A Rational Krylov Method for Model Order Reduction

Daniel Skoogh *

November 1998

Abstract

An algorithm to compute a reduced-order model of a linear dynamic system is described. It is based on the rational Krylov method, which is an extension of the shift-and-invert Arnoldi method where several shifts (interpolation points) are used to compute an orthonormal basis for a subspace. It is discussed how to generate a reduced-order model of a linear dynamic system, in such a way that the Laplace domain transfer function of the reduced order model interpolates the Laplace domain transfer function of the dynamic system in the interpolation points to appropriate degree. It is also discussed how to compute an error estimate of the Laplace domain transfer function of the reduced-order model. Further it is shown how to create a passive reduced-order model in an efficient way by congruence transformation of a dynamic system that models a RLC circuit. The rational Krylov method is applied to several examples in circuit theory.

Keywords: rational, Krylov, model, reduction, passive, Arnoldi
AMS subject classification 65F15, 93A30, 93B40

1 Introduction

1.1 General Introduction

This report is concerned with the use of one-sided subspace methods and multipoint approximation methods for solving linear time-invariant large-scale dynamic systems. The main application area in mind is simulation of large electrical circuits. However, the theory and algorithms developed are suitable also for other application areas.

A time-invariant electrical circuit can be described by the system of linear first-order differential algebraic equations

$$\begin{aligned} \boldsymbol{C}\dot{\boldsymbol{x}}(t) &= -\boldsymbol{G}\boldsymbol{x}(t) + \boldsymbol{b}u(t) \\ y(t) &= \boldsymbol{d}^T\boldsymbol{x}(t) + qu(t) \end{aligned} \tag{1}$$

*Report No. 1998-47 in "Blå serien" preprint series, available at URL: `http://www.math.chalmers.se/Math/Research/Preprints`. Author's address: Department of Mathematics, Chalmers University of Technology and the University of Göteborg, S-41296 Göteborg, Sweden (`skoogh@math.chalmers.se`). Partial support given by TFR, the Swedish Research Council for Engineering Sciences, Dnr 222, 96-555.

$$\boldsymbol{G}, \boldsymbol{C} \in \mathcal{R}^{n \times n},\ \boldsymbol{x}, \boldsymbol{b}, \boldsymbol{d} \in \mathcal{R}^{n},\ u, y \in \mathcal{R}.$$

The matrix $\boldsymbol{G}$ contains the contributions from memory-less elements such as resistors, and the matrix $\boldsymbol{C}$ from memory elements such as capacitors and inductors. The vector $\boldsymbol{x}(t)$ contains the state variables, $u(t)$ is the known input, and $y(t)$ is the unknown output.

In model order reduction we want to create a reduced-order model $\hat{\boldsymbol{d}}$, $\hat{\boldsymbol{G}}$, $\hat{\boldsymbol{C}}$ and $\hat{\boldsymbol{b}}$ of the exact model $\boldsymbol{d}$, $\boldsymbol{G}$, $\boldsymbol{C}$ and $\boldsymbol{b}$ in such a way that the dynamic system of the reduced-order model,

$$\begin{aligned} \hat{\boldsymbol{C}}\dot{\hat{\boldsymbol{x}}}(t) &= -\hat{\boldsymbol{G}}\hat{\boldsymbol{x}}(t) + \hat{\boldsymbol{b}}u(t) \\ \hat{y}(t) &= \hat{\boldsymbol{d}}^T\hat{\boldsymbol{x}}(t) + qu(t) \end{aligned}$$

$$\hat{\boldsymbol{G}}, \hat{\boldsymbol{C}} \in \mathcal{C}^{k \times k},\ \hat{\boldsymbol{x}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{d}} \in \mathcal{C}^{k},\ u,\ \hat{y} \in \mathcal{C}$$

$$k \ll n,$$

approximates the original dynamic system (1) in a good way in some measure and the dimension of the reduced-order model should be smaller or much smaller than the dimension of the original system.

Apply the Laplace transform to the system (1), assume zero initial conditions, and ignore the term $qu(t)$:

$$\begin{aligned} s\boldsymbol{C}\boldsymbol{x}(s) &= -\boldsymbol{G}\boldsymbol{x}(s) + \boldsymbol{b}u(s) \\ y(s) &= \boldsymbol{d}^T\boldsymbol{x}(s). \end{aligned} \tag{2}$$

Here $y(s)$, $\boldsymbol{x}(s)$ and $u(s)$ denote the Laplace transforms of $y(t)$, $\boldsymbol{x}(t)$ and $u(t)$ respectively. From (2) it follows that the Laplace domain transfer function defined as $H(s) = y(s)/u(s)$ is given by

$$H(s) = \boldsymbol{d}^T(\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{b}. \tag{3}$$

Notation

Matrices are written with upper-case bold letters like $\boldsymbol{A}$, vectors are written with lower-case bold letters like $\boldsymbol{v}$ and scalars are written with non-bold letters like s or H. By $f_{i,j}$ we mean the element $\boldsymbol{F}(i,j)$, by $\boldsymbol{F}_{i,j}$ we mean the leading $i \times j$ submatrix of $\boldsymbol{F}$, and by $\boldsymbol{f}_j$ we mean the j:th column of $\boldsymbol{F}$. By $\boldsymbol{V}_j$ we mean the first j columns of the matrix $\boldsymbol{V}$.

