
KTH Computer Science
and Communication

# Model Order Reduction with Rational Krylov Methods

K. HENRIK A. OLSSON

Doctoral Thesis in Numerical Analysis
Stockholm, Sweden 2005

TRITA-NA-0522
ISSN 0348-2952
ISRN KTH/NA/R--05/22--SE
ISBN 91-7178-126-9

KTH Numerisk analys och datalogi
SE-100 44 Stockholm
SWEDEN

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie doktorsexamen måndagen den 26 september 2005 klockan 10.15 i Salongen, KTHB, Osquars backe 31, Kungl Tekniska högskolan, Stockholm.



Tryck: Universitetsservice US AB

**Abstract**

Rational Krylov methods for model order reduction are studied. A dual rational Arnoldi method for model order reduction and a rational Krylov method for model order reduction and eigenvalue computation have been implemented. It is shown how to deflate redundant or unwanted vectors and how to obtain moment matching. Both methods are designed for generalised state space systems—the former for multiple-input-multiple-output (MIMO) systems from finite element discretisations and the latter for single-input-single-output (SISO) systems—and applied to relevant test problems. The dual rational Arnoldi method is designed for generating real reduced order systems using complex shift points and stabilising a system that happens to be unstable. For the rational Krylov method, a forward error in the recursion and an estimate of the error in the approximation of the transfer function are studied.

A stability analysis of a heat exchanger model is made. The model is a nonlinear partial differential-algebraic equation (PDAE). Its well-posedness and how to prescribe boundary data is investigated through analysis of a linearised PDAE and numerical experiments on a nonlinear DAE. Four methods for generating reduced order models are applied to the nonlinear DAE and compared: a Krylov based moment matching method, balanced truncation, Galerkin projection onto a proper orthogonal decomposition (POD) basis, and a lumping method.

**Sammanfattning**

Rationell Krylovmetoder för modellreduktion studeras. En dual rationell Arnoldimetod för modellreduktion och en rationell Krylovmetod för modellreduktion och egenvärdesberäkning har implementerats. Det visas hur överflödiga och oönskade vektorer tas bort och hur momentmatchning erhålls. Båda metoderna är konstruerade för generaliserade tillståndsrumssystem — den förra för system med flera in- och utsignaler från finita elementdiskretiseringar och den senare för system med bara en in- respektive utsignal — och tillämpade på relevanta testproblem. Den duala rationella Arnoldimetoden är gjord för att generera reella reducerade modeller utifrån komplexa skiftpunkter och stabilisera en modell som har råkat bli instabil. För den rationella Krylovmetoden studeras ett framåtfel i rekursionen och en skattning av felet i approximationen av överföringsfunktionen.

En stabilitetsanalys av en värmeväxlarmodell utförs. Modellen är en ickelinjär partiell differential-algebraisk ekvation (PDAE). Dess rättställdhet och hur randvärden skall anges undersöks genom analys av en linjäriserad PDAE och numeriska experiment på en ickelinjär DAE. Fyra metoder för generering av reducerade modeller används på den ickelinjära DAE:n och jämförs: en Krylovmetod för momentmatchning, balanserad trunkering, Galerkinprojektion på en POD-bas och en lumpningsmetod.

Keywords: Model order reduction, dual rational Arnoldi, rational Krylov, moment matching, eigenvalue computation, stability analysis, heat exchanger model

AMS 2000 subject classification: 65F15, 65F30, 65F50, 93A30, 65M60, 35L80, 76T10

## Acknowledgements

I thank:

my supervisor and co-author Axel Ruhe for supervising my work and for reading parts of this manuscript at several stages,

my co-authors Michael Hanke and Magnus Strömgren for working with me,

my family and friends for everything.

I also acknowledge that without modern technology, referring mainly to computers but also to wind instruments and running shoes, this thesis would have been something completely different.

## Preface

This thesis consists of an introduction and the following four parts:

Part I: K. Henrik A. Olsson. *Model Order Reduction in FEMLAB by Dual Rational Arnoldi*. Licentiate thesis, Chalmers University of Technology, 2002. Part of this work has been presented by the author at the workshop Model Reduction Problems and Matrix Methods, Banff, 2004.

Part II: K. Henrik A. Olsson and Axel Ruhe. *Rational Krylov for Model Order Reduction and Eigenvalue Computation*. Technical report, TRITA-NA-0520, Royal Institute of Technology, 2005. The first author is the main author of this paper, he also performed the computations and wrote most of the manuscript. Part of this work has been presented by the second author at Householder Symposium XVI, Seven Springs, Pennsylvania, May 2005.

Part III: Michael Hanke, K. Henrik A. Olsson and Magnus Strömgren. *Stability Analysis of a Degenerate Hyperbolic System Modelling a Heat Exchanger*. Technical report, TRITA-NA-0518, Royal Institute of Technology, 2005. Submitted to Mathematics and Computers in Simulation. This work has been presented by the first author at GAMM2005, Luxemburg, and a summary has been submitted for publication as a short communication in PAMM. The second author performed the numerical experiments and wrote that section of the paper.

Part IV: K. Henrik A. Olsson. *Model Order Reduction of a Heat Exchanger Model*. Technical report, TRITA-NA-0521, Royal Institute of Technology, 2005. Part of this work has been presented by the author at the workshop Stability of ODEs, DAEs, PDAEs and Their Discretizations, Berlin, 2004.

# Contents

# Chapter 1

# Introduction

*En allmän introduktion*
*ger lite information:*
*med samband och bakgrund,*
*exempel och urkund,*
*beskrivs vår situation.*

## 1.1 General Introduction

Looking up *numerical analysis* in Encyclopædia Britannica Online, 19 April 2005, we could read the following:

'The goal of numerical analysis is the efficient computation of accurate approximations of the values that satisfy mathematical equations. Two major problems confront the numerical analyst: the round-off errors that unavoidably arise during computation, and the representation of problems involving infinite amounts of information with the finite number of values that a person (or computer) can handle.'

We will keep this goal and these problems in mind while introducing the subject of this thesis.

