,\widetilde{V}_{j+1}^b$;
     $\widetilde{V}_{j+1}^b = \widetilde{V}_{j+1}^b - V_i^b\,H_{i,j}^b$;
     end for
  4. $[V_{j+1}^b,\ H_{j+1,j}^b] = QR(\widetilde{V}_{j+1}^b)$.
  5. $\mathbb{V}_{j+1} = [\mathbb{V}_j,\ V_{j+1}^b]$.

  End For.

The blocks $\mathbb{V}_m = \left[V_1^b, V_2^b, \ldots, V_m^b\right]$ with $V_i^b \in \mathbb{R}^{n\times 2r}$ have their columns mutually orthogonal provided that none of the upper triangular matrices $H_{j+1,j}^b$ are rank deficient.

Hence, after m steps, Algorithm 4.1 builds an orthonormal basis $\mathbb{V}_m$ of the extended block Krylov subspace $\mathbb{K}_m(A, V)$ and a upper block Hessenberg matrix $\mathbb{H}_m$ whose non zero blocks are the $H_{i,j}^b$. Note that each submatrix $H_{i,j}^b$ $(1 \le i \le j \le m)$ is of order $2r$.

Let $T_{i,j}^b = (V_i^b)^T\,A\,V_j^b \in \mathbb{R}^{2r\times 2r}$ and $\mathbb{T}_m = [T_{i,j}^b] \in \mathbb{R}^{2mr\times 2mr}$ be the restriction of

the matrix A to the extended Krylov subspace $\mathbb{K}_m(A, V)$, i.e.,

$$\mathbb{T}_m = \mathbb{V}_m^T \, A \, \mathbb{V}_m.$$

It is shown in [81] that $\mathbb{T}_m$ is also upper block Hessenberg with $2r \times 2r$ blocks. Moreover, a recursion is derived to compute $\mathbb{T}_m$ from $\mathbb{H}_m$ without requiring matrix-vector products with A. For more details, on how to compute $\mathbb{T}_m$ from $\mathbb{H}_m$, we refer to [81]. We note that for large problems, the inverse of the matrix A is not computed explicitly and in this case we can use iterative solvers with preconditioners to solve linear systems with A. Next, we give the extended block Arnoldi relations

$$\begin{aligned} A \, \mathbb{V}_m &= \mathbb{V}_{m+1} \, \overline{\mathbb{T}}_m, \\ &= \mathbb{V}_m \, \mathbb{T}_m + V_{m+1}^b \, T_{m+1,m}^b \, \mathbb{E}_m^T, \end{aligned}$$

where $\overline{\mathbb{T}}_m = \mathbb{V}_{m+1}^T \, A \, \mathbb{V}_m$, and $\mathbb{E}_m = [O_{2r \times 2(m-1)r}, I_{2r}]^T$ is the matrix of the last $2r$ columns of the $2mr \times 2mr$ identity matrix I_{2mr}. We will also consider the matrix defined as

$$\mathbb{L}_m = \mathbb{V}_m^T \, A^{-1} \, \mathbb{V}_m.$$

Notice that we can check that the matrix $\mathbb{L}_m = [L_{i,j}^b]$ is also an upper block Hessenberg matrix. Moreover, the sub-matrices $L_{i+1,i}^b \in \mathbb{R}^{2r \times 2r}$are such that the r first columns are zero. Hence, $L_{m+1,m}^b$ is partitioned under the form

$$L_{m+1,m}^b = \begin{bmatrix} 0_r & L_{m+1,m}^{b(1,2)} \\ 0_r & L_{m+1,m}^{b(2,2)} \end{bmatrix}. \tag{4.1}$$

In the sequel, we give some new properties that would be useful for building model order reduction for large scale dynamical systems defined by (4.19).

Proposition 4.2 *Assume that m steps of Algorithm 4.1 have been run and let $\overline{\mathbb{L}}_m = \mathbb{V}_{m+1}^T \, A^{-1} \, \mathbb{V}_m$, then we have the following relations*

$$\begin{aligned} A^{-1} \, \mathbb{V}_m &= \mathbb{V}_{m+1} \, \overline{\mathbb{L}}_m \\ &= \mathbb{V}_m \, \mathbb{L}_m + V_{m+1}^b \, L_{m+1,m}^b \, \mathbb{E}_m^T. \end{aligned} \tag{4.2}$$

Proof As $\mathbb{V}_{m+1} = [\mathbb{V}_m, V_{m+1}^b]$, we have

$$\begin{aligned}
\mathbb{L}_{m+1} &= \mathbb{V}_{m+1}^T A^{-1} \mathbb{V}_{m+1} \\
&= \begin{bmatrix} \mathbb{V}_m^T A^{-1} \mathbb{V}_m & \mathbb{V}_m^T A^{-1} V_{m+1}^b \\ V_{m+1}^{bT} A^{-1} \mathbb{V}_m & V_{m+1}^{bT} A^{-1} V_{m+1}^b \end{bmatrix} \\
&= \begin{bmatrix} \mathbb{L}_m & \mathbb{V}_m^T A^{-1} V_{m+1}^b \\ V_{m+1}^{bT} A^{-1} \mathbb{V}_m & V_{m+1}^{bT} A^{-1} V_{m+1}^b \end{bmatrix}.
\end{aligned}$$

Now, since $\mathbb{L}_{m+1}$ is an upper block Hessenberg matrix, we also have

$$(V_{m+1}^b)^T A^{-1} \mathbb{V}_m = L_{m+1,m}^b \, \mathbb{E}_m^T,$$

and so the upper block Hessenberg matrix $\overline{\mathbb{L}}_m$ can be expressed as

$$\overline{\mathbb{L}}_m = \begin{bmatrix} \mathbb{L}_m \\ L_{m+1,m}^b \, \mathbb{E}_m^T \end{bmatrix}.$$

Using the fact that $A^{-1} \mathbb{K}_m(A, V) \subseteq \mathbb{K}_{m+1}(A, V)$ and $\mathbb{V}_{m+1}$ is orthogonal, it follows that there exists an upper block Hessenberg matrix L such that $A^{-1} \mathbb{V}_m = \mathbb{V}_{m+1} L$. Then, $\mathbb{V}_{m+1}^T A^{-1} \mathbb{V}_m = L$, which shows that $L = \overline{\mathbb{L}}_m$. Hence, we obtain

$$A^{-1} \mathbb{V}_m = \mathbb{V}_{m+1} L = \mathbb{V}_{m+1} \overline{\mathbb{L}}_m = \mathbb{V}_m \mathbb{L}_m + V_{m+1}^b L_{m+1,m}^b \, \mathbb{E}_m^T,$$

which completes the proof.
Now using (4.1) and the fact that $V_{m+1}^b = [V_{m+1}^{b(1)}, V_{m+1}^{b(2)}]$, the relation (4.2) becomes

$$A^{-1} \mathbb{V}_m = \mathbb{V}_m \mathbb{L}_m + [O_{n \times (2m-1)r}, V_{m+1}^{b(1)} L_{m+1,m}^{b(1,2)} + V_{m+1}^{b(2)} L_{m+1,m}^{b(2,2)}].$$

Next, to show how to compute the columns of the matrix $\overline{\mathbb{L}}_m$ without using A^{-1}, we have to give some notations :

— Let $[V,\ A^{-1}V] = V_1^b \Lambda$ be the QR decomposition of $[V,\ A^{-1}V]$ which can be written as

$$[V, A^{-1}V] = V_1^b \Lambda = [V_1^{b(1)},\ V_1^{b(2)}] \begin{bmatrix} \Lambda_{1,1} & \Lambda_{1,2} \\ 0 & \Lambda_{2,2} \end{bmatrix}. \tag{4.3}$$

— For $k = 1, \ldots, m$, let's partition the lower triangular matrix $H_{k+1,k}^b$ under the form

$$H_{k+1,k}^b = \begin{bmatrix} H_{k+1,k}^{b(1,1)} & H_{k+1,k}^{b(1,2)} \\ 0 & H_{k+1,k}^{b(2,2)} \end{bmatrix}.$$

The following result enables us to compute $\overline{\mathbb{L}}_m$ directly from the columns of the upper block Hessenberg matrix $\overline{\mathbb{H}}_m$ obtained from Algorithm 4.1.

Proposition 4.3 *Let $\overline{\mathbb{L}}_m$ and $\overline{\mathbb{H}}_m$ be the upper block Hessenberg matrices defined earlier. Then we have the following relations*

$$\overline{\mathbb{L}}_m\, \widetilde{e}_1 = \big[\widetilde{e}_1\, \Lambda_{1,2} + \widetilde{e}_1\, \Lambda_{2,2}\big](\Lambda_{1,1})^{-1}, \tag{4.4}$$

and for $k = 1, \ldots, m$

$$\overline{\mathbb{L}}_m\, \widetilde{e}_{2k} = \overline{\mathbb{H}}_m\, \widetilde{e}_{2k}, \tag{4.5}$$

and

$$\mathbb{L}_m\, \widetilde{e}_{2k+1} = \left(\widetilde{e}_{2k-1} - \begin{bmatrix} \overline{\mathbb{L}}_k \\ 0_{2(m-k)r\times 2kr} \end{bmatrix} \mathbb{H}_k\, \widetilde{e}_{2k-1}\right) (H^{b(1,1)}_{k+1,k})^{-1}, \tag{4.6}$$

where $\widetilde{e}_i = e_i \otimes I_r$ and the e_i's are the vectors of the canonical basis.

Proof To prove (4.4), we start from the QR decomposition of $[V,\ A^{-1}V]$ given in (4.3) :

$$[V,\ A^{-1}V] = [V_1^{b(1)}\, \Lambda_{1,1}, V_1^{b(1)}\, \Lambda_{1,2} + V_1^{b(2)}\, \Lambda_{2,2}].$$

Then, if $\Lambda_{1,1}$ is nonsingular, we obtain

$$A^{-1}\, V_1^{b(1)} = A^{-1}\, V\, \Lambda_{1,1}^{-1} = [V_1^{b(1)}\, \Lambda_{1,2} + V_1^{(2)}\, \Lambda_{2,2}]\, \Lambda_{1,1}^{-1}.$$

Then we get (4.4) by pre-multiplying the above equality on the left by $\mathbb{V}_{m+1}^T$ and using the facts that $\mathbb{V}_{m+1}^T\, V_1^{b(i)} = (e_i \otimes I_r) = \widetilde{e}_i$ for $i = 1,\ 2$ and $\mathbb{V}_{m+1}^T\, A^{-1}\, V_1^{b(1)} = \overline{\mathbb{L}}_m\, (e_1 \otimes I_r) = \overline{\mathbb{L}}_m\, \widetilde{e}_1$.

To prove (4.5) and (4.6), we notice that for $k \geq 1$, $V_k = [V_k^{b(1)}, V_k^{b(2)}] \in \mathbb{R}^{n\times 2r}$ and from Algorithm 4.1, we have

$$\widehat{V}^b_{k+1} = [A\, V_k^{b(1)}, A^{-1}\, V_k^{b(2)}] - \mathbb{V}_k\, \mathbb{H}_k\, [\widetilde{e}_{2k-1}, \widetilde{e}_{2k}], \tag{4.7}$$

and

$$V^b_{k+1}\, H^b_{k+1,k} = \widehat{V}^b_{k+1}. \tag{4.8}$$

Using the relations (4.7) and (4.8), we obtain

$$\begin{aligned} A^{-1}\, V_k^{b(2)} = \widehat{V}^b_{k+1}\, \widetilde{e}_2 + \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k} &= V^b_{k+1}\, H^b_{k+1,k}\, \widetilde{e}_2 + \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k} \\ &= \mathbb{V}_{k+1}\overline{\mathbb{H}}_k\, \widetilde{e}_{2k}. \end{aligned}$$

Now, multiplying on the left by $\mathbb{V}_{m+1}^T$, we get

$$\mathbb{V}_{m+1}^T\, A^{-1}\, V_k^{b(2)} = \mathbb{V}_{m+1}^T\, \mathbb{V}_{k+1}\, \overline{\mathbb{H}}_k\, \widetilde{e}_{2k},$$

hence,

$$\mathbb{V}_{m+1}^T\, A^{-1}\, \mathbb{V}_m\, \widetilde{e}_{2k} = \begin{bmatrix} I_{2(k+1)r} \\ 0_{2(m-k)r\times 2(k+1)r} \end{bmatrix} \overline{\mathbb{H}}_k\, \widetilde{e}_{2k}$$

and so

$$\overline{\mathbb{L}}_m\, \widetilde{e}_{2k} = \begin{bmatrix} \overline{\mathbb{H}}_k \\ 0_{2(m-k)r\times 2kr} \end{bmatrix} = \overline{\mathbb{H}}_m\, \widetilde{e}_{2k},$$

which gives the relation (4.5).

Now, for the even blocks, we multiply (4.7) on the left by A^{-1} and we consider only the first r-columns of each block. We obtain the following relation

$$A^{-1}\, \widehat{V}_{k+1}^{b(1)} = V_k^{b(1)} - A^{-1}\, \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k-1}.$$

Notice that since $\widehat{V}_{k+1}^{b} = V_{k+1}^{b}\, H_{k+1,k}^{b}$, we also have

$$\widehat{V}_{k+1}^{b(1)} = V_{k+1}^{b(1)}\, H_{k+1,k}^{b(1,1)},$$

where $H_{k+1,k}^{b(1,1)}$ is the first $r\times r$ block of the upper $2r\times 2r$ triangular matrix $H_{k+1,k}^{b}$. Then if $H_{k+1,k}^{b(1,1)}$ is nonsingular, we obtain

$$A^{-1}\, V_{k+1}^{b(1)} = A^{-1}\, \widehat{V}_{k+1}^{b(1)}\, (H_{k+1,k}^{b(1,1)})^{-1} = \big(V_k^{b(1)} - A^{-1}\, \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k-1}\big)(H_{k+1,k}^{b(1,1)})^{-1}.$$

Multiplying from the left by $\mathbb{V}_{m+1}^T$, we get

$$\mathbb{V}_{m+1}^T\, A^{-1}\, V_{k+1}^{b(1)} = \Big(\mathbb{V}_{m+1}^T\, V_k^{b(1)} - \mathbb{V}_{m+1}^T\, A^{-1}\, \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k-1}\Big)\, (H_{k+1,k}^{b(1,1)})^{-1},$$

and then

$$\begin{aligned}
\mathbb{L}_{m+1}\, \widetilde{e}_{2k+1} &= \left(\mathbb{V}_{m+1}^T\, \mathbb{V}_{m+1}\, \widetilde{e}_{2k-1} - \mathbb{V}_{m+1}^T\, A^{-1}\, \mathbb{V}_m \begin{bmatrix} I_{2kr} \\ 0_{2(m-k)r\times 2kr} \end{bmatrix} \mathbb{H}_k\, \widetilde{e}_{2k-1}\right) \\
&\times \quad (H_{k+1,k}^{b(1,1)})^{-1} \\
&= \left(\widetilde{e}_{2k-1} - \overline{\mathbb{L}}_m \begin{bmatrix} I_{2kr} \\ 0_{2(m-k)r\times 2kr} \end{bmatrix} \mathbb{H}_k\, \widetilde{e}_{2k-1}\right)\, (H_{k+1,k}^{b(1,1)})^{-1} \\
&= \left(\widetilde{e}_{2k-1} - \begin{bmatrix} \overline{\mathbb{L}}_k \\ 0_{2(m-k)r\times 2kr} \end{bmatrix} \mathbb{H}_k\, \widetilde{e}_{2k-1}\right)\, (H_{k+1,k}^{b(1,1)})^{-1},
\end{aligned}$$

which gives the second relation (4.6).

4.2.2 The global case

The extended global Arnoldi process was first described in [47]. The algorithm is summarized as follows

Algorithm 4.2 The extended global Arnoldi algorithm

- Inputs : $A \in \mathbb{R}^{n\times n}$, $V \in \mathbb{R}^{n\times r}$, m.
- Compute $[V_1^g, \Omega] = gQR([V, A^{-1}V])$, (global QR decomposition) $\mathcal{V}_1 = [V_1^g]$.
- For $j = 1, \ldots, m$
  1. Set $V_j^{g(1)}$: first r columns of V_j^g ; $V_j^{g(2)}$: second r columns of V_j^g.
  2. $\widetilde{V}_{j+1}^g = [A\, V_j^{g(1)}, A^{-1}\, V_j^{g(2)}]$.
  3. F-Orthogonalize $\widetilde{V}_{j+1}^g$ with respect to $\mathcal{V}_1^g, \ldots, \mathcal{V}_j^g$ to get V_{j+1}^g, i.e.,
     for $i = 1, 2, \ldots, j$
     $H_{i,j}^g = (V_i)^{gT} \diamond \widetilde{V}_{j+1}^g$;
     $\widetilde{V}_{j+1}^g = \widetilde{V}_{j+1}^g - V_i^g\, (H_{i,j}^g \otimes I_r)$;
     end for
  4. $[V_{j+1}^g,\ H_{j+1,j}^g] = gQR(\widetilde{V}_{j+1}^g)$.
  5. $\mathcal{V}_{j+1} = [\mathcal{V}_j,\ V_{j+1}^g]$.

  End For.

We point out that if the upper 2×2 triangular matrices $H_{j+1,j}^g$ $(j = 1, \ldots, m)$ are full rank, Algorithm 4.2 constructs an $n \times 2mr$ F-orthonormal matrix $\mathcal{V}_m = [V_1^g, \ldots, V_m^g]$ with $V_i^g \in \mathbb{R}^{n\times 2r}$ $(i = 1, \ldots, m)$ and a $2(m+1) \times 2m$ upper block Hessenberg matrix $\overline{\mathcal{H}}_m = \left[H_{i,j}^g\right] = [h_{p,q}]$ with $H_{i,j}^g \in \mathbb{R}^{2\times 2}$ for $i = 1, \ldots, m+1$, $j = 1, \ldots, m$ and $h_{p,q} \in \mathbb{R}$ for $p = 1, \ldots, 2(m+1)$, $q = 1, \ldots, 2m$.