1.2 Multipoint Rational Approximation

The matrix $(\boldsymbol{G} + s\boldsymbol{C})^{-1}$ from the expression (3) of the transfer function $H(s)$ can be expressed as

$$(\boldsymbol{G} + s\boldsymbol{C})^{-1} = \frac{\mathrm{adj}(\boldsymbol{G} + s\boldsymbol{C})}{\det(\boldsymbol{G} + s\boldsymbol{C})} \tag{4}$$

where $\mathrm{adj}(\boldsymbol{G} + s\boldsymbol{C})$ stands for the adjugate of the matrix $(\boldsymbol{G} + s\boldsymbol{C})$, namely the transpose of the matrix of cofactors, and $\det(\boldsymbol{G} + s\boldsymbol{C})$ stands for the determinant

of the matrix $(\boldsymbol{G} + s\boldsymbol{C})$. The matrix $\mathrm{adj}(\boldsymbol{G} + s\boldsymbol{C})$ is a polynomial in s of maximum degree $n-1$, and the scalar $\det(\boldsymbol{G} + s\boldsymbol{C})$ is a polynomial in s of maximum degree n. From (4) and (3) we see that the transfer function $H(s)$ can be expressed as a rational function

$$H(s) = \frac{\varphi(s)}{\psi(s)}$$

where $\varphi(s)$ and $\psi(s)$ are polynomials of maximum degree $n-1$ and n respectively. A natural way to approximate $H(s)$ is by a rational function of lower degree,

$$\begin{aligned} \hat{H}(s) &= \frac{\chi_{q-1}(s)}{\phi_q(s)} \\ &= \frac{b_{q-1}s^{q-1} + \ldots + b_1 s + b_0}{a_q s^q + \ldots + a_1 s + 1}, \end{aligned} \tag{5}$$

where $\chi_{q-1}(s)$ and $\phi_q(s)$ are polynomials of degree $q-1$ and q respectively, and $q \ll n$.

Set $s = s_i + \sigma$ and expand $H(s)$ by Taylor expansion around s_i:

$$H(s) = \sum_{j=0}^{\infty} \sigma^j m_j(s_i), \tag{6}$$

where $m_j(s_i)$ is the j:th moment around the interpolation point s_i:

$$m_j(s_i) = \frac{1}{j!} \frac{\partial^j H(s)}{\partial s^j}\Big|_{s=s_i}.$$

Now choose $\bar{i}$ interpolation points $s_i, i = 1, \ldots, \bar{i}$. Here and later we use overbar $\bar{i}$, $\bar{j}$ and $\bar{k}$ to denote the last largest member of a sequence of integers. If the coefficients $b_{q-1}, \ldots, b_0$ and $a_q, \ldots, a_1$ are chosen in such a way that the moments around the interpolation points of the approximate transfer function $\hat{H}(s)$ match the moments of the transfer function $H(s)$,

$$m_j(s_i) = \hat{m}_j(s_i),\ j = 0, \ldots, \bar{j}_i - 1,\ i = 1, \ldots, \bar{i}$$

and

$$\sum_{i=1}^{\bar{i}} \bar{j}_i = 2q$$

then the approximation is a multipoint Padé approximation. If $\bar{i} = 1$ then it is a Padé approximation. This is the maximum number of moments one can hope to match, given (5). A Padé approximation is a rational function where the maximum number of moments is matched around an interpolation point; see [3].

A Padé-type approximation or a multipoint Padé-type approximation is a rational approximation where the maximum number of moments is not matched

around the interpolation points $s_i, i = 1, \ldots, \bar{i}$. The method presented in this report is a multipoint Padé-type approximation.

Matching the moments at the interpolation points $s_i, i = 1 \ldots \bar{i}$ means that $\hat{H}(s)$ interpolates $H(s)$ in the interpolation points $s_i, i = 1, \ldots, \bar{i}$:

$$\frac{\partial^j \hat{H}(s)}{\partial s^j}\Big|_{s=s_i} = \frac{\partial^j H(s)}{\partial s^j}\Big|_{s=s_i},\ j = 0, \ldots, \bar{j}_i - 1,\ i = 1, \ldots, \bar{i}\,.$$

1.3 Relation Between Eigenvalues and Poles

Consider the eigenvalue problem

$$\boldsymbol{G}\boldsymbol{u}_j = -\lambda_j \boldsymbol{C}\boldsymbol{u}_j. \tag{7}$$