The history of calculus begins with Leibniz and Newton in the 17th century, and since then mathematicians and physicists have derived and analysed differential equations modelling different phenomena. One famous example is Newton's second law of motion, stating that the acceleration of an object multiplied by its mass is equal to the net force acting on it. This is a differential equation for computing the position of an object, since acceleration is the second derivative, with respect to time, of the position. We write it as

$$m\frac{d^2}{dt^2}x(t) = F(t), \text{ or } m\ddot{x}(t) = F(t),$$

where $x$ denotes position, $t$ time, $F$ force, and $m$ mass. If the mass is connected to

a damper and a spring, and we let $x$ denote spring displacement, we get

$$m\ddot{x}(t) + d\dot{x}(t) + kx(t) = f(t), \tag{1.1}$$

where $d$ is a damping constant, $k$ a spring constant, and $f$ the sum of all external forces acting in the $x$-direction. We note that we can write this second order equation as an equivalent first order system:

$$\begin{bmatrix} d & m \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} -k & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} f(t) \\ 0 \end{bmatrix}.$$

Given initial conditions on the position $x(t_0) = x_0$ and velocity $\dot{x}(t_0) = v_0$ at time $t_0$ it is possible to solve the differential equation for the position $x(t)$ for subsequent times $t \geq t_0$. Proving that there exists a solution is a crucial step in the analysis of a differential equation. It is also desired that the solution is well-posed, in the sense of Hadamard, which means that for each set of given data there is a unique solution that depends continuously on the data. In this example the data consists of the initial conditions. Once we know that there is a solution we try to find it. Unfortunately—or fortunately, for numerical analysts—not all differential equations have a known analytical solution. In particular, facing a partial differential equation defined on a non-trivial domain we must turn to numerical techniques for obtaining an approximate solution.

Assume that we want to compute the deflection of a beam in response to an applied dynamic load, or the vibrations of a drum, or the flow of air past a car. Each of these problems can be approached through the formulation of a partial differential equation, containing spatial and temporal derivatives of the unknown solution and accompanied by appropriate initial values and boundary conditions. This gives a continuous problem where the solution $u(x, t)$ satisfies the equation for all positions $x$ in the given domain and all times $t$ in the interval from the initial time to a final time. This solution belongs to an infinite dimensional function space and in general we are not able to find it. The key is then to discretise the problem and look for an approximate solution, with only a finite number of unknowns, using a computer. For example, we can use a finite difference [13] or finite element [5] discretisation of the spatial variable to transform the partial differential equation into a system of ordinary differential equations in time, and then solve it using a time-stepping algorithm designed for such systems. In the discretisation we usually choose a mesh size parameter; a smaller parameter value gives a larger discrete system with a solution closer to that of the continuous problem, and conversely a larger value gives a smaller system with a less accurate approximate solution. This is an important trade-off between accuracy and cost (in terms of the computing time and computer memory required).

To illustrate the discretisation process, we use the example of a two dimensional drum skin fastened at a plane boundary $\Gamma$ and occupying the interior $\Omega$. The

vibrations of the skin can be computed from the wave equation

$$\begin{aligned} \ddot{u}(x,t) - \Delta u(x,t) &= f(x,t), \quad x \in \Omega \subset \mathbb{R}^2, \quad t \geq 0, \\ u(x,t) &= 0, \quad x \in \Gamma, \quad t \geq 0, \\ u(x,0) &= u_0(x), \\ \dot{u}(x,0) &= v_0(x), \end{aligned} \tag{1.2}$$

where $u$ is the displacement of the skin perpendicular to its flat rest position and $f$ is the external force acting on the skin. The Laplace operator $\Delta$ contains second order spatial derivatives. If $\Gamma$ is not of simple shape such as a circle we resort to finding a numerical solution. In a basic finite element or finite difference method, we introduce a mesh that covers the domain $\Omega$ and look for the value of the solution in the mesh points. The finer the mesh, the more unknown point values to compute. With these values collected in a vector $U$, we transform the term $\Delta u$ into a matrix-vector product $AU$ through a variational formulation or application of a finite difference stencil, taking the boundary values into account. The matrix $A$ is then large and sparse, with only a small fraction of the many elements nonzero. Finally, from the given initial conditions we get the discrete initial conditions necessary for starting the time-stepping and can compute a solution up to a final time.

## 1.2 Model Order Reduction

From a spatial discretisation of a partial differential equation, as described in the previous section, we can get a linear time-invariant dynamical system known as a generalised state space system:

$$\begin{aligned} E\dot{x} &= Ax + bu, \\ y &= cx, \end{aligned} \tag{1.3}$$

with constant matrices $E, A \in \mathbb{R}^{n \times n}$ and vectors $b \in \mathbb{R}^{n \times 1}$, $c \in \mathbb{R}^{1 \times n}$. The vector $x$, called the state, contains the unknown values to be computed, and there is an input function $u$ affecting the system and an output function $y$ observing the state. The matrices $E$ and $A$ are in this case usually sparse, and $E$ may be singular when either the original problem is a partial differential algebraic equation or boundary values are included in the state. If the number of states $n$ is large, a computer simulation of the system can be costly in terms of CPU time and memory used. Therefore, we would like to derive a reduced order system of the same form with $r \ll n$ states,

$$\begin{aligned} \widehat{E}\dot{\widehat{x}} &= \widehat{A}\widehat{x} + \widehat{b}u, \\ \widehat{y} &= \widehat{c}\widehat{x}, \end{aligned} \tag{1.4}$$

such that the output error $y - \widehat{y}$ is small for a given input $u$.

For simplicity, we consider single-input-single-output (SISO) systems in this section, although some of the methods we describe may be applied to MIMO systems with multiple inputs and outputs.

The transfer function is useful for measuring how well the reduced order system approximates the original system. Using the Laplace transform

$$X(s) \equiv \mathcal{L}\left(x(t)\right)(s) = \int_{0-}^{\infty} e^{-st} x(t) dt,$$

and assuming zero initial conditions, we get the transfer function

$$H(s) = c\left(sE - A\right)^{-1} b$$

that takes an input signal $U(s)$ to an output signal $Y(s) = H(s)U(s)$.

Small state space systems on the standard form with an identity matrix $E = I$ are common in control theory, and much work has been done on approximation using reduced order models; see, for example, [19] and the references therein. The controllability (sometimes called reachability) and observability grammians

$$\mathcal{P} = \int_0^{\infty} e^{At} b b^T e^{A^T t} dt, \quad \mathcal{Q} = \int_0^{\infty} e^{A^T t} c^T c\, e^{At} dt,$$

describe the past input energy needed to reach a given state and the future output energy from a given state, respectively. They can be computed from the following Lyapunov equations:

$$\begin{aligned} A\mathcal{P} + \mathcal{P}A^T &= -bb^T, \\ A^T\mathcal{Q} + \mathcal{Q}A &= -c^T c. \end{aligned}$$

Then, a balancing transformation can be done such that the grammians are diagonal and equal, after which the states that are both difficult to reach and difficult to observe can be truncated; see, for example, [25] and the references therein. The Hankel singular values $\sigma_i$ are the square roots of the eigenvalues of the product $\mathcal{P}\mathcal{Q}$, and a reduced order model of size $k$ obtained from balanced truncation satisfies the error bound

$$\|H - \widehat{H}\|_{\mathbb{H}_\infty} = \sup_{\omega \in \mathbb{R}} \|H(i\omega) - \widehat{H}(i\omega)\|_2 \leq 2 \sum_{i=k+1}^{n} \sigma_i.$$

This is a global error bound in the sense that it covers all frequencies $\omega$.