Now, setting $T_{i,j}^g = V_i^{gT} \diamond (A\, V_j^g) \in \mathbb{R}^{2\times 2}$, for $i, j = 1, \ldots, m$ and introducing the matrices

$$\mathcal{T}_m = \mathcal{V}_m^T \diamond (A\, \mathcal{V}_m) = \left[T_{i,j}^g\right] \quad \text{and} \quad \overline{\mathcal{T}}_m = \mathcal{V}_{m+1}^T \diamond (A\, \mathcal{V}_m),$$

a recursive relation was given in [47] allowing the computation of $\bar{\mathcal{T}}_m$ without requiring additional matrix-vector products with A. Moreover, it was also shown that

after m steps of Algorithm 4.2, we have

$$\begin{aligned} A\,\mathcal{V}_m &= \mathcal{V}_{m+1}\,(\overline{\mathcal{T}}_m \otimes I_r) \\ &= \mathcal{V}_m\,(\mathcal{T}_m \otimes I_r) + V_{m+1}^g\,(T_{m+1,m}^g\,\mathcal{E}_m^T \otimes I_r), \end{aligned}$$

where $\mathcal{E}_m^T = [O_{2\times 2(m-1)}, I_2]$ is the matrix of the last 2 rows of the $2m \times 2m$ identity matrix I_{2m}.

Now, as for the block case seen in the previous subsection, we consider the matrix $\mathcal{L}_m = \left[L_{i,j}^g\right] = [l_{p,q}]$ defined by

$$\mathcal{L}_m = \mathcal{V}_m^T \diamond (A^{-1}\,\mathcal{V}_m),$$

where $L_{i,j}^g \in \mathbb{R}^{2\times 2}$ for $i,j = 1,\ldots,m$ and $l_{p,q} \in \mathbb{R}$ for $p,q = 1,\ldots,2m$. We mention that it can be easily verified that the matrix $\mathcal{L}_m$ is a $2m \times 2m$ upper block Hessenberg matrix and that the sub-matrices $L_{i+1,i}^g \in \mathbb{R}^{2\times 2}$ are such that the first column is zero. So, the sub-matrix $L_{m+1,m}^g$ is such that $l_{2m+1,2m-2} = l_{2m+2,2m-2} = 0$, i.e.,

$$L_{m+1,m}^g = \begin{bmatrix} 0 & l_{2m+1,2m} \\ 0 & l_{2m+2,2m} \end{bmatrix}. \tag{4.9}$$

In order to build order model reduction for large scale dynamical systems, we give some new properties of the extended global Arnoldi process.

Proposition 4.4 *Assume that m steps of Algorithm 4.2 have been run and let $\overline{\mathcal{L}}_m = \mathcal{V}_{m+1}^T\,A^{-1}\,\mathcal{V}_m$, then we have the following relations*

$$\begin{aligned} A^{-1}\,\mathcal{V}_m &= \mathcal{V}_{m+1}\,(\overline{\mathcal{L}}_m \otimes I_r) \\ &= \mathcal{V}_m\,(\mathcal{L}_m \otimes I_r) + V_{m+1}^g\,(L_{m+1,m}^g\,\mathcal{E}_m^T \otimes I_r). \end{aligned} \tag{4.10}$$

The proof is similar to the one given for Proposition 4.2.

Notice that since $V_{m+1}^g = [V_{m+1}^{g(1)}, V_{m+1}^{g(2)}]$, then by using (4.9), the Arnoldi relation (4.10) becomes

$$A^{-1}\,\mathcal{V}_m = \mathcal{V}_m\,\mathcal{L}_m + [O_{n\times(2m-1)},\; l_{2m+1,2m}\,V_{m+1}^{g(1)} + l_{2m+2,2m}\,V_{m+1}^{g(2)}].$$

Now, in order to update progressively the columns of the matrix $\overline{\mathcal{L}}_m$ without inverting A or solving linear systems with A, we recall some elementary results :

— Let $[V,\ A^{-1}V] = V_1^g\,(\Gamma \otimes I_r)$ be the global QR decomposition of $[V,\ A^{-1}V]$ which can be written as

$$[V, A^{-1}V] = V_1^g\,(\Gamma \otimes I_r) = [V_1^{g(1)},\ V_1^{g(2)}] \begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} \\ 0 & \gamma_{2,2} \end{bmatrix}.$$

— For $k = 1, \ldots, m$, let us partition the lower triangular matrix $H_{k+1,k}^g$ under the form

$$H_{k+1,k}^g = \begin{bmatrix} h_{2k+1,2k-1} & h_{2k+1,2k} \\ 0 & h_{2k+2,2k} \end{bmatrix}.$$

As in the block case, the following result enables us to compute $\overline{\mathbb{L}}_m$ directly from the columns of the upper block Hessenberg matrix $\overline{\mathbb{H}}_m$ obtained from Algorithm 4.1.

Proposition 4.5 *Let $\overline{\mathcal{L}}_m = [l_{:,1}, \ldots, l_{:,2m}]$ and $\overline{\mathcal{H}}_m = [h_{:,1}, \ldots, h_{:,2m}]$ be the upper block Hessenberg matrices defined earlier. Then we have the following relations*

$$l_{:,1} = (\gamma_{1,2}\, e_1 + \gamma_{2,2}\, e_2)/\gamma_{1,1},$$

and for $k = 1, \ldots, m$, we have

$$\bar{l}_{:,2k} = h_{:,2k} \tag{4.11}$$

and

$$l_{2k+1} = \left(e_{2k-1} - \begin{bmatrix} \overline{\mathbb{L}}_k \\ 0_{2(m-k)\times 2k} \end{bmatrix} \mathbb{H}_k\, e_{2k-1} \right) / h_{2k+1,2k},$$

where the e_i's are the vectors of the canonical basis.

The proof can be obtained in a similar way as the one for Proposition 4.3 in the block case.

The results of the previous two subsections are used to prove other properties in the next section which is devoted to the application of the extended block and global Arnoldi methods to obtain reduced order models in large scale dynamical systems. As we will see, the methods allow one to approximate low and high frequencies of the corresponding transfer function at the same time.

Next, we give some properties that are used to show that the first m moments of the transfer function F are matched.

Proposition 4.6 *Let* $\mathbb{V}_m = [V_1^b, \ldots, V_m^b]$, $\mathbb{L}_m := \mathbb{V}_m^T A^{-1} \mathbb{V}_m$ *be respectively the matrix and the upper block Hessenberg matrix defined in the extended block Arnoldi process and let* $\mathbb{E}_1 = [I_{2r}, 0_{2r}, \ldots, 0_{2r}]^T$. *Then for* $j = 0 \ldots, m-1$, *we have the following relation*

$$A^{-j} \mathbb{V}_m \mathbb{E}_1 = \mathbb{V}_m \mathbb{L}_m^j \mathbb{E}_1, \tag{4.12}$$

and

$$\mathbb{T}_m^{-1} \mathbb{E}_j = \mathbb{L}_m \mathbb{E}_j, \ j = 1, \ldots, m-1. \tag{4.13}$$

Proof The relation (4.12) can be derived directly by multiplying relation (4.2) from the left by A^{-j+1} and from the right by $\mathbb{E}_1$. Then we obtain

$$A^{-j} \mathbb{V}_m \mathbb{E}_1 = \mathbb{V}_m \mathbb{L}_m^j \mathbb{E}_1 + \sum_{i=1}^{j} A^{-(i-1)} V_{m+1}^b L_{m+1,m}^b \mathbb{E}_m^T \mathbb{L}_m^{j-i} \mathbb{E}_1.$$

Now, as $\mathbb{L}_m$ is an upper block Hessenberg matrix, it follows that $\mathbb{E}_m^T \mathbb{L}_m^{j-i} \mathbb{E}_1 = 0$, for $j = 1, \ldots, m-1$.

To prove the relation (4.13), we multiply (4.2) from the right by $\mathbb{E}_j$, and then we get

$$A^{-1} \mathbb{V}_m \mathbb{E}_j = \mathbb{V}_m \mathbb{L}_m \mathbb{E}_j, \quad \text{for} \quad j = 1, \ldots, m-1.$$

Since, $\mathbb{V}_m$ is orthogonal, pre-multiplying the above equality by $\mathbb{V}_m^T A$, we get $\mathbb{E}_j = \mathbb{T}_m \mathbb{L}_m \mathbb{E}_j$ for $j = 1, \ldots, m-1$ and finally we obtain (4.13), if we assume that $\mathbb{T}_m$ is nonsingular.

Next, we establish a similar result for the extended global Arnoldi process.

Proposition 4.7 *Let* $\mathcal{V}_m = [V_1^g, \ldots, V_m^g]$, $\mathcal{L}_m := \mathcal{V}_m^T \diamond (A^{-1} \mathcal{V}_m)$ *be respectively the matrix and the upper block Hessenberg matrix defined by the extended global Arnoldi process and let* $\mathcal{E}_1 = [I_2, 0_2, \ldots, 0_2]^T$. *Then for* $j = 1 \ldots, m-1$, *we have the following relation*

$$A^{-j} \mathcal{V}_m \mathbb{E}_1 = A^{-j} \mathcal{V}_m (\mathcal{E}_1 \otimes I_r) = \mathcal{V}_m (\mathcal{L}_m^j \mathcal{E}_1 \otimes I_r), \tag{4.14}$$

and for $j = 1, \ldots, m-1$ *we have*

$$\mathcal{T}_m^{-1} \mathcal{E}_j = \mathcal{L}_m \mathcal{E}_j. \tag{4.15}$$

Proof Pre-multiplying (4.10) by A^{-j+1} and using the properties of the Kronecker product we get

$$A^{-j} \mathcal{V}_m = \mathcal{V}_m (\mathcal{L}_m^j \otimes I_r) + \sum_{i=1}^{j} A^{-(i-1)} V_{m+1}^g (L_{m+1,m}^g \mathcal{E}_m^T \mathcal{L}_m^{j-i} \otimes I_r).$$

Post-multiplying the above equality by $(\mathcal{E}_1 \otimes I_r)$, we obtain

$$\begin{aligned} A^{-j}\,\mathcal{V}_m\,\mathbb{E}_1 &= A^{-j}\,\mathcal{V}_m\,(\mathcal{E}_1 \otimes I_r) \\ &= \mathcal{V}_m\,(\mathcal{L}_m^j\ \mathcal{E}_1 \otimes I_r) + \sum_{i=1}^{j} A^{-(i-1)}\,V_{m+1}^g\,(L_{m+1,m}^g\,\mathcal{E}_m^T\,\mathcal{L}_m^{j-i}\,\mathcal{E}_1 \otimes I_r), \\ &= \mathcal{V}_m\,(\mathcal{L}_m^j\ \mathcal{E}_1 \otimes I_r). \end{aligned}$$

In the last equality, we used the fact that $\mathcal{E}_m^T\,\mathcal{L}_m^{j-i}\,\mathcal{E}_1 = 0$ since $\mathcal{L}_m$ is an upper block Hessenberg matrix. Using again the properties of the $\otimes$ and $\diamond$ products, the proof of the second relation (4.15) can be derived in a similar fashion to that of (4.13).

Next, we give a general result that is satisfied by upper Hessenberg matrices. This result will be used when establishing moment matching properties for the block and global Arnoldi processes.

Proposition 4.8 *Let $T = (T_{i,j})$ and $L = (L_{i,j})$ be two upper block Hessenberg matrices with blocks $T_{i,j},\ L_{i,j} \in \mathbb{R}^{r\times r}$ for $i, j = 1, \ldots, m$ and suppose that*

$$T\,\mathbb{E}_j = L\,\mathbb{E}_j, \quad \textit{for} \quad j = 1, \ldots, m-1 \tag{4.16}$$

where $\mathbb{E}_j = [0_r, \ldots, 0_r, I_r, 0_r, \ldots, 0_r]^T$, is the $mr \times r$ matrix whose columns are the column $j, \ldots, j+r$ of the identity matrix I_{mr}. Then

$$T^k\,\mathbb{E}_1 = L^k\,\mathbb{E}_1, \textit{ for} \quad k = 1, \ldots, m-1. \tag{4.17}$$

Proof For $k = 1, \ldots, m-1$, we denote by $T_{i,j}^{(k)}$ and $L_{i,j}^{(k)}$ the (i,j)-th block of T^k, L^k respectively, i.e.,

$$T^k = \left(T_{i,j}^{(k)}\right) \quad \text{and} \quad L^k = \left(L_{i,j}^{(k)}\right).$$

Since T and L are upper block Hessenberg matrices, we can easily verify that

$$T_{i,j}^{(k)} = L_{i,j}^{(k)} = 0_r, \text{ for } \quad i > j + k. \tag{4.18}$$

Now, we proceed by induction on $k \in \{1, 2, \ldots, m-1\}$.
We first remark that the property is verified for $k = 1$ and we suppose that the property is valid for $k \in \{1, \ldots, m-2\}$.

Obviously, we have

$$T^{k+1}\,\mathbb{E}_1 = T\,\left(T^k\,\mathbb{E}_1\right) = T(\,L^k\,\mathbb{E}_1\,) = \begin{pmatrix} \sum_{p=1}^{m} T_{1,p}^{(1)}\,L_{p,1}^{(k)} \\ \sum_{p=1}^{m} T_{2,p}^{(1)}\,L_{p,1}^{(k)} \\ \vdots \\ \sum_{p=1}^{m} T_{m,p}^{(1)}\,L_{p,1}^{(k)} \end{pmatrix},$$

and thanks to (4.18) and (4.16), we have $L_{p,1}^{(k)} = 0_r$ for $p = m$ and $T_{i,p}^{(1)} = L_{i,p}^{(1)}$ for $i = 1, \ldots, m$ and $p = 1, \ldots, m-1$. So,

$$T^{k+1}\,\mathbb{E}_1 = \begin{pmatrix} \sum_{p=1}^{m-1} T_{1,p}^{(1)}\,L_{p,1}^{(k)} \\ \sum_{p=1}^{m-1} T_{2,p}^{(1)}\,L_{p,1}^{(k)} \\ \vdots \\ \sum_{p=1}^{m-1} T_{m,p}^{(1)}\,L_{p,1}^{(k)} \end{pmatrix} = \begin{pmatrix} \sum_{p=1}^{m-1} L_{1,p}^{(1)}\,L_{p,1}^{(k)} \\ \sum_{p=1}^{m-1} L_{2,p}^{(1)}\,L_{p,1}^{(k)} \\ \vdots \\ \sum_{p=1}^{m-1} L_{m,p}^{(1)}\,L_{p,1}^{(k)} \end{pmatrix},$$

and since $L_{p,1}^{(k)} = 0_r$ for $p = m$, we finally get

$$T^{k+1}\,\mathbb{E}_1 = \begin{pmatrix} \sum_{p=1}^{m} L_{1,p}^{(1)}\,L_{p,1}^{(k)} \\ \sum_{p=1}^{m} L_{2,p}^{(1)}\,L_{p,1}^{(k)} \\ \vdots \\ \sum_{p=1}^{m} L_{m,p}^{(1)}\,L_{p,1}^{(k)} \end{pmatrix} = L^{k+1}\,\mathbb{E}_1.$$

4.3 Application for model reduction techniques

We consider the following Linear Time Independent (LTI) dynamical system

$$\begin{cases} \dot{x}(t) &= A\,x(t) + B\,u(t), \\ y(t) &= C\,x(t), \end{cases} \tag{4.19}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t),\ y(t) \in \mathbb{R}^r$ are the input and the output vectors of the system (4.19), respectively. The matrices $B,\ C^T$ are in $\mathbb{R}^{n\times r}$ and

$A \in \mathbb{R}^{n\times n}$ is assumed to be large and sparse. The transfer function of the original system (4.19) is given as

$$F(s) = C\,(s\,I_n - A)^{-1}\,B. \tag{4.20}$$

In many applications, the dimension n of the system (4.19) is large which makes the computations infeasible in terms of execution time and memory. Then the goal of model reduction problems is to produce a low-order system of the form

$$\left\{\begin{array}{rcl} \dot{x}_m(t) &=& A_m\,x_m(t) + B_m\,u(t) \\ y_m(t) &=& C_m\,x_m(t), \end{array}\right. \tag{4.21}$$

where $A_m \in \mathbb{R}^{p\times p}$, $B_m,\ C_m^T \in \mathbb{R}^{p\times r}$. The basic technique is to project the system's state space of dimension n onto a space of lower dimension $p \ll n$, in such a way that the reduced-order model preserves the important properties of the original system like stability and passivity and such that the output y_m is close to the output y of the original system. The associated low-order transfer function is denoted by

$$F_m(s) = C_m\,(s\,I_p - A_m)^{-1}\,B_m.$$

There are two well known sets of model reduction methods for MIMO systems which are currently in use, SVD based methods and Krylov (moment matching) based methods; see [35, 40] and the references therein. One of the most common approach of the first category is the so-called balanced reduced order model which was introduced by Moore [70]. Krylov subspace methods have been extensively used for SISO (the case $r = 1$) and MIMO dynamical systems; see [19, 26–28, 34, 41, 48, 82] and the references therein. Unfortunately the standard version of these methods builds reduced order models that poorly approximate low and high frequency dynamics at the same time. In order to address this problem, we consider the extended Arnoldi process associated to the matrices A and A^{-1}. The transfer function F relates the Laplace transform of the output vector to that of the input vector. For that reason, it is called the transfer function matrix of the system. Each entry $F_{i,j}(s)$ 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.