Assume that the pencil (7) is diagonalisable and $\boldsymbol{C}$ nonsingular:

$$\boldsymbol{W}^H \boldsymbol{G}\boldsymbol{U} = -\boldsymbol{D}, \quad \boldsymbol{W}^H \boldsymbol{C}\boldsymbol{U} = \boldsymbol{I},$$

where

$$\boldsymbol{D} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n).$$

Now $(\boldsymbol{G} + s\boldsymbol{C})^{-1}$ can be expressed as

$$\begin{aligned}(\boldsymbol{G} + s\boldsymbol{C})^{-1} &= (\boldsymbol{W}^{-H}(-\boldsymbol{D} + s\boldsymbol{I})\boldsymbol{U}^{-1})^{-1} \\ &= \boldsymbol{U}(-\boldsymbol{D} + s\boldsymbol{I})^{-1}\boldsymbol{W}^H.\end{aligned}$$

Set $\boldsymbol{r} = \boldsymbol{W}^H\boldsymbol{b}$ and $\boldsymbol{l} = \boldsymbol{U}^H\boldsymbol{d}$; the transfer function $H(s)$ (3) can be expressed as

$$\begin{aligned}H(s) &= \boldsymbol{l}^H(-\boldsymbol{D} + s\boldsymbol{I})^{-1}\boldsymbol{r} \\ &= \sum_{j=1}^{n} \frac{l_j^H r_j}{s - \lambda_j}.\end{aligned}$$

The poles λ_j of the transfer function $H(s)$ are equal to the eigenvalues of the eigenproblem (7).

1.4 Moments

Let us now describe how the moments can be expressed with matrix algebra. Set $s = s_i + \sigma$ and expand $(\boldsymbol{G} + s\boldsymbol{C})^{-1}$ around s_i in a power series

$$\begin{aligned}(\boldsymbol{G} + s\boldsymbol{C})^{-1} &= (\boldsymbol{G} + (s_i + \sigma)\boldsymbol{C})^{-1} \\ &= (\boldsymbol{I} + \sigma(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C})^{-1}(\boldsymbol{G} + s_i\boldsymbol{C})^{-1} \\ &= \sum_{j=0}^{\infty}(-1)^j\sigma^j((\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}.\end{aligned} \tag{8}$$

From (6) and (8) we see that the moments can be expressed as

$$m_j(s_i) = (-1)^j \boldsymbol{d}^T((\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}. \tag{9}$$

1.5 Explicit Moment Computation Methods

In explicit moment-matching computation methods, the moments $m_j(s_i)$, $j = 0, \ldots, \bar{j}_i - 1$, $i = 1, \ldots, \bar{i}$ (9) are computed, and the coefficients $b_{q-1}, \ldots, b_0$ and $a_q, \ldots, a_1$ of the approximate transfer function (5) are chosen in such a way that the moments $\hat{m}_j(s_i)$, $j = 0, \ldots, \bar{j}_i - 1$, $i = 1, \ldots, \bar{i}$ of the approximate transfer function will match those of the original transfer function. Examples of methods which compute the moments are the asymptotic waveform evaluation AWE [24], which is a single-point Padé approximation method, and the methods discussed in [4] which are multipoint Padé approximation methods.

The major disadvantage with computing the moments explicitly is that the sequence of vectors

$$\boldsymbol{v}_j(s_i) = ((\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b},$$

$j = 1, \ldots, \bar{j}_i - 1$, needed in the computations of the moments (9) will become more linearly dependent with increasing $\bar{j}_i$. The sequence of vectors $\boldsymbol{v}_j(s_i)$, $j = 1, \ldots, \infty$ will converge to the eigenvector corresponding to the dominant eigenvalue of $((\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C})^{-1}$. This is the power method; for example see [32]. Before the convergence of the eigenvector, the vectors $\boldsymbol{v}_j(s_i)$, $j = 0, \ldots, \bar{j}_i - 1$ will become nearly linearly dependent, and thus the explicit moment computation methods will suffer from numerical instabilities.

1.6 Krylov Subspace Methods

Instead of explicitly matching the moments, we want to use a numerically stable method to create a reduced-order model $\hat{\boldsymbol{d}}$, $\hat{\boldsymbol{G}}$, $\hat{\boldsymbol{C}}$ and $\hat{\boldsymbol{b}}$ in such a way that the approximate transfer function

$$\hat{H}(s) = \hat{\boldsymbol{d}}^T(\hat{\boldsymbol{G}} + s\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{b}}$$

matches as many moments around the interpolation points s_i, $i = 1, \ldots, \bar{i}$ as possible, given the restrictions of the method used.