In designing a controller, it can be desired to reduce the number of states from say $n = 10$ to $r = 4$, and the methods that are used for solving Lyapunov equations and balancing grammians use dense matrix operations with complexity $O(n^3)$. For systems with large, sparse matrices this is not reasonable, and methods for approximate balanced truncation are developed, taking advantage of the facts that the right hand sides in the Lyapunov equations are of low rank and the Hankel singular values often decay rapidly; see, for example, [15], [25], [1].

The main alternative to balanced truncation for model order reduction is moment matching. With an expansion of the transfer function at a point $\sigma$ we get

$$\begin{aligned} H(s) &= c\,(sE - A)^{-1}\,b \\ &= -c\left(I - (s-\sigma)(A-\sigma E)^{-1}E\right)^{-1}(A-\sigma E)^{-1}b \\ &= \sum_{j=0}^{\infty} -(s-\sigma)^j\, c\big\{\,(A-\sigma E)^{-1}\,E\big\}^j\,(A-\sigma E)^{-1}\,b \\ &= \sum_{j=0}^{\infty} (s-\sigma)^j\, m_j(\sigma), \end{aligned}$$

where the coefficients

$$m_j(\sigma) \equiv -c\big\{\,(A-\sigma E)^{-1}\,E\big\}^j\,(A-\sigma E)^{-1}\,b$$

are called moments at $\sigma$. Explicitly computing these moments and constructing a matching reduced order model can be done, but the process usually becomes ill-conditioned already for matching a few moments; see, for example, [6]. Fortunately, we can get moment matching using a projection method with basis matrices $V$ and $Z$ to arrive at a reduced order model (1.4) with $\widehat{E} = Z^T E V$, $\widehat{A} = Z^T A V$, $\widehat{b} = Z^T b$, $\widehat{c} = cV$, and an approximate state vector given by $x = V\widehat{x}$. We will see in Section 1.4 how to choose $V$ and $Z$ to match a desired number of moments in desired points, and that these matrices can be constructed using rational Krylov methods. Note that with moment matching we only know the approximation quality locally in the points used, there is no global error bound.

From some applications, we naturally get a second order system

$$\begin{aligned} M\ddot{x} + D\dot{x} + Kx &= bu, \\ y &= cx, \end{aligned} \tag{1.5}$$

analogous to the equation (1.1) for the mass connected to a spring and a damper. Then we can either use what is called a linearisation and treat the system

$$\begin{aligned} \begin{bmatrix} D & M \\ I & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} &= \begin{bmatrix} -K & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix} u, \\ y &= [c\ 0] \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \end{aligned}$$

of twice the size or use a method designed for second order systems.

## 1.3 Eigenvalue Computation

We are often interested in computing the eigenvalues of a matrix $A$, that is, the values of $\lambda$ such that $Ax = \lambda x$ for a nonzero vector $x$ which is then a corresponding eigenvector. For each such eigenpair $(\lambda, x)$, the matrix $A - \lambda I$ is obviously singular

so the eigenvalues are the solutions to the equation $\det(A - \lambda I) = 0$. This is a polynomial equation in $\lambda$ of degree equal to the size of $A$, which means that for matrices of size larger than four we generally need to use an iterative method for computing the eigenvalues since there are no algebraic expressions for them.

From some applications we get a generalised eigenvalue problem $Ax = \lambda Ex$ with a matrix pair $(A, E)$ also known as a pencil $A - \lambda E$. We restrict our attention to regular pencils having a finite number of eigenvalues. For more details on different eigenvalue problems and methods for solving them, we refer to [2].

The drum example (1.2) gives rise to an eigenvalue problem when we want to compute the eigenfrequencies of the drum to find out how it will sound. The continuous eigenvalue problem is

$$\begin{aligned} -\Delta\phi(x) &= \mu\phi(x), \quad x \in \Omega, \\ \phi(x) &= 0, \quad x \in \Gamma, \end{aligned}$$

with eigenfunctions $\phi(x)$ also called eigenmodes, and after spatial discretisation it becomes

$$Az = \lambda z,$$

where the eigenvectors $z$ approximate the eigenfunctions. The eigenvalue closest to zero corresponds to the basic tone of the drum.

Given a large eigenvalue problem of size $n$ we may not be interested in computing all the $n$ eigenvalues, either because it requires to much computer time and memory or because we are only interested in finding certain parts of the spectrum. For example, for a dynamical system of the form (1.3) with $E = I$ and a diagonalisable matrix $A$, such that $A = V\Lambda V^H$ with $V$ unitary and $\Lambda$ diagonal, we get

$$\dot{z} = \Lambda z, \quad x = Vz,$$

which is a decoupled system with solutions $z_i(t) = z_i(0)\exp(\lambda_i t)$. If any eigenvalue $\lambda_i$ on the diagonal of the matrix $\Lambda$ has a positive real part, the corresponding component $z_i$ will grow without bound as the time goes. For the system to be stable, all eigenvalues must have negative real parts. Then, the eigenvalues closest to the imaginary axis correspond to the components $z_i$ with the slowest decay towards zero. Therefore, it is often interesting to compute these eigenvalues close to the imaginary axis. A method for model order reduction called mode reduction consists of keeping precisely the least damped eigenmodes, corresponding to these eigenvalues.