The rational function F can be expressed as a sum of a Taylor series around $(s = \infty)$ in the following form

$$F(s) = \frac{1}{s}\,C\,(I_n - \frac{A}{s})^{-1}\,B = \frac{1}{s}\sum_{i=0}^{\infty} M_i\,s^{-i},\ \text{with}\ M_i = C\,A^i\,B.$$

Recall that the matrix coefficients M_i are called the Markov parameters of F. Now applying the extended block Arnoldi process to the pair (A, B), we can verify that the original transfer function F can be approximated by

$$\mathbb{F}_m(s) = \mathbb{C}_m \, (s \, I_{2mr} - \mathbb{T}_m)^{-1} \, \mathbb{B}_m,$$

where $\mathbb{T}_m = \mathbb{V}_m^T \, A \, \mathbb{V}_m$, $\mathbb{C}_m = C \, \mathbb{V}_m$ and $\mathbb{B}_m = \mathbb{V}_m^T \, B$. This reduced transfer function is related to the low-order dynamical system (4.21) with $A_m = \mathbb{T}_m$.
Similarly, if m iterations of the extended global Arnoldi algorithm are applied to the pair (A, B), then we can approximate F by

$$\mathcal{F}_m(s) = \mathcal{C}_m \, (s \, I_{2mr} - (\mathcal{T}_m \otimes I_r))^{-1} \, \mathcal{B}_m,$$

where $\mathcal{T}_m = \mathcal{V}_m^T \diamond (A \, \mathcal{V}_m)$, $\mathcal{C}_m = C \, \mathcal{V}_m$ and $\mathcal{B}_m = \mathcal{V}_m^T \diamond B = \|B\|_F \, (e_1^{(2m)} \otimes I_r)$. In this case, the reduced transfer function is related to the low-order dynamical system (4.21) with $A_m = \mathcal{T}_m \otimes I_r$.
The developments of $\mathbb{F}_m$ and $\mathcal{F}_m$ around $s = \infty$ give the following expressions

$$\mathbb{F}_m(s) = \frac{1}{s} \sum_{i=0}^{\infty} m_i^b \, s^{-i}, \quad \text{with } m_i^b = \mathbb{C}_m \, \mathbb{T}_m^i \, \mathbb{B}_m,$$

and

$$\mathcal{F}_m(s) = \frac{1}{s} \sum_{i=0}^{\infty} m_i^g \, s^{-i}, \quad \text{with } m_i^g = \mathcal{C}_m \, (\mathcal{T}_m \otimes I_r)^i \, \mathcal{B}_m.$$

In this case, one can show that the first m Markov parameters are matched, i.e. in the block case

$$M_i = m_i^b, \quad i = 0, \ldots, m-1,$$

and in the global case

$$M_i = m_i^g, \quad i = 0, \ldots, m-1.$$

Now, the development of the Neumann series of F around $s = 0$ gives the following expression

$$F(s) = \sum_{i=0}^{\infty} \widetilde{M}_{i+1} \, s^i.$$

The matrix coefficients $\widetilde{M}_i$ are called the moments of F and they are given by

$$\widetilde{M}_j = -C \, A^{-j} \, B, \; j = 1, 2, \ldots.$$

By considering the Taylor series of $\mathbb{F}_m$ and $\mathcal{F}_m$, we get the following expansion of $\mathbb{F}_m$ around $s = 0$

$$\mathbb{F}_m(s) = \sum_{i=0}^{\infty} \widetilde{m}_{i+1}^b s^i, \text{ with } \widetilde{m}_i^b = -\mathbb{C}_m \, \mathbb{T}_m^{-i} \, \mathbb{B}_m,$$

while for $\mathcal{F}_m$, we get

$$\mathcal{F}_m(s) = \sum_{i=0}^{\infty} \widetilde{m}_{i+1}^g s^i, \text{ with } \widetilde{m}_i^b = -\mathcal{C}_m \, (\mathcal{T}_m \otimes I_r)^{-i} \, \mathcal{B}_m.$$

As for the Markov parameters, the following result shows that the first m moments resulting from the Newman series of the transfer function F around $s = 0$ are also matched either by those of $\mathbb{F}_m$ when using the extended block Arnoldi process or by those of $\mathcal{F}_m$ when using the extended global Arnoldi process.

Proposition 4.9 *Let $\widetilde{M}_j$ and $\widetilde{m}_j^b$ be the matrix moments given by the Newman expansions of F and $\mathbb{F}_m$, respectively around $s = 0$. Then we have*

$$\widetilde{M}_j = \widetilde{m}_j^b, \quad \textit{for } \; j = 0, \ldots, m-1.$$

Proof The equality is verified for $j = 0$. For $j \geq 1$, we obtain

$$\begin{aligned} \widetilde{M}_j &= C \, A^{-j} \, B \\ &= C \, A^{-j} \, V_1^b \begin{bmatrix} \Lambda_{1,1} \\ 0 \end{bmatrix} = C \, A^{-j} \, \mathbb{V}_m \, \mathbb{E}_1 \begin{bmatrix} \Lambda_{1,1} \\ 0 \end{bmatrix}. \end{aligned}$$

Therefore, using the result of Proposition 4.6, we get

$$\widetilde{M}_j = C \, \mathbb{V}_m \, \mathbb{L}_m^j \, \mathbb{E}_1 \begin{bmatrix} \Lambda_{1,1} \\ 0 \end{bmatrix}.$$

On the other hand, applying Proposition 4.8 to the upper Hessenberg matrices $\mathbb{L}_m$ and $\mathbb{T}_m^{-1}$, we get

$$\mathbb{L}_m^j \, \mathbb{E}_1 = \mathbb{T}_m^{-j} \, \mathbb{E}_1; \quad j = 0, \ldots, m-1$$

and this gives for $j = 1, \ldots, m-1$

$$\begin{aligned} \widetilde{M}_j &= C \, \mathbb{V}_m \, \mathbb{T}_m^{-j} \, \mathbb{V}_m^T \, V_1^b \begin{bmatrix} \Lambda_{1,1} \\ 0 \end{bmatrix} \\ &= C \, \mathbb{V}_m \, \mathbb{T}_m^{-j} \, \mathbb{V}_m^T \, B = \mathbb{C}_m \, \mathbb{T}_m^{-j} \, \mathbb{B}_m \\ &= \widetilde{m}_j^b. \end{aligned}$$

Now, using the extended global Arnoldi process, we can also state the following result

Proposition 4.10 *Let $\widetilde{M}_j$ and $\widetilde{m}_j^g$ be the matrix moments given by the Newman expansions of F and $\mathcal{F}_m$, respectively around $s = 0$. Then we have*

$$\widetilde{M}_j = \widetilde{m}_j^g, \quad \textit{for } \; j = 0, \ldots, m-1.$$

Proof The equality is verified for $j = 0$. For $j \geq 1$, we obtain

$$\begin{aligned} \widetilde{M}_j &= C\, A^{-j}\, B \\ &= C\, A^{-j}\, V_1^g \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix} = C\, A^{-j}\, \mathcal{V}_m\, \mathbb{E}_1 \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix}. \end{aligned}$$

Therefore, using the result of Proposition 4.7, we get

$$\widetilde{M}_j = C\, \mathcal{V}_m\, (\mathcal{L}_m^j\, \mathcal{E}_1 \otimes I_r) \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix}.$$

Now, similarly to the block case, applying proposition 4.8 to $\mathcal{L}_m$ and $\mathcal{T}_m^{-1}$, we also have $\mathcal{L}_m^j\, \mathcal{E}_1 = \mathcal{T}_m^{-j}\, \mathcal{E}_1$ for $j = 0, \ldots, m-1$ and so we get

$$\begin{aligned} \widetilde{M}_j = C\, \mathcal{V}_m\, (\mathcal{T}_m^{-j}\, \mathcal{E}_1 \otimes I_r) \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix} &= C\, \mathcal{V}_m\, (\mathcal{T}_m^{-j} \otimes I_r)\, (\mathcal{E}_1 \otimes I_r) \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix} \\ &= C\, \mathcal{V}_m\, (\mathcal{T}_m^{-j} \otimes I_r)\, (\mathcal{V}_m^T \diamond V_1^g) \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix} \\ &= C\, \mathcal{V}_m (\mathcal{T}_m^{-j} \otimes I_r) \left(\mathcal{V}_m^T \diamond \left(V_1^g \begin{bmatrix} \gamma_{1,1} \\ 0 \end{bmatrix} \otimes I_r \right) \right) \\ &= C\, \mathcal{V}_m\, (\mathcal{T}_m^{-j} \otimes I_r)\, (\mathcal{V}_m^T \diamond B) \\ &= \mathcal{C}_m\, \mathcal{T}_m^{-j}\, \mathcal{B}_m = \widetilde{m}_j^g. \end{aligned}$$

We would like to mention here that these moment matching results don't influence the extended Arnoldi algorithms themselves but just to clarify for example why the extended block and global Arnoldi algorithms allow us to match some moments and Markov parameters of transfer functions. This will be shown with some numerical experiments in the next section.

4.4 Numerical tests

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 R2010a. We used different known benchmark models listed in Table 4.1.

TABLE 4.1 – Test matrices

Matrix A	size n	$\|A\|_F$	$cond(A)$
FOM	1006	1.82e+04	1000
RAIL5177	5177	5.64e+03	3.74e+07
CDplayer	120	2.31e+05	1.81e+04
Eady	598	1.26e+02	5.37e+02
MNA3	4863	2.11e+05	1.81e+08
Flow	9669	2.54e+04	1.61e+07
FDM	160000	2.87e+08	9.79e+04

The matrices for the benchmark problems CDplayer, FOM, Eady, MNA3 were obtained from NICONET [68] while the matrices for the Flow and RAIL5177 models are from the Oberwolfach collection [1]. Some informations on these matrices are reported in Table 4.1. For the FDM model, the corresponding matrix A is obtained from the 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

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

and the matrices B and C of sizes $n \times r$ and $r \times n$, respectively, where random matrices with entries uniformly distributed in $[0,1]$. The number of inner grid points in each direction was $n_0 = 400$ and the dimension of A is $n = n_0^2$.

1. Oberwolfach model reduction benchmark collection, 2003. http ://www.imtek.de/simulation/benchmark

We notice that in all the figures of Example 1, the parameter m denotes the maximal iteration number for extended block Arnoldi, extended global Arnoldi and for IRKA algorithms. When using balanced truncation, the number m denotes also the maximal iteration number for convergence of the extended block Arnoldi algorithm when applied for solving the coupled Lyapunov equations. We also notice that for the results presented in our plots, the dimension of the reduced models are $2mr$ for extended block and global Arnoldi methods and also for balanced truncation while for IRKA, the reduced models are of dimension mr. For Example 2, the sizes of the obtained reduced-order models are given in Table 4.2.

Example 1. In the first experiment, we considered the models `CDplayer` and `FOM`. Although the matrices of these models have small sizes they are usually considered as benchmark examples. The plots of Figure 4.1 show the norms of the errors $\|F(i\omega)-F_m(i\omega)\|_2$ for the extended block (dashed), extended global (solid) and the balanced-truncation (dashed-dotted) methods with $\omega \in [10^{-5}, 10^5]$. We denote here that the balanced truncation method needs the solution of two low-rank right hand sides Lyapunov matrix equations that we solved by using the extended block Arnoldi method [81] and we stopped the iterations when the norm of the residual was less than 10^{-8}. Figure 4.1 shows that the three methods return similar results with an advantage, in the right plots of this figure, for balanced truncation for medium frequencies. However, balanced-trucation is generally more expensive as compared to the two other methods.

For the second experiment, we considered the models `RAIL5177` and `MNA3` given in Table 4.1. In Figure 4.2, we plotted the norms of the errors $\|F(i\omega) - F_m(i\omega)\|_2$ for the extended block (dashed), extended global (solid) and the balanced-truncation (dashed-dotted) methods with $\omega \in [10^{-6}, 10^6]$.

As can be seen from Figure 4.2, the three methods work well for small and high frequencies with a little advantage for the extended block and global Arnoldi methods for high freqencies.

The plots in Figure 4.3, represent the norms of the errors $\|F(i\omega) - F_m(i\omega)\|_2$ corresponding to the extended block and global Arnoldi methods and to the balanced-truncation method for the models : the `Eady` model with $m = 10$ and $r = 3$ and the `Flow` model with $m = 15$ and $r = 3$ for the frequencies $\omega \in [10^{-5}, 10^5]$.

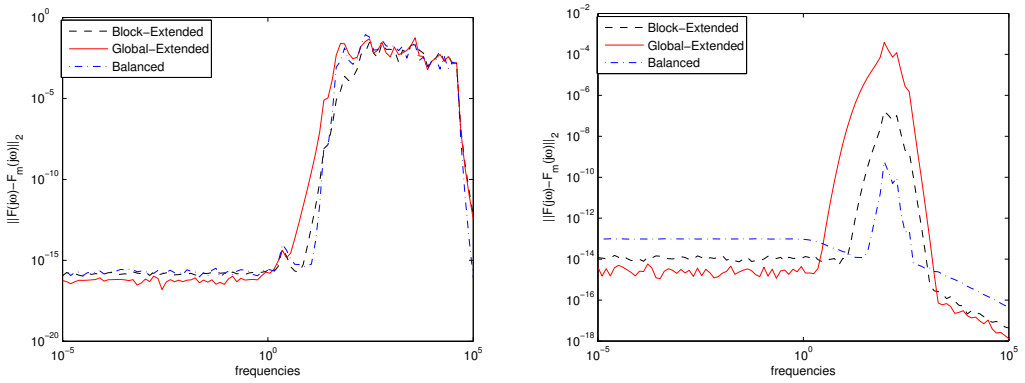

FIGURE 4.1 – The norms of the errors $\|F(i\omega) - F_m(i\omega)\|_2$ for the extended block (dashed), extended global (solid) and the balanced-truncation (dashed-dotted) methods with $\omega \in [10^{-5}, 10^5]$. Left the `CDplayer` model with $m = 10$ and $r = 2$. Right : the `FOM` model with $m = 15$, $r = 3$.

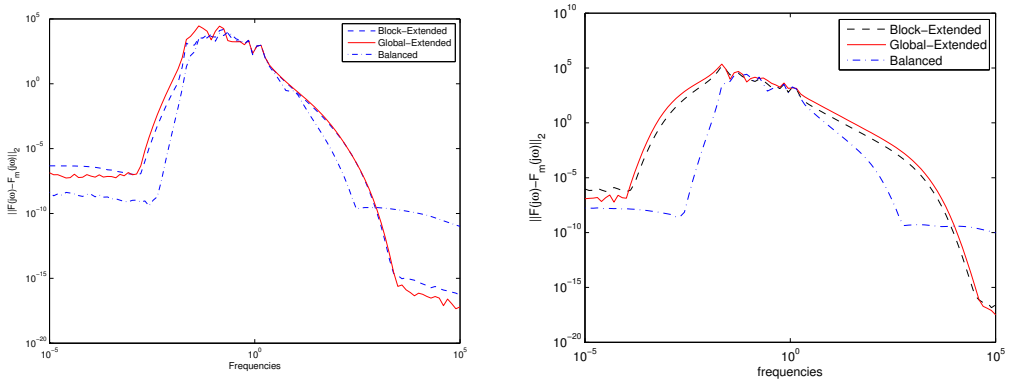

FIGURE 4.2 – The norm of the errors $\|F(i\omega) - F_m(i\omega)\|_2$ for the extended block (dashed), extended global (solid) and the balanced-truncation (dashed-dotted) methods. Left : the `RAIL5177` model with $m = 40$ and $r = 2$. Right : the `MNA3` model with $m = 12$ and $r = 3$.

In the last experiment of Example 1, we compared the extended block and global Arnoldi methods with the well known IRKA method [41] using the `CDplayer` model. In the left side of Figure 4.4, we plotted the error norms while the right part of this figure shows the sigma plots for the three methods. As shown in this figure, the two Arnoldi based methods return good results.

Example 2. For this example, we compared the obtained H_∞ error-norms $\|F - F_m\|_{H_\infty}$, the execution times and the reduced space dimensions for the extended

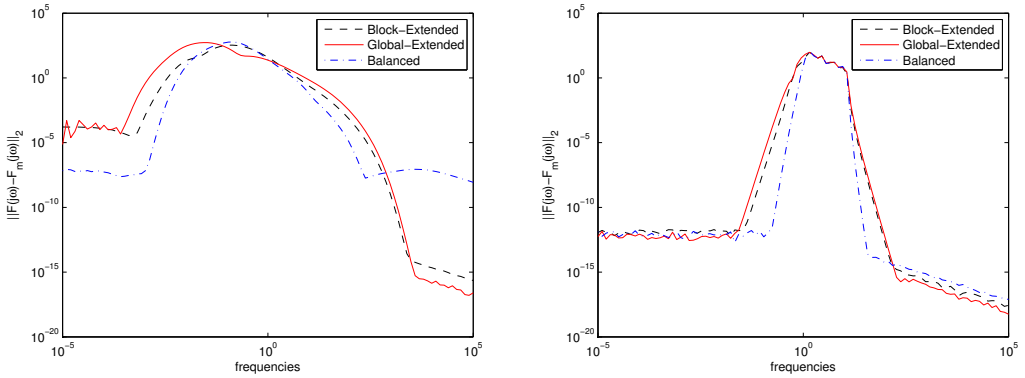

FIGURE 4.3 – The norm of the errors $\|F(i\omega) - F_m(i\omega)\|_2$ for the extended block (dashed), extended global (solid) and the balanced-truncation (dashed-dotted) methods with $\omega \in [10^{-5}, 10^5]$. Left : the `Flow` model with $m = 15$ and $r = 3$. Right : the `Eady` model with $m = 10$ and $r = 3$.