The reduced-order model is computed by the rational Krylov method, which builds up an orthonormal basis $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}$ for a union of Krylov spaces

$$\mathcal{S}_{\bar{k}} = \bigcup_{i=1}^{\bar{i}} \mathcal{K}_{\bar{j}_i}((\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C}, (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}), \tag{10}$$

where as usual $\mathcal{K}_j(\boldsymbol{A}, \boldsymbol{b})$ stands for the Krylov space

$$\mathcal{K}_j(\boldsymbol{A}, \boldsymbol{b}) = \text{span}\{\boldsymbol{b}, \boldsymbol{A}\boldsymbol{b}, \boldsymbol{A}^2\boldsymbol{b}, \ldots, \boldsymbol{A}^{j-1}\boldsymbol{b}\}. \tag{11}$$

We do not need to compute the moments explicitly; it is enough to find an approximation $\hat{H}(s)$ that interpolates $H(s)$ to the appropriate order. The approximation $\hat{H}(s)$ is computed as follows. First an approximation $\tilde{\boldsymbol{x}}(s) \in \mathcal{S}_{\bar{k}}$ to the linear system of equations

$$(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{x}(s) = \boldsymbol{b} \tag{12}$$

is found and then $H(s)$ is approximated by $\hat{H}(s) = \boldsymbol{d}^T\tilde{\boldsymbol{x}}(s)$. From the solution procedure we can identify a reduced-order model $\hat{\boldsymbol{d}}$, $\hat{\boldsymbol{G}}$, $\hat{\boldsymbol{C}}$ and $\hat{\boldsymbol{b}}$. The reduced-order model of the main method discussed in this report matches the first $\bar{j}_i$

moments of the original model around the interpolation points s_i $i = 1, \ldots, \bar{i}$. The total number of moments matched is equal to the dimension of the subspace $\mathcal{S}_{\bar{k}}$ (10), and to the dimension of the reduced-order model

$$\bar{k} = \sum_{i=1}^{\bar{i}} \bar{j}_i.$$

This is the best one can hope for, given the basis (10), but it is half the number of moments matched by a multipoint Padé approximation of the same degree.

Since the maximum number of moments is not matched, the approximation will not be unique. In fact we will derive $\bar{i}$ different reduced-order models; they are equally good in that the transfer function will match the same number of moments around each interpolation point.

To match the maximum number of moments, in addition to building a basis for the subspace (10) one needs to build a basis for the left subspace

$$\mathcal{Z}_{\bar{k}} = \bigcup_{i=1}^{\bar{i}} \mathcal{K}_{\bar{j}_i}(\boldsymbol{C}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-H}, \boldsymbol{d}), \tag{13}$$

One-sided and two-sided methods use the same number of basis vectors to match the same number of moments.

An advantage with two-sided methods is that, by using a pair of biorthogonal bases, short-term recurrences can be developed; see [18] and [9].

One disadvantage with the biorthogonalisation process is that it is not as stable as the orthogonalisation process. The biorthogonalisation process can break down in a way that the orthogonalisation process cannot do; this can be remedied with look-ahead [16, 15, 37]. For an orthogonal basis the matrix $\boldsymbol{V}_{\bar{k}} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}]$ has optimal Euclidean condition number, $\mathrm{cond}(\boldsymbol{V}_{\bar{k}}) = 1$, whereas the condition numbers for the corresponding matrices for the biorthogonal basis have no upper bound.

Another possibility would be to build up two orthonormal bases for the subspaces $\mathcal{S}_{\bar{k}}$ (10) and $\mathcal{Z}_{\bar{k}}$ (13), similar to the two-sided Arnoldi algorithm [26]. This is more expensive than the biorthogonalisation process.

1.7 Related Work

A comprehensive treatment of Krylov subspace methods for model order reduction is made by Grimme in his PhD thesis [20]. An overview of Krylov subspace methods for model order reduction in circuit theory is given by Freund [11].

The first proof about the connection between the two-sided Lanczos and Padé for the single-input single-output was given by Gragg [19]. Feldmann and Freund introduced the Lanczos process in circuit simulation with the PVL algorithm ("Pade Via Lanczos") [8, 9], and about the same time this was done by Gallivan, Grimme and Van Dooren in [17]. Further development with the PVL algorithm by Feldmann and Freund includes small-signal circuit analysis and sensitivity computations [13], the symmetric PVL algorithm SyPVL [12], a block Lanczos algorithm [10], and a symmetric block Lanczos algorithm [14]. Grimme, Sorensen and Van Dooren describe how to generate a stable reduced-order model via an implicitly restarted Lanczos method [21]. Bai, Feldmann and Freund described how to generate a stable and passive reduced-order model via

the PVLπ algorithm [1]. Bai and Ye develop an error estimate for the transfer function computed by Padé via Lanczos [2].

The rational Lanczos algorithm discussed by Gallivan, Grimme and Van Dooren in their later work [18] is a multipoint Padé approximation; the method builds up a biorthogonal pair of bases. Multipoint Padé and Padé-type methods are also discussed by Grimme in his PhD thesis [20]. Nguyen and Li discuss a block rational Lanczos algorithm [22].