## 1.4 Krylov Subspace Methods

The Lanczos and Arnoldi methods for transforming a Hermitian matrix to tridiagonal form and a non-Hermitian matrix to Hessenberg form, respectively, can be written as

$$AV_j = V_j T_{j,j} + v_{j+1} e_j^T \beta_j$$

and

$$AV_j = V_j H_{j,j} + v_{j+1} e_j^T h_{j+1,j},$$

where $V_j$ is a basis for the Krylov subspace $\mathcal{K}_j(A, v_1) \equiv \text{span}\left\{v_1, Av_1, \ldots, A^{j-1}v_1\right\}$. Originally, these methods were intended for use as an initial step in computing all eigenvalues of a matrix and the algorithms were run up until $j$ was equal to the size $n$ of the matrix $A$. Then, the matrix $V_n$ is unitary and $H_{n,n}$ is similar to $A$ (from now on we focus on the Arnoldi method for the non-Hermitian case to be specific). It was found, however, that the eigenvalues of $H_{j,j}$ gave good approximations to some of the eigenvalues of $A$ already after $j \ll n$ steps. We see that from an eigenpair $(\theta, z)$ of $H_{j,j}$, that is, $H_{j,j}z = \theta z$, we get a Ritz pair $(\theta, V_j z)$ of $A$ with the residual

$$AV_j z - \theta V_j z = V_j(H_{j,j}z - \theta z) + v_{j+1}h_{j+1,j}z_j = v_{j+1}h_{j+1,j}z_j$$

with norm equal to $|h_{j+1,j}z_j|$. If $h_{j+1,j}$ is equal to zero, we have found an invariant subspace $V_j$, that is, $AV_j = V_j H_{j,j}$, and all Ritz pairs are exact eigenpairs.

The convergence of the Ritz values $\theta$ towards the eigenvalues $\lambda$ of $A$ depends on the starting vector $v_1$ and the distribution of the eigenvalues in the complex plane. Well separated eigenvalues of large modulus will be the first to converge. Thus, when we look for eigenvalues close to a point $\sigma$ in the complex plane we can use the shift-and-invert transformation $C = (A - \sigma I)^{-1}$ since

$$Cx = \mu x \Leftrightarrow (A - \sigma I)^{-1}x = \mu x \Leftrightarrow Ax = \left(\sigma + \frac{1}{\mu}\right)x = \lambda x.$$

At each step of the Arnoldi method we have to multiply a vector by the matrix $A$, or solve a system with a shifted version of it, and then orthogonalise against all previous vectors. Usually the matrix is large and sparse, and it is crucial to take advantage of this sparsity. The orthogonalisation is done through some Gram–Schmidt method and will become costly as the size of the basis increases. Therefore, in a practical implementation we need some strategy for reducing the basis size by removing unwanted directions and restarting. We also know that an unreduced Hessenberg matrix cannot have multiple eigenvalues, so for finding multiple copies we need a method for locking converged eigenvalues.

For the generalised eigenvalue problem $Ax = \lambda Ex$, we get the shift-and-invert transformation from

$$Ax = \lambda Ex \Leftrightarrow (A - \sigma E)x = (\lambda - \sigma)Ex \Leftrightarrow (A - \sigma E)^{-1}Ex = \frac{1}{\lambda - \sigma}x = \mu x.$$

The rational Krylov sequence method for eigenvalue computation [22] is designed for using several shift points $\sigma$, and keeping the subspace found with one shift when moving to the next. With this method, starting from a vector $x$ we generate a basis for the rational Krylov sequence

$$\begin{aligned} &x,\ (A - \sigma_1 E)^{-1}Ex,\ \left\{(A - \sigma_1 E)^{-1}E\right\}^2 x,\ \ldots, \left\{(A - \sigma_1 E)^{-1}E\right\}^{j_1} x, \\ &(A - \sigma_2 E)^{-1}Ex,\ \ldots, \left\{(A - \sigma_K E)^{-1}E\right\}^{j_K} x. \end{aligned}$$

The advantage is that we can get rapid convergence of eigenvalues in different regions—at each shift $\sigma_k$—and the price to pay for this is that we make a sparse LU-factorisation of each shifted matrix $A - \sigma_k E$, for $k = 1, \ldots, K$.

Krylov subspace methods have been used for model order reduction since it was found that they could be used for generating a moment matching reduced order model without the ill-conditioned explicit matching from computed moments; see Gallivan, Grimme and Van Dooren [9], Feldmann and Freund [6]. The first such methods were designed for SISO systems on standard state space form, with matching at either zero or infinity [9] or at one arbitrary point [6]. Then a Lanczos method for Padé approximation of MIMO systems was given by Freund [7], and a rational Lanczos method was developed by Gallivan, Grimme and Van Dooren [10]. Grimme [11] showed that projections onto unions of Krylov subspaces for shift-and-invert with several shifts gives moment matching at all these points, for a MIMO system with $m$ inputs and $p$ outputs: If

$$\bigcup_{k=1}^{K} \bigcup_{l=1}^{m} \mathcal{K}_{J_b(k,l)}\left\{ (A-\sigma_k E)^{-1} E, (A-\sigma_k E)^{-1} B(:,l) \right\} \subseteq \operatorname{span}\{V\},$$
$$\bigcup_{k=1}^{K} \bigcup_{l=1}^{p} \mathcal{K}_{J_c(k,l)}\left\{ (A-\sigma_k E)^{-T} E^T, (A-\sigma_k E)^{-T} C(l,:)^T \right\} \subseteq \operatorname{span}\{Z\},$$

then the moments $m_j(\sigma_k)$ of the original and $\widehat{m}_j(\sigma_k)$ of the reduced order models are equal for $j = 0, \ldots, J_b(k, l_b) + J_c(k, l_c) - 1$; $k = 1, \ldots, K$; $l_b = 1, \ldots, m$; and $l_c = 1, \ldots, p$. This also shows what starting vectors to use in a rational Krylov method for moment matching. Such methods were studied by Skoogh [24].

The systems we apply our model order reduction techniques to are usually causal and stable, that is, the output values at a certain time depend on past input values but not on future input values and the output is bounded for all bounded inputs. Then, starting from zero initial conditions, a periodic input signal $u(t) = u_0 e^{i\omega t}$, with $u_0 \in \mathbb{R}^m$ and $i = \sqrt{-1}$, gives the periodic output signal $y(t) = H(i\omega)u_0 e^{i\omega t}$. Thus, if $\widehat{H}(i\omega) = H(i\omega)$ the reduced order system gives the same output as the full order system when the input is a sine or cosine function of frequency $\omega$. This is one of the reasons for matching moments at the imaginary axis, another is that with poles located there we expect to get a rational Krylov subspace containing good approximations to the dominant eigenmodes, that is, the ones with eigenvalues closest to the imaginary axis.

During the last years most of the work on Krylov subspace methods for model order reduction has been on how to handle second order systems without working on a linearised system; see [4], [8], [29], [16]. Preserving stability and passivity [27] is also an important issue, and there is much work to be done on methods for nonlinear systems [3].