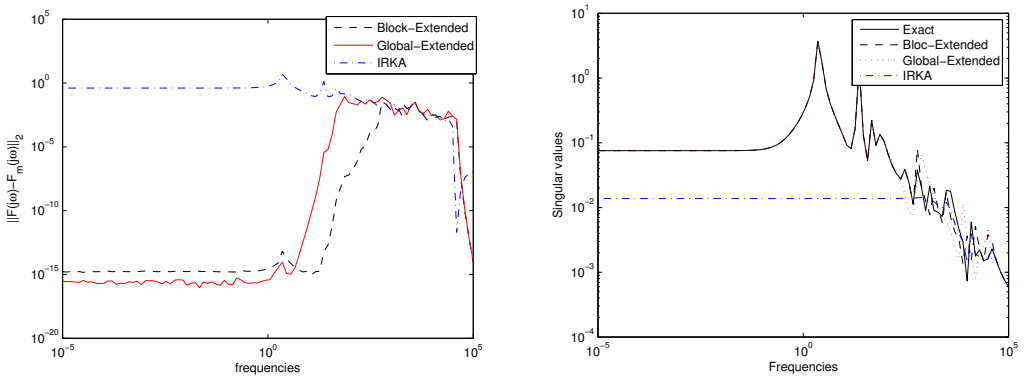

FIGURE 4.4 – The `CDplayer` model. Left : The error norms $\|F(i\omega) - F_m(i\omega)\|_2$ for the extended-block (dashed), extended-global (solid) and IRKA (dashed-dotted) methods with $\omega \in [10^{-5}, 10^5]$. Right : Sigma plots for the three methods. We used $m = 15, r = 2$ for the two extended Arnoldi methods and $m = 30, r = 2$ for IRKA.

block and global Arnoldi algorithms with those obtained by the balanced-truncation method in which the two coupled low-rank right hand sides Lyapunov matrix equations were solved by the extended block Arnoldi algorithm. For the latter method, the inner iterations were stopped when the norm of the residual was less than 10^{-8} and the obtained approximate solution was given as a product of a matrix with a low rank with its transpose. We considered three models : `FDM` with $n = 160000$ and $r = 5$, the `flow-meter` model with $n = 9669$ and $r = 3$, and the `MNA3` model with $n = 4863$ and $r = 4$.

TABLE 4.2 – The H_∞ error-norms $\|F - F_m\|_{H_\infty}$, execution times and reduced space dimensions for extended block, extended global and balanced-truncation methods with the frequencies $\omega \in [10^{-5}, 10^{-2}]$

Model / Method	Bl-Extended	Gl-Extended	B-Truncation
FDM, $n = 160000$, $r = 5$			
H_∞-Error norms	2.5×10^{-4}	4.6×10^{-4}	6.5×10^{-4}
Times in seconds	88	92	364
Space dimension	100	104	120
Flow, $n = 9669$, $r = 3$			
H_∞-Error norms	2.6×10^{-4}	3.9×10^{-4}	1.5×10^{-6}
Times in seconds	1.9	2.1	4.5
Space dimension	180	190	220
MNA3, $n = 4863$, $r = 4$			
H_∞-Error norms	2.6×10^{-6}	3.9×10^{-6}	2.4×10^{-7}
Times in seconds	8.1	8.6	14.8
Space dimension	300	320	360

The results of Table 4.2 show that the cost of balanced truncation method is generally higher than the cost of the extended block or global Arnoldi methods. However some of the obtained H_∞ norms are good when using the balanced truncation method.

4.5 Conclusion

In this chapter, we considered the extended block and global Arnoldi methods. We gave some new algebraic properties of these two algorithms. We also showed how these properties could be used in moment maching methods for model reduction in large-scale dynamical systems. The proposed numerical results on some Benchmark models, show that the extended block and global Arnoldi algorithms are efficient. Generally, the two methods return similar results. One advantage of

the extended global Arnoldi is the fact that a break-down cannot occur which may be the case for the extended block Arnoldi algorithm.

CHAPITRE 5

BALANCED TRUNCATION-RATIONAL KRYLOV METHODS FOR MODEL REDUCTION IN LARGE SCALE DYNAMICAL SYSTEMS

5.1 Introduction

Consider the following linear time invariant (LTI) dynamical system

$$(\textbf{LTI}) \begin{cases} x'(t) = Ax(t) + Bu(t);\ x(t_0) = x_0 \\ y(t) = Cx(t) \end{cases} \tag{5.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^r$, $y(t) \in \mathbb{R}^s$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times r}$ and $C \in \mathbb{R}^{s\times n}$ with $r, s \ll n$. The vector x is called the state vector and it belongs to the state space. The vector u is the input (or the control) vector and $y(t)$ is the output (to be measured). If $s = r = 1$, then the LTI dynamical system (5.1) is called Single-Input Single-Output (SISO) and is called Multiple-Input Multiple-Output (MIMO) otherwise. The control problem consists in acting on the input vector $u(t)$ so that the output vector $y(t)$ has a desirable time trajectory and 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 (5.1) can also be denoted as

$$(\textbf{LTI}) \equiv \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]. \tag{5.2}$$

In many applications, such as circuit simulation, or time dependent PDE control problems, the dimension n of the dynamical system (5.2) is quite large, while the number of inputs and outputs is small $r, s \ll n$. In these large-scale settings, the system dimension makes the computation infeasible due to memory, time limitations and ill-conditioning. The goal is to produce a low dimensional system that has similar response characteristics as the original system with lower storage requirements and evaluation times.

The reduced order dynamical system can be stated as follows

$$(\mathbf{LTI})_m \begin{cases} x'_m(t) = A_m x_m(t) + B_m u(t) \\ y_m(t) = C_m x_m(t) \end{cases} \tag{5.3}$$

where $x_m(t) \in \mathbb{R}^m$, $y_m(t) \in \mathbb{R}^s$, $A_m \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{m\times r}$ and $C_m \in \mathbb{R}^{s\times m}$ with $m \ll n$. The system (5.3) is also represented as

$$(\mathbf{LTI})_m \equiv \left[\begin{array}{c|c} A_m & B_m \\ \hline C_m & 0 \end{array}\right]. \tag{5.4}$$

The reduced order dynamical system (5.3) should be constructed such that

— The output $y_m(t)$ of the reduced system approaches the output $y(t)$ of the original system.

— Some properties of the original system such as passivity and stability (if possible) are preserved.

— The computation methods are robust and efficient.

One of the most known reduction model techniques is the Balanced Model Reduction first introduced by Mullis and Roberts [71] and later in the systems and control literature by Moore [70]. When applied to stable systems, Lyapunov balanced reduction preserves stability and provides a bound for the approximation error. For small-to-medium scale problems, Lyapunov balancing can be implemented efficiently. However, for large-scale settings, exact balancing is expensive to implement because it requires dense matrix factorizations and results in a computational complexity of $O(n^3)$ and a storage requirement of $O(n^2)$, see [7, 13, 40]. For large problems, direct methods could not be applied and then Krylov-based [39, 52, 55, 56] or ADI-based methods [13, 72] are required to compute these Gramians that are given in factored forms which allows to save memory. Besides the Lyapunov balancing method, other types of balancing exist such as stochastic balancing, bounded real balancing, positive real balancing, LQG balancing and frequency weighted

balancing requiring the solution of continuous time algebraic Riccati equations; see [31, 71].

5.2 Lyapunov-balanced truncation

In the sequel we assume that the initial system is stable which means that A is a stable matrix (all its eigenvalues are in the left open part of the complex plane) .

5.2.1 The transfer function

The state space representation is usually referred to 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

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

to the state equation (5.1), and we get

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

where $X(s) = \mathcal{L}(x(t))$ and $U(s) = \mathcal{L}(u(t))$. Therefore

$$X(s) = (sI - A)^{-1}BU(s),$$

and by substituting $X(s)$ in the output equation of (5.1), we get

$$Y(s) = F(s)\,U(s), \tag{5.5}$$

with

$$F(s) = C(sI - A)^{-1}B. \tag{5.6}$$

The rational function $F(s)$ is called the `transfer function` related to the dynamical system (5.1). The elements of this function are real rational functions. The transfer function $F(.)$ is stable if its poles lie in the open left-half plane $\mathbb{C}^-$.
We recall that two LTI systems $\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$ and $\left[\begin{array}{c|c} \widetilde{A} & \widetilde{B} \\ \hline \widetilde{C} & 0 \end{array}\right]$ are called equivalent if

they have the same transfer function. It is easy to verify that for any nonsingular $n \times n$ 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 $\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]$. Therefore, if the main concern is the output 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 $x(t) = T\widetilde{x}(t)$.

5.2.2 Controllability and Observability Gramians

We assume that the LTI dynamical system is stable.

Definition 5.1 *The controllability Gramian associated to the LTI system* (5.1) *is defined as*

$$P = \int_0^\infty e^{tA} B B^T e^{tA^T} dt,$$

and the observability Gramian is defined by

$$Q = \int_0^\infty e^{tA^T} C^T C e^{tA} dt.$$

By using the Parseval relation, we obtain the following expressions of the Gramians

$$\begin{aligned} P &= \int_{-\infty}^{+\infty} (j\omega I - A)^{-1} B B^T (j\omega I - A^T)^{-1} d\omega, \\ Q &= \int_{-\infty}^{+\infty} (j\omega I - A^T)^{-1} C^T C (j\omega I - A)^{-1} d\omega. \end{aligned}$$

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

$$AP + PA^T + BB^T = 0, \tag{5.7}$$

and

$$A^T Q + QA + C^T C = 0. \tag{5.8}$$

We will see later that the product PQ plays an important role in model reduction. Consider the new equivalent LTI dynamical system

$$\widetilde{(\mathbf{LTI})} \equiv \left[\begin{array}{c|c} T^{-1}AT & T^{-1}B \\ \hline CT & 0 \end{array}\right]$$

where T is a nonsingular matrix. Then the associated controllability and observability Gramians $\widetilde{P}$, and $\widetilde{Q}$ are expressed as

$$\widetilde{P} = \int_0^\infty e^{t\widetilde{A}}\widetilde{B}\widetilde{B}^T e^{t\widetilde{A}^T} dt \qquad \text{and} \qquad \widetilde{Q} = \int_0^\infty e^{t\widetilde{A}^T}\widetilde{C}^T\widetilde{C}e^{t\widetilde{A}} dt,$$

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

$$\widetilde{P} = T^{-1}PT^{-T}, \quad \text{and} \quad \widetilde{Q} = T^T QT. \tag{5.9}$$

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

$$\widetilde{P}\widetilde{Q} = T^{-1}PQT.$$

5.2.3 Lyapunov balanced truncation

A well known model reduction scheme called Lyapunov-Balanced Truncation was first introduced by Mullis and Roberts [71] and later in the systems and control by Moore and Glover; see [35, 70]. 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 nonsingular matrix T such that the new Gramians $\widetilde{P}$ and $\widetilde{Q}$ given by (5.9) are such that

$$\widetilde{P} = \widetilde{Q} = diag(\sigma_1, \ldots, \sigma_n)$$

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

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

Let us see how to obtain the matrix T. Consider the Cholesky decompositions of the Gramians P and Q :

$$P = L_c {L_c}^T, \quad Q = L_o {L_o}^T, \tag{5.10}$$

and consider also the singular value decomposition of ${L_c}^T L_o$ as

$${L_c}^T L_o = Z\Sigma Y^T, \tag{5.11}$$

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

$$T = L_c Z \Sigma^{1/2}, \tag{5.12}$$

then it can be easily verified that

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

where Σ is also the diagonal matrix whose elements are the Hankel singular values $\sqrt{\lambda_i(PQ)}$ since PQ is similar to $\widetilde{P}\widetilde{Q}$. There are other possible ways for the construction of the matrix T. It was remarked by Glover [35] that the balanced transformation is not unique but unique up to a nonsingular transformation.
As the concept of balancing has the property that the states which are difficult to reach are also 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 5.1 *[7] Assume that the LTI dynamical system* (5.1) *is stable and minimal and has the following balanced realization*

$$\widetilde{(\mathbf{LTI})} \equiv \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & 0 \end{array}\right],$$

with $P = Q = diag(\Sigma_m, \widetilde{\Sigma}_m)$, $\Sigma_m = diag(\sigma_1, \ldots, \sigma_m)$ *and* $\widetilde{\Sigma}_m = diag(\sigma_{m+1}, \ldots, \sigma_n)$.
Then, the reduced order model is represented by

$$\widetilde{(\mathbf{LTI})}_m \equiv \left[\begin{array}{c|c} A_{11} & B_1 \\ \hline C_1 & 0 \end{array}\right],$$

is stable and we have

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

The preceding theorem shows that if the neglected singular values $\sigma_{m+1}, \ldots, \sigma_n$ are small, then the reduced order LTI system is close to the original one.
Let us see now see how to construct the low order model $(\mathbf{LTI})_m$. We set

$$\mathcal{W}_m = L_o Y_m \Sigma_m^{-1/2} \text{ and } \mathcal{V}_m = L_c Z_m \Sigma_m^{-1/2}, \tag{5.13}$$

where $\Sigma_m = diag(\sigma_1, \ldots, \sigma_m)$ and Z_m and Y_m correspond to the leading m columns of the matrices Z and Y given by the singular value decomposition (5.11). The matrices of the reduced LTI system

$$\widetilde{(\mathbf{LTI})}_m \equiv \left[\begin{array}{c|c} A_m & B_m \\ \hline C_m & 0 \end{array}\right],$$

are given by

$$A_m = \mathcal{W}_m^T A \mathcal{V}_m,\ B_m = \mathcal{W}_m^T B \text{ and } C_m = C\mathcal{V}_m. \tag{5.14}$$

Notice that $\mathcal{V}_m \mathcal{W}_m^T$ is an oblique projector, $\widetilde{P}\mathcal{W}_m = \mathcal{V}_m \Sigma_m$ and $\widetilde{Q}\mathcal{V}_m = \mathcal{W}_m \Sigma_m$. 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 in factored forms and use them to construct an approximate Lyapunov-balanced truncation model.
Let $\widetilde{A}$, $\widetilde{B}$ and $\widetilde{C}$ be the following matrices

$$\widetilde{A} = \begin{pmatrix} A & 0 \\ 0 & A_m \end{pmatrix},\ \widetilde{B} = \begin{pmatrix} B \\ B_m \end{pmatrix},\ \widetilde{C} = \begin{pmatrix} C & C_m \end{pmatrix}. \tag{5.15}$$

Then, the Gramians corresponding to the error dynamical system

$$\widetilde{(\mathbf{LTI})} \equiv \left(\begin{array}{c|c} \widetilde{A} & \widetilde{B} \\ \hline \widetilde{C} & 0 \end{array}\right)$$

are the solutions of the following Lyapunov matrix equations

$$\widetilde{A}\widetilde{P} + \widetilde{P}\widetilde{A}^T + \widetilde{B}\widetilde{B}^T = 0 \qquad \text{and} \qquad \widetilde{A}^T\widetilde{Q} + \widetilde{Q}\widetilde{A} + \widetilde{C}^T\widetilde{C} = 0.$$

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

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

We notice that other model reduction techniques such as the Cross-Gramian method [7] requires the solution of large Sylvester matrix equations to construct the reduced order model. Next, we apply the rational block Arnoldi algorithm for solving large Lyapunov (or in general large Sylvester) matrix equations that are used in the construction of reduced order models using the balanced truncation techniques.

5.3 The rational block Arnoldi method for solving large Sylvester matrix equations

Consider the following Sylvester matrix equation

$$AX + XD + EF^T = 0, \tag{5.16}$$

where $A \in \mathbb{R}^{n\times n}$ and $D \in \mathbb{R}^{p\times p}$ are large and sparse stable matrices. We assume that $E \in \mathbb{R}^{n\times r}$, and $F \in \mathbb{R}^{p\times r}$ are of full rank r, with $r \ll n, p$.

The Bartels-Stewart algorithm [11] is the standard and widely used direct method for the solution of Sylvester equations of small to moderate size. Therefore, this direct method is unsuitable when either one of the matrices A or D is of medium size or large and sparse. For medium and large coefficient matrices, iterative schemes have to be used. Krylov-type subspace methods such as those based on the Arnoldi process [39, 55, 56, 81] are attractive if the matrices are sparse and if no information about the spectra of A and D is available. The Smith method [72] and the alternating directional implicit (in short ADI) iterations could also be applied if a spectral information about A and D is given. Note that, ADI iterations allow faster convergence if sub-optimal shifts to A and D can be effectively computed and linear systems with shifted coefficient matrices are solved effectively at low cost. Here, we will use a method based on the rational Krylov subspace.

Let us first recall the following rational block Krylov subspaces

$$\mathbf{K}_m(A, E) = \text{Range}\{E, (A - s_2 I)^{-1}E, \ldots, \prod_{i=2}^{m}(A - s_i I)^{-1}E\}, \tag{5.17}$$

and

$$\mathbf{K}_m(D^T, F) = \text{Range}\{F, (D^T - \tilde{s}_2 I)^{-1}F, \ldots, \prod_{i=2}^{m}(D^T - \tilde{s}_i I)^{-1}F\}, \tag{5.18}$$

where the shift-parameters s_i and $\tilde{s}_i$, $i = 2, \ldots, m$ are generated during the construction of the process are selected a posteriori. In our numerical tests, we used two strategies : the first one is an priori selection from Lyapack [68] and the second strategy consists in selecting, at each iteration, a new shift s_{m+1} which is used to compute a new basis vector. For the second case, we used the adaptive selection descibed in Chapter 2.

The rational block Arnoldi algorithm for the pair (A, V) where $V \in \mathbb{R}^{n\times r}$ is summarized as follows.

Algorithm 5.1 The Rational block Arnoldi Algorithm (RBA)

— *Input :* $A \in \mathbb{R}^{n\times n}$, $V \in \mathbb{R}^{n\times r}$ and a fixed integer m.

— Compute $V_1 = QR(V)$, $\mathcal{V}_1 = [V_1]$.

— For $j = 1, \ldots, m-1$

1. $\widetilde{V}_{j+1} = (A - s_{j+1}I)^{-1}V_j$.
2. Orthogonalization step :

   For $i = 1, 2, \ldots, j$

   $H_{i,j} = V_i^T\widetilde{V}_{j+1}$;

   $\widetilde{V}_{j+1} = \widetilde{V}_{j+1} - V_iH_{i,j}$;

   End For
3. $QR\big(\widetilde{V}_{j+1}\big) = V_{j+1}H_{j+1,j}$.
4. $\mathcal{V}_{j+1} = [\mathcal{V}_j,\ V_{j+1}]$.

— End For.