Silveria, Kamon, Elfadel and White create a passive reduced-order model through an L-orthogonal Arnoldi algorithm [33]; the algorithm is further discussed by Elfadel and Ling [7].

Odabasioglu, Celik and Pileggi[23] use the Arnoldi method to generate the basis, and congruence transformation to generate passive reduced-order models.

The Block Rational Arnoldi algorithm by Elfadel and Ling [6] is a multipoint approximation method. They basically use a block version of the algorithm discussed in this report; however, they use a different strategy to derive the reduced-order model. They use congruence transformation to generate passive reduced-order models.

2 The Rational Krylov Iteration

The rational Krylov algorithm was originally developed by Ruhe for eigenvalue computations [27, 28, 29, 30]. This version is adapted for model order reduction. In eigenvalue computations the first basis vector is often a normalised random vector; in model order reduction the choice of the first basis vector is given in the system (1), and here it is $\boldsymbol{v}_1 = (\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{b} / \parallel (\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{b} \parallel$.

Rational Krylov Algorithm

1 $\boldsymbol{r} = \boldsymbol{b}$
2 $k = 0$
3 **for** $i = 1 : \bar{i}$
4 **for** $j = 1 : \bar{j}_i$
5 $\boldsymbol{r} = (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{r}$
6 $[\boldsymbol{v}_{k+1}, \boldsymbol{f}_k] =$ **GramSchmidt** $(\boldsymbol{V}_k, \boldsymbol{r})$
7 $k = k + 1$
8 $\boldsymbol{r} = \boldsymbol{V}_k\boldsymbol{t}_k$ (choose continuation combination for next step)
9 $\boldsymbol{r} = \boldsymbol{C}\boldsymbol{r}$ (prepare for next)
10 **end**
11 $\hat{\boldsymbol{b}}_i = \boldsymbol{V}_k^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}$,
12 **end**
13 $\boldsymbol{r} = (\boldsymbol{G} + s_{\bar{i}}\,\boldsymbol{C})^{-1}\boldsymbol{r}$
14 $[\boldsymbol{v}_{k+1}, \boldsymbol{f}_k] =$ **GramSchmidt** $(\boldsymbol{V}_k, \boldsymbol{r})$

The right-hand sides of the reduced-order models $\hat{\boldsymbol{b}}_i$ are computed in line 11 of the algorithm; the reduced-order model is further discussed in section 5. The GramSchmidt function is just the standard Gram-Schmidt orthogonalisation; however, it is implemented with one reorthogonalisation.

Gram-Schmidt

```
1   function [v, f] = GramSchmidt(V, r)
2   if V = [] then
3       f_1 =|| r ||
4       v = r/f_1
5   else
6       f = V^H r (orthogonalise)
7       r = r − V f
8       k = length(f)
9       f_{k+1} =|| r ||
10      v = r/f_{k+1} (new basis vector)
11  end if
12  return
```

Set i_k to be the shift number at step k. Eliminate the intermediate vector $\boldsymbol{r}$ in the rational Krylov algorithm and we get

$$\boldsymbol{V}_{k+1}\boldsymbol{f}_k = (\boldsymbol{G} + s_{i_k}\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_k\boldsymbol{t}_k,\ 1 \le k \le \bar{k}.$$

Multiply from the left by $(\boldsymbol{G} + s_{i_k}\boldsymbol{C})$:

$$(\boldsymbol{G} + s_{i_k}\boldsymbol{C})\boldsymbol{V}_{k+1}\boldsymbol{f}_k = \boldsymbol{C}\boldsymbol{V}_k\boldsymbol{t}_k.$$

Put a zero at the bottom of $\boldsymbol{t}_k$ and rearrange the equation:

$$\boldsymbol{G}\boldsymbol{V}_{k+1}\boldsymbol{f}_k = \boldsymbol{C}\boldsymbol{V}_{k+1}(-s_{i_k}\boldsymbol{f}_k + \boldsymbol{t}_k). \tag{14}$$

Set $\boldsymbol{F}_{\bar{k}+1,\bar{k}} = [\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{\bar{k}}]$ and $\boldsymbol{T}_{\bar{k}+1,\bar{k}} = [\boldsymbol{t}_1, \ldots, \boldsymbol{t}_{\bar{k}}]$ with appropriate number of zeros added to the bottom of each $\boldsymbol{f}_k$ and $\boldsymbol{t}_k$. Join the columns $k = 1, \ldots, \bar{k}$ and get

$$\boldsymbol{G}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}_{\bar{k}+1,\bar{k}} = \boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{L}_{\bar{k}+1,\bar{k}}, \tag{15}$$

where

$$\boldsymbol{L}_{\bar{k}+1,\bar{k}} = -\boldsymbol{F}_{\bar{k}+1,\bar{k}}\mathrm{diag}(s_{i_k}) + \boldsymbol{T}_{\bar{k}+1,\bar{k}}. \tag{16}$$

We have represented the pencil $(\boldsymbol{G}, \boldsymbol{C})$ by the smaller pencil $(\boldsymbol{L}, \boldsymbol{F})$ in the basis $\boldsymbol{V}_{\bar{k}}$.