# Chapter 2

# Summary of Reports

*Kanhända är tiden på jorden för kort*
*för alla att läsa varenda rapport,*
*läs här referat*
*av vårt resultat*
*så flyger väl färre av timmarna bort.*

## 2.1 Model Order Reduction in FEMLAB by Dual Rational Arnoldi

Solving a time-dependent partial differential equation (PDE) in a finite element program such as FEMLAB is done through a spatial discretisation leading to a system of ordinary differential equations (in time). The aim for reasonably high accuracy leads to a choice of a small mesh size and thereby to an oftentimes vast number of degrees of freedom. It is therefore natural to consider techniques for model order reduction, especially if the model is to be used in a control system program such as Simulink designed for working on systems with small, dense matrices.

Our aim in the work presented here was to implement a method for model order reduction in FEMLAB that would be more general and efficient than the mode reduction by projection onto dominant eigenmodes and static modes that was already available.

We start from a system in generalised state space form

$$\begin{aligned} E\dot{x} &= Ax + Bu, \\ y &= Cx, \end{aligned}$$

obtained after linearisation in the case of a nonlinear PDE. The matrices $E$ and $A$ are large and sparse, and the pencil $A - \lambda E$ is regular although $E$ may be singular. There can be several input and output signals, in which case $B$ and $C$ have several columns and rows, respectively.

The main methods for model order reduction are balanced truncation and moment matching. Since we have a generalised state space system with large, sparse matrices we choose to use a Krylov method for moment matching. In general, the matrices $A$ and $E$ are non-normal and $C^T \neq B$ so the system is not symmetric. We want a robust method for generating a small reduced order model—rather spending a bit more time generating a smaller model—and use a dual rational Arnoldi method.

In the dual rational Arnoldi method we generate basis matrices $V$ and $Z$, each with orthonormal columns, such that

$$\begin{aligned} \operatorname{span}\{V\} &= \bigcup_{k=1}^{K} \bigcup_{l=1}^{m} \mathcal{K}_{J_b(k)}\left\{ (A-\sigma_k E)^{-1} E, (A-\sigma_k E)^{-1} b_l \right\}, \\ \operatorname{span}\{Z\} &= \bigcup_{k=1}^{K} \bigcup_{l=1}^{p} \mathcal{K}_{J_c(k)}\left\{ (A-\sigma_k E)^{-H} E^H, (A-\sigma_k E)^{-H} C(l,:)^H \right\}. \end{aligned}$$

The shift points $\sigma_k$, $k = 1, \ldots, K$, are given by the user or chosen as imaginary points covering a given frequency interval. The number of vectors to generate at each shift point from each starting vector is also specified by the user and determines the size of the reduced order model. We generate a reduced order model by explicit multiplication with the basis matrices to arrive at

$$\begin{aligned} \widehat{E}\dot{\widehat{x}} &= \widehat{A}\widehat{x} + \widehat{B}u, \\ \widehat{y} &= \widehat{C}\widehat{x}, \end{aligned}$$

where $\widehat{E} = Z^H E V$, $\widehat{A} = Z^H A V$, $\widehat{B} = Z^H B$, and $\widehat{C} = CV$, since we know from Grimme [11] that this will give a reduced order model matching a maximal number of moments in the chosen shift points.

We use a vector-wise construction with deflation for building up the bases $V$ and $Z$; note that we handle MIMO systems and use several shift points so it is not entirely unlikely that we will find linear dependencies in the unions of Krylov spaces. For each shift point $\sigma_k$, we make a sparse LU-factorisation of $(A - \sigma_k E)$ that we use for solving systems with this matrix when generating the corresponding parts of $V$ and $Z$. If more vectors have been deflated in the construction of either $V$ or $Z$, we add more vectors to this smaller matrix to get quadratic matrices $\widehat{E}$ and $\widehat{A}$ and match extra moments. Usually, the state space matrices of the original system are real and we use imaginary shift points but want to arrive at a real reduced order model. We do this by implicitly using the complex conjugate of each complex shift point and adding two real vectors to the basis using the ideas of Ruhe [21].

It may happen that the reduced order model is unstable although the original model was stable. In such cases, we have seen that usually only one or two eigenvalues lie in the right half plane and we truncate this unstable part using an additive decomposition of the transfer function [14]. Keeping only the stable part of the transfer function, we no longer get the moment matching. Hopefully, and

this is usually a reasonable hopefulness, the unstable poles are weak in the sense that they have a small influence on the transfer function, and we cheaply evaluate the error in the transfer function caused by the truncation and give a warning if it is not small.

We discuss the choice of shift points and the use of an iterative solver instead of a matrix factorisation.

Finally, we apply our code to two test problems in FEMLAB and compare the results with those obtained using projection onto eigenmodes and static modes.

## 2.2 Rational Krylov for Model Order Reduction and Eigenvalue Computation

Since the rational Krylov sequence method for eigenvalue computation was introduced by Ruhe in 1984 [20] it has undergone a few changes and also been applied to model order reduction. In this report we try to take it one step further, using the knowledge gathered during these twenty years and making a new implementation.

We consider the matrix pencil $A - \lambda E$, corresponding to a generalised eigenvalue problem $Ax = \lambda Ex$ or a generalised state space system $E\dot{x} = Ax + bu$, $y = cx$. In a rational Krylov method, starting from a vector $v$ we look for approximations using a basis $V$ for the space spanned by the vectors in the sequence

$$\begin{aligned} &v,\ (A-\sigma_1 E)^{-1}Ev,\ \left\{(A-\sigma_1 E)^{-1}E\right\}^2 v,\ \ldots,\left\{(A-\sigma_1 E)^{-1}E\right\}^{j_1} v,\\ &(A-\sigma_2 E)^{-1}Ev,\ \ldots,\left\{(A-\sigma_K E)^{-1}E\right\}^{j_K} v. \end{aligned}$$

In an earlier formulation [22], two Hessenberg matrices $H$ and $K$ are constructed such that

$$AV_{j+1}H_{j+1,j} = EV_{j+1}K_{j+1,j},$$

but here we consider the more recent version [23] using only one Hessenberg matrix. We also include a shift $\mu$, in addition to the pole $\sigma$, since this is used for nonlinear problems and might give some extra freedom in the formulation (although the choice of shift does not affect the Krylov space generated), and get the basic recursion

$$(A-\sigma E)^{-1}(A-\mu E)\,V_j = V_{j+1}H_{j+1,j}\,,$$

where $H_{j+1,j}$ is an unreduced Hessenberg matrix.