After m steps, the rational block Arnoldi algorithm generates a block matrix $\mathcal{V}_m = [V_1, \ldots, V_m] \in \mathbb{R}^{n\times mr}$ whose columns form an orthonormal basis of the rational block Krylov subspace $\mathbf{K}_m(A, V)$ and an upper $(m+1)r \times mr$ block Hessenberg matrix $\overline{\mathcal{H}}_{A,m}$ whose blocks $H^A_{i,j}$ are defined by Algorithm 5.1. The $mr \times mr$ upper block Hessenberg matrix $\mathcal{H}_{A,m}$ is obtained from $\overline{\mathcal{H}}_{A,m}$ by deleting its last r-rows. When applied to the pairs (A, E) and (D^T, F), the rational block Arnoldi algorithm constructs a system of matrices $\{V_1, \ldots, V_m\}$ and $\{W_1, \ldots, W_m\}$ forming two orthonormal bases of the rational block Krylov subspaces $\mathbf{K}_m(A, E)$ and $\mathbf{K}_m(D^T, F)$, respectively. Let

$$\mathcal{T}_{A,m} = \mathcal{V}_m^T A\mathcal{V}_m, \qquad \mathcal{T}_{D,m} = \mathcal{W}_m^T D^T\mathcal{W}_m, \tag{5.19}$$

where $\mathcal{V}_m = [V_1, \ldots, V_m]$ and $\mathcal{W}_m = [W_1, \ldots, W_m]$. The matrices $\mathcal{T}_{A,m}$ and $\mathcal{T}_{D,m}$ could be obtained directly from $\mathcal{H}_{A,m}$ and $\mathcal{H}_{D,m}$, respectively ; see Chapter refChapitre2 as follows

$$\mathcal{T}_{A,m} = (I_{mp} + \mathcal{H}_{A,m}S_m - \mathcal{V}_m^* AV_{m+1}H^A_{m+1,m}E_m^*)\mathcal{H}^{-1}_{A,m},$$

and $S_m = diag(s_2I_r, \ldots, s_{m+1}I_r)$, and $E_m^T = [0_r, \ldots, 0_r, I_r] = (e_m^T \otimes I_r)$.

From Algorithm 5.1, we can deduce the following relations

$$A\mathcal{V}_m = \mathcal{V}_m\mathcal{T}_{A,m} - \Phi_{A,m}H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m} + V_{m+1}H^A_{m+1,m}E_m^TS_m\mathcal{H}^{-1}_{A,m}, \quad (5.20)$$

where

$$\Phi_{A,m} = (I_n - \mathcal{V}_m\mathcal{V}_m^*)AV_{m+1}. \quad (5.21)$$

We also have

$$D^T\mathcal{W}_m = \mathcal{W}_m\mathcal{T}_{D,m} - \Phi_{D,m}H^D_{m+1,m}E_m^T\mathcal{H}^{-1}_{D,m} + W_{m+1}H^D_{m+1,m}E_m^T\tilde{S}_m\mathcal{H}^{-1}_{D,m}, \quad (5.22)$$

where

$$\Phi_{D,m} = (I_n - \mathcal{W}_m\mathcal{W}_m^T)D^TW_{m+1}, \quad (5.23)$$

and $\tilde{S}_m = diag(\tilde{s}_2I_r, \ldots, \tilde{s}_{m+1}I_r)$.

We have $E_m^TS_m = (e_m^T \otimes I_r)(D_m \otimes I_r)$ where $D_m = diag(s_2, \ldots, s_{m+1})$, and then $E_m^TS_m = (e_m^TD_m \otimes I_r) = s_{m+1}(e_m^T \otimes I_r) = s_{m+1}E_m^T$, and $E_m^T\tilde{S}_m = \tilde{s}_{m+1}E_m^T$. Using the relations given in (5.20) and (5.22), we get

$$A\mathcal{V}_m = \mathcal{V}_m\mathcal{T}_{A,m} + (s_{m+1}V_{m+1} - \Phi_{A,m})H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m}, \quad (5.24)$$

and

$$D^T\mathcal{W}_m = \mathcal{W}_m\mathcal{T}_{D,m} + (\tilde{s}_{m+1}W_{m+1} - \Phi_{D,m})H^D_{m+1,m}E_m^T\mathcal{H}^{-1}_{D,m}. \quad (5.25)$$

When applying Krylov based methods for solving the Sylvester matrix equation (5.16), one seeks for a low rank approximate solution of the form

$$\mathcal{X}_m = \mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T,$$

such that the following Galerkin condition is satisfied

$$\mathcal{V}_m^T\mathcal{R}(\mathcal{X}_m)\mathcal{W}_m = 0, \quad (5.26)$$

where $\mathcal{R}(\mathcal{X}_m)$ is the residual corresponding to the approximation $\mathcal{X}_m$ and given by

$$\mathcal{R}(\mathcal{X}_m) = A\mathcal{X}_m + \mathcal{X}_mD + EF^T. \quad (5.27)$$

Therefore, replacing $\mathcal{X}_m$ by $\mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T$ in (5.26), we obtain

$$\mathcal{V}_m^TA\mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T\mathcal{W}_m + \mathcal{V}_m^T\mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^TD\mathcal{W}_m + \mathcal{V}_m^TEF^T\mathcal{W}_m = 0.$$

We get the following low-dimensional Sylvester matrix equation

$$\mathcal{T}_{A,m}\mathcal{Y}_m + \mathcal{Y}_m\mathcal{T}_{D,m}^T + (\mathcal{V}_m^T E)(\mathcal{W}_m^T F)^T = 0. \tag{5.28}$$

Now, as $[V_1, R] = QR(E)$ and $[W_1, S] = QR(F)$ (the QR factorisation of the matrices V_1 and W_1, respectively), we have

$$\mathcal{V}_m^T E = \mathcal{V}_m^T V_1 R = E_1 R \qquad \text{et} \qquad \mathcal{W}_m^T F = \mathcal{W}_m^T W_1 S = E_1 S,$$

with $E_1 = e_1 \otimes I_r$. The Sylvester matrix equation (5.28) will be solved by a direct method such as the Bartels-Stewart algorithm [11]. In the next theorem, we give a computational expression for the norm of the residual without computing the approximate solution which is given only at the end of the process and in a factored form.

Theorem 5.2 *Let $\mathcal{V}_m = [V_1, \ldots, V_m]$ and $\mathcal{W}_m = [W_1, \ldots, W_m]$ the matrices whose columns form bases of the rational Krylov subspaces given by (5.17) and (5.18), respectively. Let $\mathcal{X}_m = \mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T$ be the approximate solution of the Sylvester matrix equation (5.16), then the residual norm is given as follows*

$$\|\mathcal{R}(\mathcal{X}_m)\|_2 = \|S_1 J S_2^T\|_2, \qquad J = \begin{bmatrix} 0 & I_r \\ I_r & o \end{bmatrix}$$

where S_1 and S_2 are the 2×2 upper triangular matrices obtained from the QR decomposition of the matrices

$$U_1 = \left[\mathcal{V}_m\mathcal{Y}_m\mathcal{H}_{D,m}^{-T}E_m H_{m+1,m}^{D^T} \qquad s_{m+1}V_{m+1} - \Phi_{A,m}\right]$$

and

$$U_2 = \left[\mathcal{W}_m\mathcal{Y}_m^T\mathcal{H}_{A,m}^{-T}E_m H_{m+1,m}^{A^T} \qquad \tilde{s}_{m+1}W_{m+1} - \Phi_{D,m}\right].$$

The quantities $\Phi_{A,m}$ and $\Phi_{D,m}$ are given by the expressions (5.21) and (5.23), respectively.

Proof We have

$$\begin{aligned} \mathcal{R}(\mathcal{X}_m) &= A\mathcal{X}_m + \mathcal{X}_m D + EF^T \\ &= A\mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T + \mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T D + EF^T. \end{aligned}$$

Replacing $A\mathcal{V}_m$ and $\mathcal{W}_m^T D$ by (5.24) and (5.25), respectively, we get

$$\begin{aligned}
\mathcal{R}(X_m) &= \big[\mathcal{V}_m\mathcal{T}_{A,m} + (s_{m+1}V_{m+1} - \Phi_{A,m})H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m}\big]\mathcal{Y}_m\mathcal{W}_m^T \\
&+ \mathcal{V}_m\mathcal{Y}_m\big[\mathcal{W}_m\mathcal{T}_{D,m} + (\tilde{s}_{m+1}W_{m+1} - \Phi_{D,m})H^D_{m+1,m}E_m^T\mathcal{H}^{-1}_{D,m}\big]^T + EF^T \\
&= \mathcal{V}_m\mathcal{T}_{A,m}\mathcal{Y}_m\mathcal{W}_m^T + (s_{m+1}V_{m+1} - \Phi_{A,m})H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m}\mathcal{Y}_m\mathcal{W}_m^T \\
&+ \mathcal{V}_m\mathcal{Y}_m\mathcal{T}^T_{D,m}\mathcal{W}_m^T + \mathcal{V}_m\mathcal{Y}_m\mathcal{H}^{-T}_{D,m}E_mH^{D^T}_{m+1,m}(\tilde{s}_{m+1}W_{m+1}^T - \Phi^T_{D,m}) \\
&+ EF^T.
\end{aligned}$$

Using the fact that $\mathcal{Y}_m$ solves the low dimentional Sylvester equation (5.28), it follows that

$$\begin{aligned}
\mathcal{R}(\mathcal{X}_m) &= (s_{m+1}V_{m+1} - \Phi_{A,m})H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m}\mathcal{Y}_m\mathcal{W}_m^T \\
&+ \mathcal{V}_m\mathcal{Y}_m\mathcal{H}^{-T}_{D,m}E_mH^{D^T}_{m+1,m}(\tilde{s}_{m+1}W_{m+1}^T - \Phi^T_{D,m}) \\
&= \begin{bmatrix}\mathcal{V}_m\mathcal{Y}_m\mathcal{H}^{-T}_{D,m}E_mH^{D^T}_{m+1,m} & s_{m+1}V_{m+1} - \Phi_{A,m}\end{bmatrix}\begin{bmatrix}0 & I_r\\ I_r & o\end{bmatrix} \\
&\times \begin{bmatrix}H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m}\mathcal{Y}_m\mathcal{W}_m^T \\ \tilde{s}_{m+1}W_{m+1}^T - \Phi^T_{D,m}\end{bmatrix} \\
&= U_1JU_2^T.
\end{aligned}$$

Therefore, using the QR factorizations of $U_1 = \widetilde{Q}_1S_1$ and $U_2 = \widetilde{Q}_2S_2$, the result follows. The following result shows that $\mathcal{X}_m$ is an exact solution of a perturbed Sylvester matrix equation.

Proposition 5.1 *The approximate solution $\mathcal{X}_m = \mathcal{V}_m\mathcal{Y}_m\mathcal{W}_m^T$ solves the following perturbed Sylvester matrix equation*

$$\big[A - \Delta_{A,m}\big]\mathcal{X}_m + \mathcal{X}_m\big[D - \Delta_{D,m}\big] + EF^T = 0,$$

where

$$\Delta_{A,m} = (s_{m+1}V_{m+1} - \Phi_{A,m})H^A_{m+1,m}E_m^T\mathcal{H}^{-1}_{A,m}\mathcal{V}_m^T,$$

and

$$\Delta_{D,m} = \mathcal{W}_m\mathcal{H}^{-T}_{D,m}E_mH^{D^T}_{m+1,m}(\tilde{s}_{m+1}W_{m+1}^T - \Phi^T_{D,m}).$$

Proof Multiplying equation (5.28) from the left by $\mathcal{V}_m$ and from the right by $\mathcal{W}_m^T$, and using the relations (5.24) and (5.25), the result follows.

An important issue when dealing with high dimensional problems is the storage that requires a large amount of memory. In order to save memory, we can give the approximation $\mathcal{X}_m$ in a factored form. Let $\mathcal{Y}_m = \widetilde{U}\widetilde{\Sigma}\widetilde{V}^T$ be the SVD of $\mathcal{Y}_m$ where $\widetilde{\Sigma}$ is the matrix of the singular values of $\mathcal{Y}_m$ sorted in decreasing order; $\widetilde{U}$ and $\widetilde{V}$ are unitary matrices. We choose a tolerance $dtol$ and define $\widetilde{U}_l$ and $\widetilde{V}_l$ the matrices of the first l columns of $\widetilde{U}$ and $\widetilde{V}$ corresponding to the l singular values of magnitude greater than $dtol$.
Setting $\widetilde{\Sigma}_l = \text{ diag}[\sigma_1, ..., \sigma_l]$, we get the approximation $\mathcal{Y}_m \approx \widetilde{U}_l\widetilde{\Sigma}_l\widetilde{V}_l^T$ (which is the best l-rank approximation of $\mathcal{Y}_m$).
Then we have the low rank approximation

$$\mathcal{X}_m \approx Z_m^A (Z_m^D)^T, \tag{5.29}$$

with $Z_m^A = \mathcal{V}_m\widetilde{U}_l\widetilde{\Sigma}_l^{1/2}$ and $Z_m^D = \mathcal{W}_m\widetilde{V}_l\widetilde{\Sigma}_l^{1/2}$.

Remark 5.1 *When considering the Lyapunov balanced truncation method, we have seen that one has to compute the Gramians P and Q by solving the two coupled Lyapunov matrix equations (5.7) and (5.8), respectively. For large problems, these Gramians are computed by using the rational block Arnoldi algorithm and given in factored form $P \approx Z_P {Z_P}^T$ with $Q \approx Z_Q {Z_Q}^T$. These factorizations are then used instead of Cholesky factors to build the balanced truncation reduced order model.*

The rational block Arnoldi algorithm for solving large-scale Sylvester matrix equations is given as follows

5.4 The Riccati-balanced truncation method

5.4.1 The LQG-Riccati method for model reduction

In this subsection, we assume that the original system is no longer stable and present a reduced order model method called the Linear Quadratic Gaussian (LQG) balanced truncation method; see [31, 71]. The basic idea of (LQG) balanced truncation is to replace the Lyapunov Gramians P and Q used for the classical balanced truncation for stable systems by the stabilizing solutions of the dual Kalman Filtering Algebraic Riccati Equation (FARE) and Continuous-time Algebraic Riccati

Algorithm 5.2 The rational block Arnoldi algorithm for Sylvester equations (RBAS)

— Input : $A \in \mathbb{R}^{n\times n}$, $D \in \mathbb{R}^{p\times p}$, $E \in \mathbb{R}^{n\times r}$, $F \in \mathbb{R}^{p\times r}$.

— Choose ϵ, $m_{\max}$ and $dtol$.

— For $m = 1, 2, \ldots, m_{\max}$

1. Apply the rational block Arnoldi algorithm to the pairs (A, E) and (D^T, F) to get $\mathcal{V}_m$, $\mathcal{W}_m$, $\mathcal{T}_{A,m}$ and $\mathcal{T}_{D,m}$.
2. Solve for $\mathcal{Y}_m$ the low dimensional Sylvester equation (5.28) and compute the residual norm $\|\mathcal{R}(X_m)\|_2$ using Theorem 5.2.
   If $\|\mathcal{R}(X_m)\|_2 < \epsilon$, stop.

— End

— Compute $Z^A_{m_{\max}}$ and $Z^D_{m_{\max}}$ to get $\mathcal{X}_{m_{\max}}$ from (5.29) in a factored form.

Equation (CARE) defined as follows

$$AP + PA^T - PC^TCP + BB^T = 0, \quad (FARE) \tag{5.30}$$

and

$$A^TQ + QA - QBB^TQ + C^TC = 0. \quad (CARE) \tag{5.31}$$

Assuming that the classical conditions of controllability and detectability are satisfied, let P_+ and Q_+ be the stabilizing and positive semidefinite solutions of the matrix Riccati equations (FARE) and (CARE), respectively which means that the eigenvalues of the closed loops $A - P_+C^TC$ and $A^T - Q_+BB^T$ lie in the open left half plane $\mathbb{C}^-$. It is known [31] that, as for the classical balanced truncation, the eigenvalues of the product P_+Q_+ are invariant quantities under any state coordinate transformation $\tilde{x}(t) = Tx(t)$ where T is a nonsingular $n \times n$ matrix and we have

$$\widetilde{P}_+\widetilde{Q}_+ = TP_+Q_+T^{-1}.$$

The Riccati Gramians P_+ and Q_+ are then used as we explained in Subsection 4.2, to construct the model reduction in the same way as when using balanced truncation via the Lyapunov Gramians obtained by solving two coupled Lyapunov matrix equations. Here those Lyapunov Gramians are replaced by the Riccati ones : P_+ and Q_+.

As we mentioned earlier, the solutions P_+ and Q_+ have usually low numerical rank and could then be approximated by low rank factorizations $P_+ \approx Z_m Z_m^T$ and $Q_+ \approx Y_m Y_m^T$ where the matrix factors Y_m and Z_m have low ranks. As in the classical Lyapunov balanced truncation method, the factors Y_m and Z_m could be used to consruct the (LQG)-balanced truncation reduced order model.
Next, we show how to construct low rank approximate solutions to algebraic Riccati equations (5.30) and (5.31) by using the rational block Arnoldi process.

5.4.2 The rational block Arnoldi for continuous-time algebraic Riccati equations

Consider the continuous-time algebraic Riccati equation

$$A^T X + XA - XBB^T X + C^T C = 0, \tag{5.32}$$

where $A \in \mathbb{R}^{n\times n}$ is nonsingular, $B \in \mathbb{R}^{n\times r}$ and $C \in \mathbb{R}^{r\times n}$. The matrices B and C are assumed to be of full rank with $r \ll n$.
Riccati equations play a fundamental role in many other areas such as control, filter design theory, differential equations and robust control problems. For historical developments, applications and importance of algebraic Riccati equations, we refer to [5, 15, 23, 46, 57, 58, 63] and the references therein.
Under the hypotheses : the pair (A, B) is stabilizable (i.e., $\exists$ a matrix S such that $A - B\,S$ is stable) and the pair (C, A) is detectable (i.e., (A^T, C^T) stabilizable), equation (5.32) has a unique symmetric positive semidefinite and stabilizing solution.
To extract low rank approximate solutions to the continuous-time algebraic Riccati equation (5.32), we project the initial problem onto the rational block Krylov subspace $\mathcal{K}_m(A^T, C^T)$. Applying the Rational block Arnoldi process (Algorithm 5.1) to the pair (A^T, C^T) gives us an orthonormal basis $\{V_1, \ldots, V_m\}$ of $\mathcal{K}_m(A^T, C^T)$.
We consider low-rank approximate solutions that have the form

$$\mathcal{X}_m = \mathcal{V}_m\, \mathcal{Y}_m\, \mathcal{V}_m^T, \tag{5.33}$$

where $\mathcal{V}_m = [V_1, \ldots, V_m]$ and $\mathcal{Y}_m \in \mathbb{R}^{2mr\times 2mr}$.