From now on we omit subscripts when submatrices are not taken, and let $\boldsymbol{F}$ and $\boldsymbol{L}$ stand for $\boldsymbol{F}_{\bar{k}+1,\bar{k}}$ and $\boldsymbol{L}_{\bar{k}+1,\bar{k}}$. Note that we have not included $\boldsymbol{f}_0$ in the relation (15).

Since the last row of the matrix $\boldsymbol{T}$ is zero, we get from (16)

$$l_{\bar{k}+1,\bar{k}} = -s_{\bar{\imath}}\, f_{\bar{k}+1,\bar{k}}. \tag{17}$$

3 Subspace

The rational Krylov algorithm generates basis vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}$ that span the subspace

$$\mathcal{K}_{\bar{j}_1}((\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{C}, (\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{b}) \bigcup$$
$$\bigcup_{i=2}^{\bar{\imath}} \mathcal{K}_{\bar{j}_i+1}((\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C}, (\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{b}).$$

Compare to (10) and see [28, 29, 30]. However, with the use of the lemma below another view can be taken.

Lemma 1. *If α and β are two non-identical interpolation points, then*

$$\begin{aligned}&(\boldsymbol{G}+\alpha\boldsymbol{C})^{-1}\boldsymbol{C}((\boldsymbol{G}+\beta\boldsymbol{C})^{-1}\boldsymbol{C})^{j-1}(\boldsymbol{G}+\beta\boldsymbol{C})^{-1}\boldsymbol{b}\\ &\in\ \text{span}\{(\boldsymbol{G}+\alpha\boldsymbol{C})^{-1}\boldsymbol{b}\}\bigcup\mathcal{K}_j((\boldsymbol{G}+\beta\boldsymbol{C})^{-1}\boldsymbol{C},(\boldsymbol{G}+\beta\boldsymbol{C})^{-1}\boldsymbol{b})\end{aligned}$$

The proof is given in [18].

From the lemma and the rational Krylov algorithm we can draw the conclusion that the basis vectors $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_{\bar{k}}$ span the subspace

$$\mathcal{S}_{\bar{k}}=\bigcup_{i=1}^{\bar{i}}\mathcal{K}_{\bar{j}_i}((\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C},(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}). \tag{18}$$

If we include the last vector $\boldsymbol{v}_{\bar{k}+1}$, the basis vectors $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_{\bar{k}+1}$ will span the subspace

$$\mathcal{S}_{\bar{k}+1}=\mathcal{S}_{\bar{k}}\bigcup\text{span}\{((\boldsymbol{G}+s_{\bar{i}}\,\boldsymbol{C})^{-1}\boldsymbol{C})^{\bar{j}_{\bar{i}}}\,(\boldsymbol{G}+s_{\bar{i}}\,\boldsymbol{C})^{-1}\boldsymbol{b}\}. \tag{19}$$

In our approach the approximate solution $\tilde{\boldsymbol{x}}(s)$ to the linear system of equations(12) will belong to the subspace (18), and under a certain condition the residual will be a scalar times $\boldsymbol{v}_{\bar{k}+1}$ and thus belong to the subspace (19); see section 4.

4 Approximate Solutions

4.1 General Introduction to Approximate Solutions

In this section we will describe how to use the rational Krylov algorithm to compute an approximate solution to the linear system of equations

$$(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{x}(s)=\boldsymbol{b}. \tag{20}$$

One of the solution methods described can be viewed as an oblique projection method, which solves the equivalent premultiplied linear system of equations

$$(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{x}(s)=(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}, \tag{21}$$

by using orthogonal projection. The other solution methods are closely related to projection methods.

Lemma 2. *If $\boldsymbol{V}_{\bar{k}+1}$, $\boldsymbol{L}$ and $\boldsymbol{F}$ are the results of the rational Krylov algorithm, then for any $\alpha,\beta\ \in\mathcal{C}$ we have*

$$(\boldsymbol{G}+\alpha\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+\beta\boldsymbol{F})=(\boldsymbol{G}+\beta\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+\alpha\boldsymbol{F}) \tag{22}$$

Proof: Rewrite equation (22) as

$$\begin{aligned}&\boldsymbol{G}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{L}+\alpha\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{L}+\beta\boldsymbol{G}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}+\alpha\beta\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}\\ =&\boldsymbol{G}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{L}+\alpha\boldsymbol{G}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}+\beta\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{L}+\alpha\beta\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}.\end{aligned} \tag{23}$$