We show how to update the recursion without operating with the large matrices $A$ and $E$ when moving to a new pole $\sigma_1 \neq \sigma$ and a new shift $\mu_1 \neq \mu$, where obviously $\mu_1 \neq \sigma_1$, to get

$$(A-\sigma_1 E)^{-1}(A-\mu_1 E)\,\widetilde{V}_j = \widetilde{V}_{j+1}\widetilde{H}_{j+1,j}$$

with $\operatorname{span}\{\widetilde{V}_{j+1}\} = \operatorname{span}\{V_{j+1}\}$. This involves a QR-decomposition

$$Q_{j+1,j+1}R_{j+1,j} = \frac{\mu-\sigma_1}{\mu_1-\sigma_1}\begin{bmatrix} I_{j,j} \\ 0_{1,j} \end{bmatrix} - \frac{\sigma-\sigma_1}{\mu_1-\sigma_1}H_{j+1,j}\,,$$

and a transformation to Hessenberg form of the matrix

$$Q_{j+1,j+1}^H \left( \frac{\mu_1 - \sigma}{\mu_1 - \sigma_1} H_{j+1,j} - \frac{\mu_1 - \mu}{\mu_1 - \sigma_1} \begin{bmatrix} I_{j,j} \\ 0_{1,j} \end{bmatrix} \right) R_{j,j}^{-1},$$

where we obviously want to avoid an ill-conditioned $R_{j,j}$.

When computing eigenvalues we must know how to lock converged eigenvalues and purge unwanted ones. We use the ideas of Meerbergen [17] and Stewart [26] for locking and purging by computing and reordering a Schur form of the Hessenberg matrix to get a new basis. Starting from a recursion of size $j$ with $l$ locked eigenvalues

$$(A - \sigma E)^{-1} (A - \mu E) \begin{bmatrix} V_l & V_{j-l} \end{bmatrix} = \begin{bmatrix} V_l & V_{j+1-l} \end{bmatrix} \begin{bmatrix} T_{l,l} & M_{l,j-l} \\ 0 & H_{j-l+1,j-l} \end{bmatrix},$$

where $T_{l,l}$ is triangular and $H_{j-l+1,j-l}$ is unreduced Hessenberg, we expand it to size $k$, at each step operating on the last vector and orthogonalising against all vectors. We show how to do locking and purging working on the active Hessenberg part $H$, and how to update the passive locked parts $T$ and $M$ when changing poles and shifts.

Assuming errors in the locked and active parts of the recursion,

$$(A - \sigma E)^{-1} (A - \mu E) \begin{bmatrix} V_1 & V_2 \end{bmatrix} = \begin{bmatrix} V_1 & V_2 & v_3 \end{bmatrix} \begin{bmatrix} T & M \\ 0 & H \end{bmatrix} + \begin{bmatrix} \Phi_1 & \Phi_2 \end{bmatrix},$$

and an error in the QR-decomposition, we derive expressions for the forward errors in the recursion after a change of poles and shifts similarly to Meerbergen [17]. For the locked part we find that we should not place the new pole close to an already converged eigenvalue, which is rather obvious. For the active part we find that the norms of $R^{-1}$ and $(A - \sigma_1 E)^{-1}(A - \mu_1 E)$ are the important factors to control.

For model order reduction, we show that using the reduced order system matrices

$$\begin{aligned} \widehat{A} &= \tfrac{1}{\sigma - \mu} \left( \sigma H_{j,j} - \mu I_{j,j} \right), \\ \widehat{E} &= \tfrac{1}{\sigma - \mu} \left( H_{j,j} - I_{j,j} \right), \\ \widehat{c} &= c V_j, \\ \widehat{b} &= V_j^H (A - \sigma E)^{-1} b, \end{aligned}$$

we get moment matching at all poles used up until step $j$. We write the error in the approximation of the transfer function,

$$e(s) = \widehat{H}(s) - H(s) = e_1(s) e_2(s),$$

as the product of a factor

$$e_1(s) \equiv c (sE - A)^{-1} (\sigma E - A) v_{j+1}$$

containing the large matrices and a factor

$$e_2(s) \equiv -k_{j+1,j}(s)e_j^H K_{j,j}^{-1}(s)t_j$$

containing only quantities from the small system which thereby is cheaply evaluated. A similar splitting was done by Gallivan, Grimme and Van Dooren [10]. Our aim is to use the part $e_2(s)$ as an error estimate for choosing where to place the next pole, and in the numerical experiments we investigate the possibility of this.

We discuss how to handle second order systems, and find that we have to use the linearisation to a system of twice the size. We also give some thought to the problems of constructing real reduced order systems using complex shifts and choosing poles and shifts, and present the not entirely successful results of our attempts.

We have implemented the algorithms for eigenvalue computation and model order reduction in MATLAB and present results from applications to one illustrative toy example and two benchmark problems. Using these test problems we study strategies for pole placement, how the error in the recursion evolves, if the error estimate is useful, and the trade-off between the cost of a new factorisation when changing poles and the cost of generating a larger basis at the old pole.

## 2.3 Stability Analysis of a Degenerate Hyperbolic System Modelling a Heat Exchanger

We would like to derive and analyse a numerical model of a carbon dioxide heat pump, since experiments have shown two different steady-states with different coefficients of performance. Our first step is to have a look at one of the heat exchangers, the evaporator. Assuming a low Mach-number, we use the following simplified one-dimensional compressible Euler equations for modelling the two-phase flow in a long slender tube with constant cross section area $A$:

$$\begin{aligned} A\frac{\partial f}{\partial p}\frac{\partial p}{\partial t} + A\frac{\partial f}{\partial h}\frac{\partial h}{\partial t} + \frac{\partial F}{\partial x} &= 0, \\ A\frac{\partial p}{\partial x} &= R, \\ -A\frac{\partial p}{\partial t} + Af\frac{\partial h}{\partial t} + F\frac{\partial h}{\partial x} &= Q, \end{aligned}$$

where $p$ is the pressure, $h$ the mass specific enthalpy, $\rho = f(p,h)$ the density, $T = g(p,h)$ the temperature, and $F = A\rho u$ the mass flow rate for a velocity $u$. The friction is modelled as

$$R = -L_f F|u| = -L_f \frac{F^2}{A\rho}\text{sign}(F),$$

where $L_f$ is a constant, and the energy exchange is given by

$$Q = A_{exch}\alpha(T_{air} - T),$$

where $A_{exch}$ is the heat exchanger area, $\alpha$ a heat conductivity coefficient and $T_{air}$ the temperature of the surrounding air.