From now on, the matrix $\mathcal{T}_m$ is defined by $\mathcal{T}_m = \mathcal{V}_m^T\, A^T\, \mathcal{V}_m$.
Setting $R_m(\mathcal{X}_m) = A^T\mathcal{X}_m + \mathcal{X}_m A - \mathcal{X}_m BB^T\mathcal{X}_m + C^T C$ and using the Galerkin

condition $\mathcal{V}^T\mathcal{R}_m(\mathcal{X}_m)\mathcal{W} = 0$, we get the low-dimensional continuous-time algebraic Riccati equation

$$\mathcal{T}_m\,\mathcal{Y}_m + \mathcal{Y}_m\,\mathcal{T}_m^T - \mathcal{Y}_m\,\tilde{B}_m\,\tilde{B}_m^T\,\mathcal{Y}_m + \tilde{C}_m^T\,\tilde{C}_m = 0, \tag{5.34}$$

where $\tilde{B}_m = \mathcal{V}_m^T B$, $\tilde{C}_m = C\mathcal{V}_m$. We assume that the projected algebraic Riccati equation (5.34) has a unique symmetric positive semidefinite and stabilizing solution $\mathcal{Y}_m$. This solution can be obtained by a standard direct method such as the Schur method [65].

Proposition 5.2 *Let $\mathcal{V}_m = [V_1, \ldots, V_m]$ whose columns form an orthonormal basis of the rational block Krylov subspace $\mathbf{K}_m(A^T, C^T)$. Let $\mathcal{X}_m = \mathcal{V}_m\mathcal{Y}_m\mathcal{V}_m^T$ the approximate solution given by (5.33) and let $\mathcal{R}_m(\mathcal{X}_m)$ be the corresponding residual. Then*

$$\|\mathcal{R}_m(\mathcal{X}_m)\|_2 = \|\tilde{S}J\tilde{S}^T\|_2, \qquad J = \begin{bmatrix} 0 & I_r \\ I_r & o \end{bmatrix}, \tag{5.35}$$

where $\tilde{S}$ is the 2×2 upper triangular matrix obtained from the QR decomposition of

$$\tilde{U} = \begin{bmatrix} \mathcal{V}_m\mathcal{Y}_m\mathcal{H}_{A^T,m}^{-T}E_m H_{m+1,m}^{A^T} & s_{m+1}V_{m+1} - \Phi_{A^T,m} \end{bmatrix} \tag{5.36}$$

Proof The proof is similar to the one given for Theorem 5.2.

Remark 5.2 *The approximate solution $\mathcal{X}_m$ could also be approximated here as a product of two matrices of low ranks. Consider the eigendecomposition of the matrix $\mathcal{Y}_m = \widetilde{U}\,\widetilde{D}\,\widetilde{U}^T$ where $\widetilde{D}$ is the diagonal matrix of the eigenvalues of the symmetric and positive semi-definite solution $\mathcal{Y}_m$ sorted in decreasing order. Let $\widetilde{U}_l$ be the matrix of the first l columns of $\widetilde{U}$ corresponding to the l eigenvalues of magnitude greater than some tolerance $toler$. We obtain the truncated eigendecomposition $\mathcal{Y}_m \approx \widetilde{U}_l\,D_l\,\widetilde{U}_l^T$ where $D_l = \mathrm{diag}[\lambda_1, \ldots, \lambda_l]$. Setting $\mathcal{Z}_m = \mathcal{V}_m\,\widetilde{U}_l\,D_l^{1/2}$, we get*

$$\mathcal{X}_m \approx \mathcal{Z}_m\mathcal{Z}_m^T. \tag{5.37}$$

As we have seen earlier for, the factor $\mathcal{Z}_m$ is used to build the low order (LQG) balanced truncation model.

The Rational block Arnoldi algorithm for computing low rank approximate solutions to the continuous-time algebraic Riccati equation (5.32) is described as follows

Algorithm 5.3 The rational block Arnoldi algorithm for CAREs (RBA-CARE)

— Input : $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times r}$, $C \in \mathbb{R}^{r\times n}$.

— Choose ϵ, $m_{\max}$ and $toler$.

— For $m = 1, 2, \ldots, m_{\max}$

1. Apply the rational block Arnoldi algorithm to the pairs (A^T, C^T) to get $\mathcal{V}_m$ and the block Hessenberg matrix $\mathcal{T}_m$..
2. Solve for $\mathcal{Y}_m$ the low dimensional CARE (5.34) and compute the residual norm $\|\mathcal{R}(X_m)\|_2$ using Theorem 5.2.
   If $\|\mathcal{R}(X_m)\|_2 < \epsilon$, stop.

— End

— Compute $\mathcal{Z}_{m_{\max}}$ to get $\mathcal{X}_{m_{\max}}$ from (5.32) in a factored form.

5.5 Numerical experiments

In this section, we give some experimental results to show the effectiveness of the proposed approaches. The experiments were performed on a computer of Intel Core i5 at 1.3GHz and 8GB of RAM. The algorithms were coded in Matlab R2010a and we used different known benchmark models listed in Table 5.1.

TABLE 5.1 – Test matrices

Matrix A	size n	$\|A\|_F$	$cond(A)$
`FOM`	1006	1.8283e+04	1006
`CDplayer`	120	2.3095e+05	1.8149e+04
`Flow`	9669	2.5438e+04	1.6193e+07
`FDM`	90.000	1.56e+08	7.8768e+04

The matrices for the benchmark problems `CDplayer`, `FOM` were obtained from NICONET [68] while the matrices for the `Flow` model were obtained from the discretization of a 2D convective thermal flow problem (flow meter model v0.5) from the Oberwolfach collection [1]. Some information on these matrices is reported

1. Oberwolfach model reduction benchmark collection, 2003.
http ://www.imtek.de/simulation/benchmark

in Table 5.1. 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

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

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 $n_0 = 300$ and the dimension of A is $n = n_0^2$.

Example 1. In the first experiment, we considered the models CDplayer and FOM. For the FOM model B and C are random matrices. Although the matrices of these models have small sizes they are usually considered as benchmark examples. In Figure 5.1 we plotted the maximum singular values of the exact (solid) and approximated (dashed) transfer functions for the CDplayer (left) with a space dimension of $m = 25$ and the FOM model (right) with a space dimension of $m = 20$. For the two models we used $s = r = 2$.

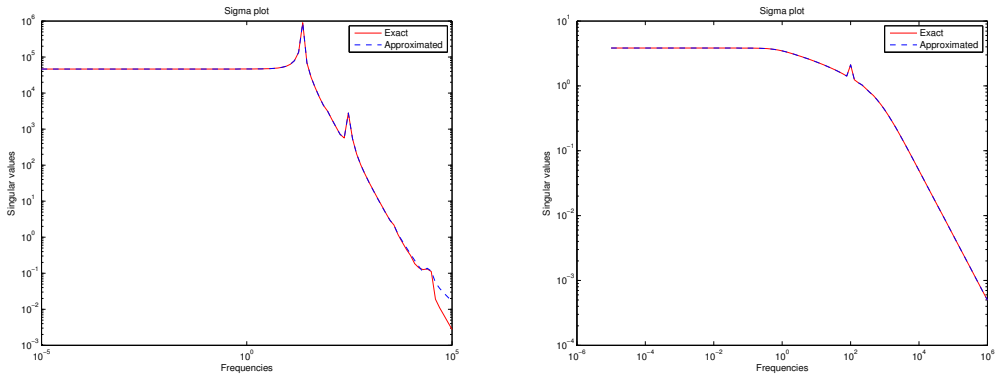

FIGURE 5.1 – The maximum singular values for the balanced-rational block Arnoldi (solid) and IRKA (dashed) with $\omega \in [10^{-5}, 10^5]$. Left the CDplayer model with $r = s = 2$. Right : the FOM model with $s = r = 2$.

The plots of Figure 5.2 show the norms of the errors for the balanced-rational block Arnoldi (solid) and IRKA (dashed) with $\omega \in [10^{-5}, 10^5]$ for CDplayer (left figure)

and FOM (right figure). The used sizes of the reduced order dynamical systems are $m = 30$ for the CDplayer model and $m = 40$ for the FOM model.

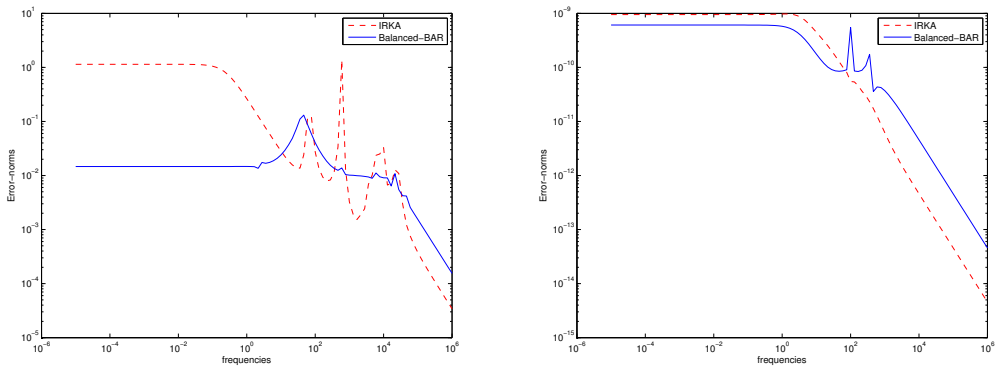

FIGURE 5.2 – The norms of the errors for the balanced-rational block Arnoldi (solid) and IRKA (dashed) with $\omega \in [10^{-5}, 10^{5}]$. Left the CDplayer model with $r = s = 2$. Right : the FOM model with $r = s = 2$.

Example 2. For this example, we compared the performances of the balanced Rational block Arnoldi and the IRKA algorithms. In Table 5.2, we reported the H_∞ norm of the errors, the size of the reduced order system and the execution times. As seen from this table, IRKA has difficulties for large-scale problems and cannot converge within the maximum of 300 seconds. The $\mathcal{H}_\infty$ norm of the errors was computed for the frequencies : $\omega \in [10^{-5},\ 10^{5}]$.

TABLE 5.2 – The $\mathcal{H}_\infty$ of the transfert function errors, the size of the reduced order system and the execution times for Rational balanced-truncation and IRKA methods

Model / Method	Rational−Balanced	IRKA
FDM, $n = 90.000$, $p = 3$, $dim = 80$,		
$\mathcal{H}_\infty$-error norms	$2.5^{\cdot}10^{-9}$	$--$
Times in seconds	109	> 300
Flow , $n = 9669$, $p = 3$, $dim = 100$		
$\mathcal{H}_\infty$-error norms	$2.6^{\cdot}10^{-6}$	$--$
Times in seconds	13.5	> 300

5.6 Conclusion

In this chapter, we proposed a new method based on the rational block Arnoldi algorithm to compute low rank approximate solutions to large Sylvester (or Lyapunov) and continuous-time algebraic Riccati equations having low rank right-hand sides. These approximate solutions are given in factored forms and are used to build reduced order models that approximate the original large-scale dynamical linear system. We showed how the obtained approximate Gramians could be used in the balanced truncation method. We gave some theoretical results and present numerical experiments on some benchmark examples.

CHAPITRE 6

THE EXTENDED-RATIONAL KRYLOV METHOD

In this chapter we introduce a new method that will be used for reducing the transfer function and can be extended to approach the solutions of Sylvester and Riccati equations.
The general idea of this method is to provide a new Krylov subspace that is richer than the rational Krylov subspace as well as the extended Krylov subspace [66, 67]. This idea comes from the lack of information on the matrix A when using rational Krylov subspace. That is why, we introduce a new method that we name the extended-rational Krylov method.

6.1 A new Krylov subspace method

In this section we introduce the extended-rational Krylov subspace and the extended-rational Arnoldi algorithm, an algorithm used to build an orthonormal basis of this subspace.
For $A \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{n\times s}$, the extended-rational Krylov subspace is the subspace of $\mathbb{R}^n$ spanned by the columns of the matrices $A^k\,V$, $k = 0, \ldots, m-1$ and $(A - s_1 I)^{-1}\,V, (A - s_1 I)^{-1}(A - s_2 I)^{-1}\,V, \ldots, \prod_{i=1}^{m}(A - s_i I)^{-1}\,V$. This subspace is denoted by

$$\mathbb{K}_m^{er}(A, V) = \text{Range}\{\prod_{i=1}^{m}(A - s_i I)^{-1}\,V, \ldots, , (A - s_1 I)^{-1}\,V, V, A\,V, \ldots, A^{m-1}\,V\}.$$

where $s_2, \ldots, s_m$ are some selected complex parameters.
As in all projection methods we have to extract an orthonormal basis of the extended-

rational Krylov subspace where the projection occurs. To get that, we introduce the extended-rational Arnoldi algorithm based on the Gram-Schmidt orthogonalization process.

Starting with the pair $\{V, (A - s_1 I)^{-1} V\}$ and by adding two vectors at a time, one multiplied by A and the other one by $(A - s_j I)^{-1} V$ at each iteration, the extended-rational Arnoldi algorithm generates a sequence of blocks $\{V_1, \ldots, V_m\}$ of size $n \times 2s$ such that their columns form an orthonormal basis of the extended-rational Krylov subspace $\mathbb{K}_m^{er}(A, V)$. The algorithm is defined as follows.

Algorithm 6.1 The extended-rational block Arnoldi algorithm

- Inputs : $A \in \mathbb{R}^{n\times n}$, $V \in \mathbb{R}^{n\times s}$, $\{s_1, \ldots, s_m\} \subset \mathbb{C}$ and m.
- Compute $[V_1, \Lambda] = QR([V, (A - s_1 I)^{-1} V])$, $\mathbb{V}_1 = [V_1]$.
- For $j = 1, \ldots, m$
  1. Set $V_j^{(1)}$: first s columns of V_j ; $V_j^{(2)}$: second s columns of V_j.
  2. $\widetilde{V}_{j+1} = [A\, V_j^{(1)}, (A - s_j I)^{-1} V_j^{(2)}]$.
  3. Orthogonalize $\widetilde{V}_{j+1}$ with respect to $\mathbb{V}_1, \ldots, \mathbb{V}_j$ to get V_{j+1}, i.e.,
     for $i = 1, 2, \ldots, j$
     $H_{i,j} = (V_i)^T \widetilde{V}_{j+1}$;
     $\widetilde{V}_{j+1} = \widetilde{V}_{j+1} - V_i H_{i,j}$;
     end for
  4. $[V_{j+1},\ H_{j+1,j}] = QR(\widetilde{V}_{j+1})$.
  5. $\mathbb{V}_{j+1} = [\mathbb{V}_j,\ V_{j+1}]$.

  End For.

The matrix $\mathbb{V}_m = [V_1, V_2, \ldots, V_m]$ with $V_i \in \mathbb{R}^{n\times 2s}$ have their columns mutually orthogonal provided that none of the upper triangular matrices $H_{j+1,j}$ are rank deficient.

Hence, after m steps, Algorithm 6.1 builds an orthonormal basis $\mathbb{V}_m$ of the extended-rational Krylov subspace $\mathbb{K}_m^{er}(A, V)$ and an upper block Hessenberg matrix $\mathbb{H}_m$ whose non zero blocks are the $H_{i,j}$'s. Note that each submatrix $H_{i,j}$ $(1 \leq i \leq j \leq m)$ is of order $2s$.

Let $T_{i,j} = (V_i)^T A\, V_j \in \mathbb{R}^{2s\times 2s}$ and $\mathbb{T}_m = [T_{i,j}] \in \mathbb{R}^{2ms\times 2ms}$ be the restriction of

the matrix A to the extended Krylov subspace $\mathbb{K}_m^{er}(A, V)$, i.e.,

$$\mathbb{T}_m = \mathbb{V}_m^T \, A \, \mathbb{V}_m.$$

The matrix $\mathbb{T}_m$ is of great importance for the model reduction by the projection methods. When we manipulate big size models, the direct calculus of the matrix $\mathbb{T}_m$ has an elevated cost.

In the sequel, we propose a recursion to compute $\mathbb{T}_m$ from $\mathbb{H}_m$ without requiring matrix-vector products with A and extra inner products of long block-vectors.

First, we have to give some notations :

— Let $[V,\, (s_1 I - A)^{-1} V] = V_1 \Lambda$ be the QR decomposition of $[V,\, (s_1 I - A)^{-1} V]$ which can be written as

$$[V, (s_1 I - A)^{-1} V] = V_1 \Lambda = [V_1^{(1)},\, V_1^{(2)}] \begin{bmatrix} \Lambda_{1,1} & \Lambda_{1,2} \\ 0 & \Lambda_{2,2} \end{bmatrix}. \tag{6.1}$$

— For $k = 1, \ldots, m$, let's partition the lower triangular matrix $H_{k+1,k}$ under the form

$$H_{k+1,k} = \begin{bmatrix} H_{k+1,k}^{(1,1)} & H_{k+1,k}^{(1,2)} \\ 0 & H_{k+1,k}^{(2,2)} \end{bmatrix}.$$

In following, we will provide a technique to compute $\overline{\mathbb{T}}_m$ directly from the columns of the upper block Hessenberg matrix $\overline{\mathbb{H}}_m$ obtained from Algorithm 6.1.