By using (15) we see that equation (22) holds. □

To see how to calculate the approximate solution, set $\alpha = s$ and $\beta = s_i$:

$$(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s_i\boldsymbol{F}) = (\boldsymbol{G} + s_i\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s\boldsymbol{F}).$$

Multiply from the left by $(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}$:

$$(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s_i\boldsymbol{F}) = \boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s\boldsymbol{F}). \tag{24}$$

The point $s_i,\ 1 \leq i \leq \bar{i}$ is an interpolation point used in the rational Krylov algorithm. The point s is an arbitrary point in the complex plane. Make the approximation $\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s_i\boldsymbol{F}) \simeq \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})$; from (17) we see that $(l_{\bar{k}+1,\bar{k}} + s_{\bar{i}}\, f_{\bar{k}+1,\bar{k}}) = 0$ and thus equality holds for $s_i = s_{\bar{i}}$. Multiply (24) from the right by $(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$:

$$(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{V}_{\bar{k}} \simeq \boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s\boldsymbol{F})(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1} \tag{25}$$

where equality holds for $s_i = s_{\bar{i}}$.

4.2 Approximate Solutions

From (25) and (21) we see that one way of calculating the approximate solution to the linear system (20) is to solve the $\bar{k} \times \bar{k}$ system

$$(\boldsymbol{L}_{\bar{k},\bar{k}} + s\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s) = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b},\ 1 \leq i \leq \bar{i} \tag{26}$$

for $\boldsymbol{y}_i(s)$ and take

$$\tilde{\boldsymbol{x}}_i(s) = \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s),\ 1 \leq i \leq \bar{i} \tag{27}$$

as the approximate solution. The point $s_i,\ 1 \leq i \leq \bar{i}$ is an interpolation point.

To simplify the notation in the following, set

$$\hat{\boldsymbol{b}}_i = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}.$$

This is the right-hand side of the reduced-order model; it is calculated in line 11 of the rational Krylov algorithm. The reduced-order model will be further discussed in section 5.

The right-hand side of the premultiplied system (21) belongs to the subspace (18), so it can be expressed as a linear combination of the basis vectors

$$(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b} = \boldsymbol{V}_{\bar{k}}\hat{\boldsymbol{b}}_i. \tag{28}$$

Note that we have $\bar{i}$ different approximate solutions; they are equally good in the sense that the corresponding transfer functions match the same number of moments around each interpolation point $s_i,\ 1 \leq i \leq \bar{i}$. This is discussed in section 5.

From (25) we see that for $s_i = s_{\bar{i}}$ an approximate solution is found by using orthogonal projection of the premultiplied linear system of equations (21) onto the subspace spanned by the basis vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}$.

4.3 Residuals

We will derive an expression first for the residual to the premultiplied linear system of equations (21) and then for the original linear system of equations (20), where the approximate solution is calculated by (26) and (27).

Rewrite the relation (24) as

$$\begin{aligned}&(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})(\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})+\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+s_i f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T)\\&=\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s\boldsymbol{F}_{\bar{k},\bar{k}})+\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+sf_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T\end{aligned}\tag{29}$$

To simplify the notation, set

$$\rho_i(s)=f_{\bar{k}+1,\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s).\tag{30}$$

Solve the approximate problem (26),(27) and the residual of the premultiplied linear system of equations (21) will be

$$\begin{aligned}\boldsymbol{r}_i(s)=&(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}-(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\tilde{\boldsymbol{x}}_i(s)\\=&\boldsymbol{V}_{\bar{k}}\hat{\boldsymbol{b}}_i-(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s)\\=&\boldsymbol{V}_{\bar{k}}\hat{\boldsymbol{b}}_i-(\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s)+\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+sf_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s)\\&-(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+s_if_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s))\\=&(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+s_if_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s)\\&-\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+sf_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s)\\=&(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(s_i-s_{\bar{i}})\rho_i(s)\\&-\boldsymbol{v}_{\bar{k}+1}(s-s_{\bar{i}})\rho_i(s)\end{aligned}\tag{31}$$

where $\boldsymbol{y}_i(s)$ is the solution to (26). The second equality follows from (27) and (28). The third equality follows from (29), the fourth from (26) and the fifth from (17) and (30).

The quantity $\rho_i(s)$ will be small if $f_{\bar{k}+1,\bar{k}}$ is small; in that case we have a nearly invariant subspace and we will get a good approximation for the whole complex plane. If the solution has converged in some regions in the complex plane, the quantity $\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s)$ will be small in those regions.