We now have a PDAE for the unknowns $F$, $p$ and $h$, and want to find out how to specify boundary conditions and initial conditions to obtain a well-posed problem.

We linearise the PDAE and freeze coefficients to arrive at a constant coefficient linear system of the form

$$Au_t + Bu_x + Cu = G,$$

that can be transformed to the canonical form

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} v_t + \begin{pmatrix} u^c & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} v_x + Dv = H,$$

with a hyperbolic equation for $v_1$, a parabolic equation for $v_2$, and an algebraic equation for $v_3$. Thus, we have to give one boundary condition according to the characteristic and the other two at opposite boundaries. Using such boundary conditions, we derive the energy estimate

$$\begin{aligned} \|v(\cdot,t)\|^2 \quad \leq \quad & C(t)\{\|v_1(\cdot,0)\|^2 + \|v_2(\cdot,0)\|^2 + \|v_{1,x}(\cdot,0)\|^2 + \|v_{2,x}(\cdot,0)\|^2 \\ & + \int_0^t \|H(\cdot,\tau)\|^2 + \|H_x(\cdot,\tau)\|^2 d\tau\} \end{aligned}$$

containing first spatial derivatives of initial data and forcing functions and thus indicating that the problem is weakly ill-posed.

Going back to the physical variables we find that $h$ should be given at the inflow while $F$ and $p$ should be given at opposite boundaries.

After this linear stability analysis, we also make a numerical stability analysis of the nonlinear system on the form

$$\begin{aligned} \frac{\partial \rho}{\partial t} &= -\frac{1}{A}\frac{\partial F}{\partial x}, \\ 0 &= -A\frac{\partial p}{\partial x} - L_f \frac{F|F|}{A\rho}, \\ \frac{\partial e}{\partial t} &= -\frac{1}{A}\frac{\partial Fh}{\partial x} + \frac{A_{exch}\alpha(T_{air}-T)}{A}, \\ 0 &= -\rho + f(p,h), \\ 0 &= \rho h - p - e, \\ 0 &= -T + g(p,h), \end{aligned}$$

using a finite difference discretisation in space. We now need to give numerical boundary conditions for $F$, $p$ and $h$ at either the inflow or the outflow. To investigate stability, we compute a steady-state solution to start from and look at the spectrum of a linearisation of the DAE at this solution. Testing all eight possible combinations of where to prescribe these boundary values, we find that the two

combinations that give stable spectra are: $F$ and $h$ given at the inflow, $p$ at the outflow; and $p$ and $h$ given at the inflow, $F$ at the outflow. In both cases, the condition for $h$ is given at the inflow and the other ones are given at opposite boundaries, which agrees with the findings from the linear analysis. We also check the stability by time-stepping the system and finding that these two combinations are the ones that work, and the former is preferable since it needs fewer steps.

The equations of state we want to use for computing $f(p,h)$ and $g(p,h)$ are continuous and monotone, but they are not differentiable at the saturated vapour line in the pressure–enthalpy diagram. Since we want a differentiable right hand side for the DAE solver, we use interpolating $C^1$ monotone quadratic spline surfaces $f$ and $g$ approximating the equations of state.

As expected, the solver identifies the system as an index-1 DAE.

## 2.4 Model Order Reduction of a Heat Exchanger Model

The heat exchanger model we studied in [12] is interesting for us to use as a test problem for model order reduction methods, since it is a new model intended for further use in numerical simulations and we can compare our results with those obtained using a lumped model obtained through spatial integration as described in [28].

We start from a nonlinear PDAE model of the evaporator and discretise in space to get a system of the form

$$\begin{aligned} E\dot{z} &= f(z) + bu, \\ y &= cz, \end{aligned}$$

where $z$ is the state vector containing values of the mass flow rate $F$, pressure $p$, enthalpy $h$, density $\rho$, internal energy $e$ and temperature $T$ in the discretisation points. The descriptor matrix $E$ is singular and $f$ is a nonlinear function. We use the pressure at the outflow as input variable $u$ and the enthalpy at the outflow as output variable $y$.

Linearising at a steady state we get a generalised state space system that is not symmetric. We apply our dual rational Arnoldi method for model order reduction to this system, and also a variant of balanced truncation for descriptor systems [18]. In the Arnoldi method we need to specify poles where we want to match moments in the transfer function, and how many moments to match at each pole. These choices determine the size of the reduced order system. For balanced truncation we only have to give an error tolerance for the approximation of the transfer function; the choice of tolerance implicitly determines the reduced system size.

We try two methods for nonlinear model order reduction. First, we use a lumped model obtained after several simplifying assumptions and spatial integration. This gives a nonlinear DAE in only 4 variables. Second, we use a Galerkin projection onto a POD basis computed using a simulation of the full nonlinear system. This also gives a nonlinear DAE, but its size depends on how well the data from the simulation should be approximated.

In our numerical experiments we see the expected approximation properties in the transfer functions of the reduced order models obtained using the linearised full order system. For comparing all the methods we use a specific input signal and compare the output signals from the reduced order models with that from the original model. We then see that the linear models give similar results while that from the nonlinear POD model is a bit better. The output from the lumped model is quite far from the other ones.

# Chapter 3

# Conclusions

*Nu har vi kommit fram till slutsatserna*
*där vi skall se vad vi har kommit fram till.*

We have studied two rational Krylov methods for model order reduction by moment matching. The first is a dual rational Arnoldi method where two basis matrices are constructed and used for generating a reduced order model by explicit multiplication with the full order system matrices; this is a robust way to compute a model matching a maximal number of moments. The second is a rational Krylov method where the reduced order system matrices are obtained from the Hessenberg matrix of orthogonalisation coefficients; this is a more elegant way to compute a moment matching model. In both methods, a main issue that is still not resolved is how to automatically choose poles, and the number of moments to match at each pole, in an optimal way.

In the dual rational Arnoldi method, we handle models with multiple inputs and outputs, retain real models by implicitly using the conjugate of complex poles, and stabilise when needed by truncating the unstable part. Constructing basis matrices for subsequent multiplication, we are not looking at eigenvalue convergence or keeping track of any low-order information that might be used for error estimates and choosing new poles.

In the rational Krylov method, we always have a low-order representation—based on the Hessenberg matrix—of the system in the present subspace. This is useful for constructing cheaply evaluated error estimates, but it is still not clear to us how to keep the special form of the recursion while constructing real reduced order models using complex poles and how to get moment matching for second order systems without linearisation.