Proposition 6.1 *Let $\overline{\mathbb{T}}_m$ and $\overline{\mathbb{H}}_m$ be the upper block Hessenberg matrices defined earlier. Then we have the following relations :*
for $k = 1, \ldots, m$

$$\overline{\mathbb{T}}_m \, \widetilde{e}_{2k-1} = \begin{bmatrix} \overline{\mathbb{H}}_k \\ 0_{2(m-k)s \times 2ks} \end{bmatrix} = \overline{\mathbb{H}}_m \, \widetilde{e}_{2k-1}, \tag{6.2}$$

$$\overline{\mathbb{T}}_m \, \widetilde{e}_2 = \left(s_1 \begin{bmatrix} I_{2s} \\ 0_{2ms \times 2s} \end{bmatrix} \begin{bmatrix} \Lambda_{1,2} \\ \Lambda_{2,2} \end{bmatrix} - \overline{\mathbb{T}}_m \, \widetilde{e}_1 \, \Lambda_{1,2} - \widetilde{e}_1 \, \Lambda_{1,1} \right) (\Lambda_{2,2})^{-1}, \tag{6.3}$$

and

$$\mathbb{T}_{m+1} \, \widetilde{e}_{2k+2} = \left(s_k \overline{\mathbb{H}}_m \, \widetilde{e}_{2k} \;-\; \begin{bmatrix} \overline{\mathbb{T}}_k \\ 0_{2(m-k)s \times 2ks} \end{bmatrix} \mathbb{H}_k \, \widetilde{e}_{2k} \right.$$
$$\left. -\; \mathbb{T}_{m+1} \, \widetilde{e}_{2k+1} \, H_{k+1,k}^{(1,2)} - \widetilde{e}_{2k} \right) (H_{k+1,k}^{(2,2)})^{-1}, \tag{6.4}$$

where $\widetilde{e}_i = e_i \otimes I_s$ and the e_i's are the vectors of the canonical basis.

Proof First, we notice that for $k \geq 1$, $V_k = [V_k^{(1)}, V_k^{(2)}] \in \mathbb{R}^{n \times 2s}$ and from Algorithm 6.1, we have

$$\widehat{V}_{k+1} = [A\, V_k^{(1)}, (s_k I - A)^{-1}\, V_k^{(2)}] - \mathbb{V}_k\, \mathbb{H}_k\, [\widetilde{e}_{2k-1}, \widetilde{e}_{2k}], \tag{6.5}$$

and

$$V_{k+1}\, H_{k+1,k} = \widehat{V}_{k+1}. \tag{6.6}$$

Using the relations (6.5) and (6.6), we obtain

$$\begin{aligned} A\, V_k^{(1)} = \widehat{V}_{k+1}\, \widetilde{e}_1 + \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k-1} &= V_{k+1}\, H_{k+1,k}\, \widetilde{e}_1 + \mathbb{V}_k\, \mathbb{H}_k\, \widetilde{e}_{2k-1} \\ &= \mathbb{V}_{k+1} \overline{\mathbb{H}}_k\, \widetilde{e}_{2k-1}. \end{aligned}$$

Now, multiplying on the left by $\mathbb{V}_{m+1}^T$, we get

$$\mathbb{V}_{m+1}^T\, A\, V_k^{(1)} = \mathbb{V}_{m+1}^T\, \mathbb{V}_{k+1}\, \overline{\mathbb{H}}_k\, \widetilde{e}_{2k-1},$$

hence

$$\mathbb{V}_{m+1}^T\, A\, \mathbb{V}_m\, \widetilde{e}_{2k-1} = \begin{bmatrix} I_{2(k+1)s} \\ 0_{2(m-k)s \times 2(k+1)s} \end{bmatrix} \overline{\mathbb{H}}_k\, \widetilde{e}_{2k-1},$$

and so

$$\overline{\mathbb{T}}_m\, \widetilde{e}_{2k-1} = \begin{bmatrix} \overline{\mathbb{H}}_k \\ 0_{2(m-k)s \times 2ks} \end{bmatrix} = \overline{\mathbb{H}}_m\, \widetilde{e}_{2k-1},$$

which gives the relation (6.2).

To prove (6.3), we start from the QR decomposition of $[V,\ (s_1 I - A)^{-1} V]$ given in (6.1)

$$[V,\ (s_1 I - A)^{-1} V] = [V_1^{(1)}\, \Lambda_{1,1}, V_1^{(1)}\, \Lambda_{1,2} + V_1^{(2)}\, \Lambda_{2,2}].$$

Then

$$(s_1 I - A)^{-1} V = V_1^{(1)}\, \Lambda_{1,2} + V_1^{(2)}\, \Lambda_{2,2}.$$

Now, multiplying on the left by $(s_1 I - A)$, we get

$$V = s_1 [V_1^{(1)},\ V_1^{(2)}] \begin{bmatrix} \Lambda_{1,2} \\ \Lambda_{2,2} \end{bmatrix} - A V_1^{(1)}\, \Lambda_{1,2} - A V_1^{(2)}\, \Lambda_{2,2}.$$

Since $V = V_1^{(1)}\, \Lambda_{1,1}$, we have

$$A V_1^{(2)}\, \Lambda_{2,2} = s_1 V_1 \begin{bmatrix} \Lambda_{1,2} \\ \Lambda_{2,2} \end{bmatrix} - A V_1^{(1)}\, \Lambda_{1,2} - V_1^{(1)}\, \Lambda_{1,1}.$$

Hence, if $\Lambda_{2,2}$ is nonsingular, we get (6.3) by pre-multiplying the above equality on the left by $\mathbb{V}_{m+1}^T$ and using the facts that $\mathbb{V}_{m+1}^T V_1^{(1)} = (e_1 \otimes I_s) = \widetilde{e}_1$ and $\mathbb{V}_{m+1}^T A V_1^{(i)} = \overline{\mathbb{T}}_m (e_i \otimes I_s) = \overline{\mathbb{T}}_m \widetilde{e}_i$ for $i = 1, 2$.
Now, in order to prove (6.4), we use the relations (6.5) and (6.6) and we get

$$\begin{aligned}(s_k I - A)^{-1} V_k^{(2)} &= \widehat{V}_{k+1} \widetilde{e}_2 + \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k} \\ &= V_{k+1} H_{k+1,k} \widetilde{e}_2 + \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k}.\end{aligned}$$

Multiplying on the left by $(s_k I - A)$, we have

$$V_k^{(2)} = s_k \left[V_{k+1} H_{k+1,k} \widetilde{e}_2 + \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k}\right] - A V_{k+1} H_{k+1,k} \widetilde{e}_2 - A \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k}.$$

Since

$$A V_{k+1} H_{k+1,k} \widetilde{e}_2 = A V_{k+1}^{(1)} H_{k+1,k}^{(1,2)} + A V_{k+1}^{(2)} H_{k+1,k}^{(2,2)},$$

we deduce the following relations

$$\begin{aligned}A V_{k+1}^{(2)} H_{k+1,k}^{(2,2)} &= s_k \left[V_{k+1} H_{k+1,k} \widetilde{e}_2 + \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k}\right] - A \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k} \\ &- A V_{k+1}^{(1)} H_{k+1,k}^{(1,2)} - V_k^{(2)} \\ &= s_k \mathbb{V}_{k+1} \overline{\mathbb{H}}_k \widetilde{e}_{2k} - A \mathbb{V}_k \mathbb{H}_k \widetilde{e}_{2k} - A V_{k+1}^{(1)} H_{k+1,k}^{(1,2)} - V_k^{(2)}.\end{aligned}$$

Finally, multiplying on the left by $\mathbb{V}_{m+1}^T$, we get

$$\begin{aligned}\mathbb{T}_{m+1} \widetilde{e}_{2k+2} &= \left(s_k \begin{bmatrix} I_{2(k+1)s} \\ 0_{2(m-k)s \times 2(k+1)s} \end{bmatrix} \overline{\mathbb{H}}_k \widetilde{e}_{2k} - \begin{bmatrix} \overline{\mathbb{T}}_k \\ 0_{2(m-k)s \times 2ks} \end{bmatrix} \mathbb{H}_k \widetilde{e}_{2k} \right. \\ &- \left. \mathbb{T}_{m+1} \widetilde{e}_{2k+1} H_{k+1,k}^{(1,2)} - \widetilde{e}_{2k} \right) (H_{k+1,k}^{(2,2)})^{-1} \\ &= \left(s_k \overline{\mathbb{H}}_m \widetilde{e}_{2k} - \begin{bmatrix} \overline{\mathbb{T}}_k \\ 0_{2(m-k)s \times 2ks} \end{bmatrix} \mathbb{H}_k \widetilde{e}_{2k} - \mathbb{T}_{m+1} \widetilde{e}_{2k+1} H_{k+1,k}^{(1,2)} - \widetilde{e}_{2k} \right) \\ &\times (H_{k+1,k}^{(2,2)})^{-1}.\end{aligned}$$

Which ends the proof.

Remark 6.1 *Note that in the previous proposition, the matrix $\mathbb{T}_m$ has a block Hessenberg form.*

As for any Krylov subspace method, a set of identities known as the Arnoldi relations are satisfied and are used to compute error bounds, residuals, stop tests and to perform the perturbation analysis. In the case of rational Krylov subspaces, some

relations have been established in the literature; see [28, 64, 76]. However those identities are much more complex in the rational case when compared to the standard Arnoldi equations.

Proposition 6.2 *Assume that m steps of Algorithm 6.1 have been run and let $\overline{\mathbb{T}}_m = \mathbb{V}_{m+1}^T A \mathbb{V}_m$. Then we have the following relations*

$$\begin{aligned} A\,\mathbb{V}_m &= \mathbb{V}_{m+1}\,\overline{\mathbb{T}}_m \\ &= \mathbb{V}_m\,\mathbb{T}_m + V_{m+1}\,T_{m+1,m}\,\mathbb{E}_m^T. \end{aligned} \tag{6.7}$$

Proof In order to prove the above proposition, we have first to show that

$$A\,\mathbb{K}_m^{er}(A,V) \subseteq \mathbb{K}_{m+1}^{er}(A,V).$$

Set $V = [V^{(1)}, \ldots, V^{(s)}] \in \mathbb{R}^{n\times s}$, where $V^{(k)}$ denotes the k-th column of V.
For $j = 1, \ldots, m$ and $k = 1, \ldots, s$, we have

$$\begin{aligned} A \prod_{i=1}^{j}(A - s_i I)^{-1}\,V^{(k)} &= \prod_{i=1}^{j}(A - s_i I)^{-1}\,A\,V^{(k)} \\ &= \prod_{i=1}^{j}(A - s_i I)^{-1}\big[(A - s_j I)\,V^{(k)} + s_j\,V\big] \\ &= \prod_{i=1}^{j-1}(A - s_i I)^{-1}V^{(k)} + s_j \prod_{i=1}^{j}(A - s_i I)^{-1}V^{(k)} \\ &\in \mathbb{K}_{m+1}^{er}(A,V). \end{aligned}$$

Thus $A\prod_{i=1}^{j}(A - s_i I)^{-1}\,V^{(k)} \in \mathbb{K}_{m+1}^{er}(A,V)$. It is easy to show that for $j = 1, \ldots, m$ and $k = 1, \ldots, s$, $A(A^j V^{(k)}) = A^{j+1}V^{(k)} \in \mathbb{K}_{m+1}^{er}(A,V)$.
Hence, there exists a matrix T such that

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

Multiplying by $\mathbb{V}_{m+1}$ on the left, we obtain $T = \overline{\mathbb{T}}_m$. Since $\overline{\mathbb{T}}_m$ is a Hessenberg matrix, it can be decomposed as follows

$$A\,\mathbb{V}_m = \mathbb{V}_m\,\mathbb{T}_m + V_{m+1}\,T_{m+1,m}\,\mathbb{E}_m^T.$$

Remark 6.2 *(Choice of shifts)*
It is known that the rational Krylov subspace methods requires some parameters (shifts) to build the subspace, and a good selection of this shifts is crucial for the quality of the approximation. The choice of these parameters is still an open question for the extended-rational Krylov methods. We mention that in this work we will preserve the same choice used in the second chapter.

6.2 Model reduction, transfer function

The main purpose of this section is the computation of a reduced model order of a linear dynamical system by using the extended-rational Krylov subspace techniques.

We recall the linear time-invariant (LTI) multi-input and multi-output (MIMO) dynamical system

$$\begin{cases} \dot{x}(t) &= A\,x(t) + B\,u(t) \\ y(t) &= C\,x(t). \end{cases} \tag{6.8}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t),\ y(t) \in \mathbb{R}^r$ are the input and the output vectors of the system, respectively. The matrices $B,\ C^T$ are in $\mathbb{R}^{n\times r}$ and $A \in \mathbb{R}^{n\times n}$ is assumed to be large and sparse. The transfer function associated to the above system is given by

$$F(s) = C\,(s\,I_n - A)^{-1}\,B. \tag{6.9}$$

We recall that most of the model order reduction techniques, for example the moment-matching approaches, are based on the approximation of this transfer function ; for more details, see [10, 32, 38] and the references therein. If the number of state variables is very large, it would be very difficult to use the full system for simulation or run-on-time control. So it is necessary to look for lower order models that approximate the behavior of the original models.

By applying the Galerkin projection method on the extended-rational Krylov $\mathbb{K}^{er}_m(A, V)$, we verify that the original transfer function F can be approximated by

$$F_m(s) = C_m\,(s\,I_{2ms} - \mathbb{T}_m)^{-1}\,B_m,$$

where $\mathbb{T}_m = \mathbb{V}_m^T\,A\,\mathbb{V}_m$, $C_m = C\,\mathbb{V}_m$ and $B_m = \mathbb{V}_m^T\,B$.

6.3 Numerical examples

In this section, we give some experimental results to show the effectiveness of the proposed approache. All the experiments were performed on a computer of Intel Core i5 of 1.3GHz and 8GB of RAM. The algorithms were coded in Matlab R2010a. We used some known benchmark models listed in Table 6.1.

TABLE 6.1 – Test matrices

Matrix A	size n	$\|A\|_F$	$cond(A)$
`CDplayer`	120	2.3095e+05	1.8149e+04
`Flow`	9669	2.5438e+04	1.6193e+07

The matrices for the benchmark problems `CDplayer` was obtained from NICONET [68] while the matrices for the `Flow` model was obtained from the discretization of a 2D convective thermal flow problem (flow meter model v0.5) from the Oberwolfach collection [1].

To put the contribution of the extended-rational Krylov method in perspective, we will compare it with two methods, the first is a comparison with the rational Krylov subspace method and the second with the extended Krylov subspace method.
We mention that for the first comparison the same approach was used for the choice of shifts for both methods.
To simplify the notation, we will use the following abbreviations, ERKM for the extended-rational Krylov method, RKM for the Rational Krylov method and EKM for the Extended Krylov method.

Experiment 1. In the first experiment, we considered the `CDplayer` model. Although the matrices of this model have small size they are usually considered as benchmark example. For B and C, we used the matrices of the model. Their size is $n \times 2$. In Figure 6.1, we plotted the error norm $\|F(i\omega) - F_m(i\omega)\|_2$ for the rational Krylov and extended-rational Krylov methods for $\omega \in [10^{-1}, 10^5]$ with $m = 10$ (left) and $m = 20$ (right).

Experiment 2. The second experiment is a comparison between ERKM and EKM methods both used for the Flow model. The matrices B and $C^\top$ were random matrices with entries uniformly distributed in $[0, 1]$ and they are of size $n \times 3$. In Figure 6.2, we plotted the error norm $\|F(i\omega) - F_m(i\omega)\|_2$ for the both methods for $\omega \in [10^{-1}, 10^5]$ with $m = 15$ (left) and $m = 30$ (right).

1. Oberwolfach model reduction benchmark collection, 2003. http ://www.imtek.de/simulation/benchmark

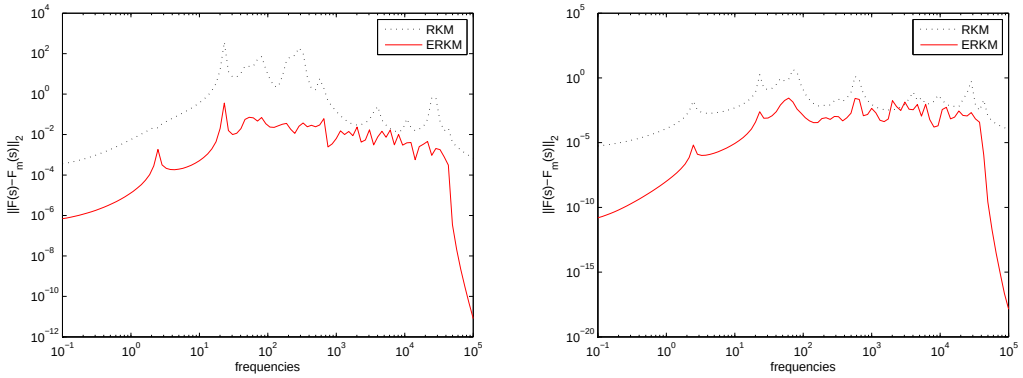

FIGURE 6.1 – The `CDplayer` model. The norms of the errors for the rational Krylov (dashed) and extended-rational Krylov (solid) methods for $\omega \in [10^{-1}, 10^{5}]$ with $m = 10$ (left) and $m = 20$ (right).

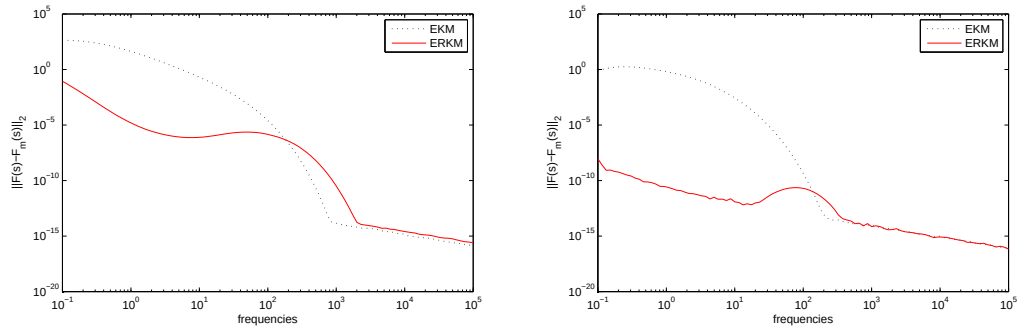

FIGURE 6.2 – The `Flow` model. The norms of the errors for the extended Krylov (dashed) and extended-rational Krylov (solid) methods for $\omega \in [10^{-1}, 10^{5}]$ with $m = 15$ (left) and $m = 30$ (right).

6.4 Conclusion

In the present chapter, we considered a new projection method for model reduction in large scale linear dynamical systems. The proposed method is a Krylov subspace type method based on the new Krylov subspace : the extended-rational Krylov subspace. We proposed a new algorithm to build an orthonormal basis of this subspace. New simple Arnoldi-like relations in the extended-rational case were also proposed. The proposed numerical results on some Benchmark models, show that the extended-rational Krylov method is usefull in the model reduction.

Conclusion and perspectives

In this thesis, we are interested to projection methods on rational Krylov subspaces for model reduction in large scale dynamical systems. We have developed the block and global Arnoldi methods based on the rational block and the global Arnoldi algorithms. A modified versions of the rational Arnoldi algorithms were also proposed and it was used (for the global case) for the two-side projection in order to improve the approximation of the reduced model to the original one. New algebraic relations were also given generalizing classical Arnoldi-like relations and used to establish upper bounds on the error's norm of the difference between the initial transfer function and its approximation.

A crucial question for projection methods based on rational Krylov subspaces is the choice of shifts. We proposed some shifts selection techniques needed in the proposed rational algorithms.

We are also interested to the extended block and global Arnoldi methods. Theoretically, the extended Krylov subspace is a particular case of the rational Krylov subspace obtained by alternating shifts between zero and infinity. We gave some new algebraic properties and used them for approaching the first moments and Markov parameters in moment matching methods for model reduction techniques.

Afterwards, we considered the balanced truncation method. To this purpose we applied the rational block Arnoldi algorithm to compute low rank approximate solutions to large Sylvester (or Lyapunov) and continuous-time algebraic Riccati equations having low rank right-hand sides.

Eventually, we proposed a new method based on a new subspace called the extended-rational Krylov subspace. We introduced the extended-rational Krylov method which will be used for model reduction in large-scale dynamical systems. Among the advantages of this technique, a projection method based on a new Kry-

lov subspace richer than the known ones and some simple algebraic relations independant from the shifts choice.

With these works realised in the area of model reduction, there still be other issues to treat.

First, despite the choice of shifts that have been proposed in this thesis and their effectiveness in the numerical tests, we are not yet satisfied and we seek to improve these choices to approach the original model on the entire frequency range in a more effective manner.

Second, a work related to the last chapter is envisaged to do. In this work we will apply the extended-rational Krylov method to approach the solutions of Sylvester and Riccati equations and use the techniques of balanced truncation based on the latter method to produce a reduced model that approaches the initial one.

In the end, we will try to apply the techniques described in this thesis to more general matrix functions, such as the exponential function.

Bibliographie

[1] O. ABIDI, M. HACHED AND K. JBILOU, *Adaptive rational block Arnoldi methods for model reductions in large-scale MIMO dynamical systems*, New Trend. Math., 4 (2016) 227-239.

[2] O. ABIDI, M. HACHED AND K. JBILOU, *Global rational Arnoldi method for model reduction*, soumis à J. Comput. Appl. Math.

[3] O. ABIDI, M. HEYOUNI AND K. JBILOU, *On some properties of the extended block and global Arnoldi methods with applications to model reduction*, Numer. Algo., (2016) doi :10.1007/s11075-016-0207-7.

[4] O. ABIDI AND K. JBILOU, *Balanced truncation-rational Krylov methods for model reduction in large scale dynamical systems*, Comp. Appl. Math., (2016) doi :10.1007/s40314-016-0359-z.

[5] H. ABOU-KANDIL, G. FREILING, V. IONESCU, G. JANK, *Matrix Riccati equations in control and sytems theory*, in Sys. & Contr. Foun. & Appl., Birkhauser, (2003).

[6] B. O. ANDERSON, J.B. MOORE, *Linear optimal control*, Prentice-Hall, Englewood Cliffs, NJ, (1971).

[7] A. C. ANTOULAS, Approximation of Large-Scale Dynamical Systems, SIAM (2005).

[8] M. ANTUNES, O. LEITE, B. DELINCHANT, J. M. GUICHON AND J. VASCONCELLOS, *Comparaison des méthodes de réduction d'ordre POD et Krylov, application à la méthode PEEC*, Symposium de Genie Electrique, v1 (2016).

[9] Z. BAI, *Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems*, Appl. Numer. Math., 43 (2002) 9-44.

[10] Z. BAI, Q. YE, *Error estimation of the Padé approximation of transfer functions via the Lanczos process*, Elect. Trans. Numer. Anal., 7 (1998) 1-17.

[11] R. H. BARTELS AND G.W. STEWART, *Solution of the matrix equation* $AX + XB = C$, Comm. ACM., 15 (1972) 820-826.

[12] P. BENNER, R.C. LI, N. TRUHAR, *On the ADI method for Sylvester equations*, J. Comput. Appl. Math., 233 (2009) 1035-1045.

[13] P. BENNER, J. LI, AND T. PENZL *Numerical solution of large Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems*, Numer. Lin. Alg. Appl., 15 (2008) 755-777.

[14] G. BERKOOZ, P. HOLMES, AND J. LUMLEY, *The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows*, Annual Review of Fluid Mechanics 25 (1993) 539–575.

[15] S. BITTANTI, A. LAUB, J.C. WILLEMS, *The Riccati equation*, Springer-Verlag, Berlin, (1991).

[16] T. BONIN, H. FASSBENDER, A. SOPPA, M. ZÄH, *A fully adaptive rational global Arnoldi method for the model-order reduction of second-order MIMO systems with proportional damping*, preprint, (2014).

[17] A. BOUHAMIDI, M. HACHED AND K. JBILOU, *A preconditioned block Arnoldi for large Sylvester matrix equations*, Numer. Lin. Alg. Appl., 2 (2013) 208-219.

[18] R. BOUYOULI, K. JBILOU, R. SADAKA, H. SADOK, *Convergence properties of some block Krylov subspace methods for multiple linear systems*, J. Comput. Appl. Math., 196 (2006) 498-511.

[19] C. BREZINSKI, *Computational Aspects of Linear Control*, Kluwer, Dordrecht, (2002).

[20] A. CHATTERJEE, *An introduction to the proper orthogonal decomposition*, Current Science 78, 7 (2000) 808–817.

[21] C. CHIA-CHI AND L. MING-HONG AND F. WU-SHIUNG, *Model-order reductions for MIMO systems using global Krylov subspace methods*, Math. Comput. Simul., 79 (2008) 1153-1164.

[22] L. CORDIER, M. BERGMANN, *Proper Orthogonal Decompositi*, Lecture series 2002-04 on post-processing of experimental and numerical data, Von Kármán Institute for Fluid Dynamics, (2002).

[23] B.N. DATTA, *Numerical Methods for Linear Control Systems Design and Analysis*, Elsevier Academic Press, (2003).

[24] B. N. DATTA, *Large-Scale Matrix computations in Control*, Appl. Numer. Math., 30 (1999) 53-63.

[25] B. N. DATTA, *Krylov Subspace Methods for Large-Scale Matrix Problems in Control*, Future Gener. Comput. Syst., 19 (2003) 1253-1263.

[26] V. DRUSKIN, L. KNIZHNERMAN, *Extended Krylov subspaces : approximation of the matrix square root and related functions*, SIAM J. Matrix Anal. Appl., 3 (1998) 755-771.

[27] V. DRUSKIN, C. LIEBERMAN, M. ZASLAVSKY, *On adaptive choice of shifts in rational Krylov subspace reduction of evolutionary problems*, SIAM J. Sci. Comput., 32 (2010) 2485-2496.

[28] V. DRUSKIN, V. SIMONCINI, *Adaptive rational Krylov subspaces for large-scale dynamical systems*, Systems Control Lett., 60 (2011) 546-560.

[29] B.F. FARELL AND P.J IOANNOU, *State estimation using a reduced kalman filter*, J. Atmosph. Sci., 58 (2001) 3666-3680.

[30] P. FELDMANN, R. W. FREUND, *Efficient Linear Circuit Analysis by Padé Approximation via The Lanczos process*, IEEE Trans. CAD Integ. Circ. Sys., 14 (1995) 639-649.

[31] L. FORTUNA, G. NUNNARI AND A. GALLO , Model Order Reduction Techniques with Applications in Electrical Engineering, Springer-Verlag, London, (1992).

[32] M. FRANGOS, I. M. JAIMOUKHA, *Adaptive rational interpolation : Arnoldi and Lanczos-like equations*, Eur. J. Cont, 14 (2008) 342-354.

[33] R. FREUND, *Krylov-subspace methods for reduced-order modeling in circuit simulation*, J. Comput. Appl. Math., 123 (2000) 395-421.

[34] K. GALLIVAN, E. GRIMME, P. VAN DOOREN, *A rational Lanczos algorithm for model reduction*, Numer. Algo., 12 (1996) 33-63.

[35] K. GLOVER, *All optimal Hankel-norm approximations of linear multivariable systems and their L-infinity error bounds*. Inter. J. Cont., 39 (1984) 1115-1193.

[36] K. GLOVER, D. J. N. LIMEBEER, J. C. DOYLE, E. M. KASENALLY, M. G. SAFONOV, *A characterization of all solutions to the four block general distance problem*, SIAM J. Control Optim., 29 (1991) 283-324.

[37] E. GRIMME, *Krylov projection methods for model reduction* , Ph.D. Thesis, The University of Illinois at Urbana-Champaign (1997).

[38] E. GRIMME, D. SORENSEN AND P. VAN DOOREN, *Model reduction of state space systems via an implicitly restarted Lanczos method*, Numer. Algo., 12 (1996) 1-32.

[39] A. EL GUENNOUNI, K. JBILOU AND A.J. RIQUET, *Block Krylov subspace methods for solving large Sylvester equations*, Numer. Algo., 29 (2002) 75-96.

[40] S. GUGERCIN AND A.C. ANTOULAS, *A survey of model reduction by balanced truncation and some new results*, Int. J. Control, 77 (2004) 748-766.

[41] S. GUGERCIN, A.C. ANTOULAS, C. BEATTIE, *$\mathcal{H}_2$ model reduction for large-scale linear dynamical systems*, SIAM J. Matrix Anal. Appl., 30 (2008) 609-638.

[42] S. Gugercin, and A. C. Antoulas, Model reduction of large scale systems by least squares, Lin. Alg. Appl., 415 (2006) 290-321.

[43] S. GÜTTEL, *Rational Krylov Methods for Operator Functions*, Ph.D. Thesis, TU Bergakademie Freiberg (2010).

[44] S. GÜTTEL, *Rational Krylov approximation of matrix functions : Numerical methods and optimal pole selection*, GAMM-Mitteilungen, 36 (2013) 8-31.

[45] K. HENRIK, A. OLSSON AND A. RUHE, *Rational Krylov for eigenvalue computation and model order reduction*, BIT, 46 (2006) 99-111.

[46] M. HEYOUNI, K. JBILOU, *An extended block Arnoldi algorithm for large-scale solutions of the continuous-time algebraic Riccati equation*, Elect. Trans. Num. Anal., 33 (2009) 53-62.

[47] M. HEYOUNI, *Extended Arnoldi methods for large Sylvester matrix equations*, App. Num. Math., 60 (2010) 1171-1182.

[48] M. HEYOUNI, K. JBILOU, *Matrix Krylov subspace methods for large scale model reduction problems*, App. Math. Comput., 181 (2006) 1215-1228.

[49] M. HEYOUNI, K. JBILOU, A. MESSAOUDI, AND K. TABAA, *Model reduction in large scale MIMO dynamical systems via the block Lanczos method*, Comp. Appl. Math., 27 (2008) 211-236.

[50] N. J. HIGHAM, Functions of Matrices : Theory and Computation, SIAM (2008).

[51] D. Y. HU AND L. REICHEL, *Krylov subspace methods for the Sylvester equation*, Lin. Alg. Appl., 174 (1992) 283-314.

[52] I.M. JAIMOUKHA AND E.M. KASENALLY, *Krylov subspace methods for solving large Lyapunov equations*, SIAM J. Matrix Anal. Appl., 31 (1994) 227-251.

[53] C. JAGELS, L. REICHEL, *The extended Krylov subspace method and orthogonal Laurent polynomials*, Lin. Alg. Appl., 431 (2009) 441-458.

[54] K. JBILOU, A. MESSAOUDI AND H. SADOK, *Global FOM and GMRES algorithms for matrix equations*, Appl. Numer. Math., 31 (1999) 49-63.

[55] K. JBILOU, *ADI preconditioned Krylov methods for large Lyapunov matrix equations*, Lin. Alg. Appl., 432 (2010) 2473-2485.

[56] K. JBILOU, *Low rank approximate solutions to large Sylvester matrix equations* Appl. Math. Comput., 177 (2006) 365-376.

[57] K. JBILOU, *Block Krylov subspace methods for large continuous-time algebraic Riccati equations*, Numer. Algo., 34 (2003) 339-353.

[58] K. JBILOU, *An Arnoldi based method for large algebraic Riccati equations*, Appl. Math. Lett., 19 (2006) 437-444.

[59] K. JBILOU, *A survey of Krylov-based methods for model reduction in large-scale MIMO dynamical systems*, Appl. Comput. Math., 15 (2016) 24-42.

[60] K. JBILOU AND A.J. RIQUET, *Projection methods for large Lyapunov matrix equations*, Lin. Alg. Appl., 415 (2006) 344-358.

[61] R E . KALMAN, Contributions to the theory of optimal control, Bol. Soc. Mat. Mex., 5 (1960) 102-119.

[62] L. KNIZHNERMAN AND V. SIMONCINI, *Convergence analysis of the extended Krylov subspace method for the Lyapunov equation*, Numer. Math., 118 (2011) 567-586.

[63] D.L. KLEINMAN, *On an iterative technique for Riccati equation computations*, IEEC Trans. Autom. Contr., 13 (1968) 114-115.

[64] 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 (2009) 953-971.

[65] A.J. LAUB, *A Schur method for solving algebraic Riccati equations*, IEEE Trans. Automat. control, Vol., AC-24 (1979) 913-921.

[66] T. MACH, M.S. PRANIC AND R. VANDEBRIL, *Computing approximate extended Krylov subspaces without explicit inversion*, Electron. Trans. Numer. Anal., 40 (2013) 414–435.

[67] T. MACH, M.S. PRANIC AND R. VANDEBRIL, *Computing approximate (block) rational Krylov subspaces without explicit inversion with extensions to symmetric matrices*, Electron. Trans. Numer. Anal., 43 (2014) 100–124.

[68] V. MEHRMANN AND T. PENZL, *Benchmark collections in SLICOT*. Technical Report SLWN1998- 5, SLICOT Working Note, ESAT, KU Leuven, Belgium, 1998. Available under http ://www.win.tue.nl/niconet/NIC2/reports.html.

[69] V. MEHRMANN, T. STYKEL, *Balanced truncation model reduction for large-scale systems in descriptor form*, in Dimension Reduction of Large-Scale Systems Series Lecture Notes in Computational Science and Engineering 45 (2005) 83-115.

[70] B. C. MOORE, *Principal component analysis in linear systems : controllability, observability and model reduction*, IEEE Trans. Automatic Contr., AC-26 (1981) 17-32.

[71] C. T. MULLIS AND R. A. ROBERTS, IEEE Trans. Acoust. Speech Signal Process., 24, 1976.

[72] T. PENZL, *LYAPACK A MATLAB toolbox for Large Lyapunov and Riccati Equations, Model Reduction Problem, and Linear-quadratic Optimal Control Problems*, http ://www.tu-chemnitz.de/sfb393/lyapack

[73] T. PENZL, *A cyclic low-rank Smith method for large sparse Lyapunov equations*, SIAM J. Sci. Comput., 21 (1999) 1064-8275.

[74] A. RUHE, *Rational Krylov sequence methods for eigenvalue computation*, Lin. Alg. Appl., 58 (1984) 391-405.

[75] A. RUHE, *Rational Krylov algorithms for nonsymmetric eigenvalue problems. II. Matrix pairs*, Lin. Algeb. Appl., 197/198 (1994) 283-295.

[76] A. RUHE, *The rational Krylov algorithm for nonsymmetric eigenvalue problems. III. Complex shifts for real matrices*, BIT, 34 (1994) 165-176.

[77] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, the PWS Publishing Company, (1996).

[78] J. SAAK, *Efficient Numerical Solution of Large Scale Algebraic Matrix Equations in PDE Control and Model Order Reduction* , Ph.D. Thesis, Chemnitz University of Technology (2009).

[79] W. SCHILDERS, *Introduction to model order reduction*, Math. Ind. Springer, Berlin (2008).

[80] Y. SHAMASH, *Stable reduced-order models using Padé type approximations*, IEEE. Trans. Automatic Control, AC-19 (1974) 615-616.

[81] V. SIMONCINI, *A new iterative method for solving large-scale Lyapunov matrix equations*, SIAM J. Sci. Comp., 29 (2007) 1268-1288.

[82] P. VAN DOOREN, K.A. GALLIVAN, P. A. ABSIL, *H_2-optimal model reduction of MIMO systems*, Appl. Math. Lett., 21 (2008) 1267-1273.

[83] P. VAN DOOREN, *Gramian based model reduction of large-scale dynamical systems*, in Numerical Analysis, Chapman and Hall, CRC Press London, (2000) 231-247.

[84] E.L. WACHSPRESS, *Iterative Solution of Elliptic Systems*, Prentice-Hall, Inc., Englewood Cliffs, N.J. (1966).

[85] E.L. WACHSPRESS, *The ADI minimax problem for complex spectra*, Academic Press, Boston, MA in Iterative Methods for Large Linear Systems, (1990) 251-271.

[86] K. WILLCOX, J. PERAIRE, *Balanced model reduction via the proper orthogonal*, decomposition, AIAA Journal 40 (2002) 2323–2330.