We have also used the rational Krylov method for eigenvalue computation. Then, locking and purging becomes important and we describe in some detail how to do this, working on an active part and updating the locked part when changing poles. From a forward error analysis we get some not too intriguing indications of when a change of poles can introduce a large error in the recursion. We have used

a heuristic for choosing where to place the next pole based on the converged Ritz values, but there are still several parameters to be chosen in concert when running the program.

Always on the lookout for interesting test problems for our model reduction methods, we have been involved in a stability analysis of an evaporator model. This model is a nonlinear PDAE. From a linear stability analysis, working on a frozen coefficient linearisation transformed to a canonical form, we derive an energy estimate and see how to prescribe boundary data. We also make numerical simulations using a finite difference discretisation in space, supporting the linear stability analysis.

The evaporator is part of a heat pump system, and we want to look at low order models for future simulations of the entire system. Our numerical evaporator model is a nonlinear DAE. We linearise it to obtain a generalised state space system, and apply one of our Krylov methods for moment matching as well as a balanced truncation method. We also have a look at nonlinear methods, using Galerkin projection onto a POD basis as well as a physically motivated lumping method. In this perspective, we find the two linear methods to give similar results. We have reasons to believe that model order reduction methods will be useful in full system simulations, but the choice of method and model size will have to depend on the range of input signals.

# Bibliography


[1] J. M. Badía, P. Benner, R. Mayo, E. S. Quintana-Ortí, G. Quintana-Ortí, and J. Saak. Parallel order reduction via balanced truncation for optimal cooling of steel profiles. Technical report, Preprint SFB393/05-02, TU Chemnitz, 2005.

[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, editors. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, Philadelphia, 2000.

[3] Z. Bai and D. Skoogh. A projection method for model reduction of bilinear dynamical systems. *Linear Algebra and its Applications*, 2005, to appear.

[4] Z. Bai and Y. Su. Dimension reduction of large-scale second-order dynamical systems via a second-order Arnoldi method. *SIAM Journal on Scientific Computing*, 26(5):1692–1709, 2005.

[5] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Studentlitteratur, Lund, 1996.

[6] P. Feldmann and R. W. Freund. Efficient linear circuit analysis by Padé approximation via the Lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14(5):639–649, 1995.

[7] R. W. Freund. Computation of Matrix Padé Approximations of Transfer Functions via a Lanczos-type Process. In C. K. Chui and L. L. Schumaker, editors, *Approximation Theory VIII*, volume 1: Approximation and Interpolation, pages 215–222. World Scientific Publishing Co., Singapore, 1995.

[8] R. W. Freund. *Padé-Type Model Reduction of Second-Order and Higher-Order Linear Dynamical Systems.* In P. Benner, G. Golub, V. Mehrmann, and D. Sorensen, editors, *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lecture Notes in Computational Science and Engineering*, pages 191–223. Springer-Verlag, Berlin/Heidelberg, 2005.

[9] K. Gallivan, E. Grimme, and P. Van Dooren. Asymptotic waveform evaluation via a Lanczos method. *Applied Mathematics Letters*, 7:75–80, 1994.

[10] K. Gallivan, E. Grimme, and P. Van Dooren. A rational Lanczos algorithm for model reduction. *Numerical Algorithms*, 12:33–63, 1995.

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

[12] M. Hanke, K. H. A. Olsson, and M. Strömgren. Stability analysis of a degenerate hyperbolic system modelling a heat exchanger. Technical Report TRITA-NA-0518, Royal Institute of Technology, Stockholm, 2005.

[13] C. Hirsch. *Numerical Computation of Internal and External Flows*, volume 1: Fundamentals of Numerical Discretization. John Wiley & Sons, 1988.

[14] B. Kågström and P. Van Dooren. A generalized state-space approach for the additive decomposition of a transfer matrix. *Journal of Numerical Linear Algebra with Applications*, 1:165–181, 1992.

[15] J. Li and J. White. Low rank solution of Lyapunov equations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):260–280, 2002.

[16] R.-C. Li and Z. Bai. Structure-preserving model reduction using a Krylov subspace projection formulation. *Comm. Math. Sci.*, 3(2):179–199, 2005.

[17] K. Meerbergen. Changing poles in the rational Lanczos method for the Hermitian eigenvalue problem. *Numerical Linear Algebra with Applications*, 8:33–52, 2001.

[18] V. Mehrmann and T. Stykel. *Balanced truncation model reduction for large-scale systems in descriptor form.* In P. Benner, V. Mehrmann, and D. Sorensen, editors, *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lecture Notes in Computational Science and Engineering*, pages 83–115. Springer-Verlag, Berlin/Heidelberg, 2005.

[19] G. Obinata and B. D. O. Anderson. *Model Reduction for Control System Design*. Springer, New York, 2001.

[20] A. Ruhe. Rational Krylov sequence methods for eigenvalue computation. *Linear Algebra and its Applications*, 58:391–405, 1984.

[21] A. Ruhe. The rational Krylov algorithm for nonsymmetric eigenvalue problems. III: Complex shifts for real matrices. *BIT*, 34:165–176, 1994.

[22] A. Ruhe. Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils. *SIAM Journal on Scientific Computing*, 19(5):1535–1551, 1998.

[23] A. Ruhe. Rational Krylov Subspace Method (Section 8.5). In Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, editors, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, pages 246–249. SIAM, Philadelphia, 2000.

[24] D. Skoogh. *Krylov Subspace Methods for Linear Systems, Eigenvalues and Model Order Reduction*. PhD thesis, Chalmers University of Technology, 1998.

[25] D. C. Sorensen and A. C. Antoulas. The Sylvester equation and approximate balanced reduction. *Linear Algebra and its Applications*, 351–352:671–700, 2002.

[26] G. W. Stewart. A Krylov–Schur algorithm for large eigenproblems. *SIAM Journal on Matrix Analysis and Applications*, 23(3):601–614, 2001.

[27] Y. Su, J. Wang, X. Zeng, Z. Bai, C. Chiang, and D. Zhou. SAPOR: Second-order Arnoldi method for passive order reduction of RCS circuits. In *Proceedings of IEEE/ACM International Conference on Computer-Aided Design (ICCAD 2004)*, pages 74–79, 2004.

[28] H. Tummescheit. *Design and Implementation of Object-Oriented Model Libraries using Modelica*. PhD thesis, Department of Automatic Control, Lund Institute of Technology, 2002.

[29] A. Vandendorpe and P. Van Dooren. Krylov techniques for model reduction of second-order systems. Technical report, Department of Mathematical Engineering, Catholic University of Louvain, 2004.