Note that for $s_i=s_{\bar{i}}$ we get a simple expression for the residual:

$$\boldsymbol{r}_{\bar{i}}(s)=-\boldsymbol{v}_{\bar{k}+1}(s-s_{\bar{i}})\rho_i(s).\tag{32}$$

Multiplying (31) by $(\boldsymbol{G}+s_i\boldsymbol{C})$, the residual of the original system (20) is

$$\begin{aligned}\boldsymbol{r}(s)=&\boldsymbol{b}-(\boldsymbol{G}+s\boldsymbol{C})\tilde{\boldsymbol{x}}_i(s)\\=&(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(s_i-s_{\bar{i}})\rho_i(s)\\&-(\boldsymbol{G}+s_i\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(s-s_{\bar{i}})\rho_i(s).\end{aligned}$$

Corollary 1. *If the approximate solution $\tilde{\boldsymbol{x}}_i(s)$, $1\le i\le\bar{i}$ is calculated by (26) and (27) at the interpolation points $s=s_q$, $q=1,\ldots,\bar{i}$, then the residual at each interpolation point s_q*

$$\begin{aligned}\boldsymbol{r}_i(s_q)&=(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}-(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s_q\boldsymbol{C})\tilde{\boldsymbol{x}}_i(s_q)\\&=0.\end{aligned}$$

The proof is given in section 6.

5 Reduced-Order Model

5.1 Transfer Function

If we use the approximate solution (26) and (27) derived in the previous section, we can approximate the transfer function

$$H(s) = \boldsymbol{d}^T(\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{b} \tag{33}$$

by

$$\begin{aligned}\hat{H}_i(s) &= \boldsymbol{d}^T\tilde{\boldsymbol{x}}_i(s) \\ &= \boldsymbol{d}^T\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}} + s\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}.\end{aligned} \tag{34}$$

Since we can choose $\bar{i}$ different approximate solutions $\boldsymbol{x}_i(s)$, $1 \le i \le \bar{i}$ (27) we will get $\bar{i}$ different transfer functions $\hat{H}_i(s)$, $1 \le i \le \bar{i}$. In theorem 1, we will prove that the $\bar{i}$ different transfer functions will all match the same number of moments around each interpolation point $s_q, i = 1, \ldots, \bar{i}$ of the original model.

For a given point s in the complex plane, which interpolation point s_i should be chosen in the approximate transfer function (34)? One strategy would be to choose the interpolation point s_i which is closest to s, and thus use different transfer functions in different regions of the complex plane. On the other hand, the simple form of the residuals (32) suggests that $s_{\bar{i}}$ could be a suitable interpolation point, giving us the same transfer function in the entire region.

5.2 Reduced-Order Model

From (33) and (34) we suggest the reduced model $\hat{\boldsymbol{b}}_i = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}$, $\hat{\boldsymbol{d}}_i^T = \boldsymbol{d}^T\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})$, $\hat{\boldsymbol{G}} = \boldsymbol{L}_{\bar{k},\bar{k}}$ and $\hat{\boldsymbol{C}} = \boldsymbol{F}_{\bar{k},\bar{k}}$. The approximate transfer function $\hat{H}_i(s)$ can be written as

$$\hat{H}_i(s) = \hat{\boldsymbol{d}}_i^T(\hat{\boldsymbol{G}} + s\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{b}}_i$$

In the theorem below, the letter i, $1 \le i \le \bar{i}$ corresponds to a particular choice of the approximate solution $\boldsymbol{x}_i(s)$ (27) and its corresponding transfer function $\hat{H}_i(s)$ (34). The letter q corresponds to a particular expansion point s_q, $1 \le q \le \bar{i}$, one of the interpolation points.

Theorem 1. *Let the j^{th} moments of the original and reduced-order systems about the interpolation point s_q be $m_j(s_q) = (-1)^j\boldsymbol{d}^T((\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ and $\hat{m}_{j,i}(s_q) = (-1)^j\hat{\boldsymbol{d}}_i^T((\hat{\boldsymbol{G}} + s_q\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{C}})^j(\hat{\boldsymbol{G}} + s_q\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{b}}_i$ respectively. If $\hat{\boldsymbol{b}}_i = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b}$, $\hat{\boldsymbol{d}}_i^T = \boldsymbol{d}^T\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})$, $\hat{\boldsymbol{G}} = \boldsymbol{L}_{\bar{k},\bar{k}}$ and $\hat{\boldsymbol{C}} = \boldsymbol{F}_{\bar{k},\bar{k}}$ where $\boldsymbol{L}$, $\boldsymbol{F}$ and $\boldsymbol{V}_{\bar{k}+1}$ are the result of the rational Krylov algorithm, then $m_j(s_q) = \hat{m}_{j,i}(s_q)$ for $i = 1, \ldots, \bar{i}$, $q = 1, \ldots, \bar{i}$, $j = 0, \ldots, \bar{j}_q - 1$.*

Before we prove the theorem we will prove some lemmas needed in the proof.

Lemma 3. *If $\boldsymbol{V}_{\bar{k}+1}$, $\boldsymbol{L}$ and $\boldsymbol{F}$ are the results of the rational Krylov algorithm, then*

$$\begin{aligned}&\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{V}_{\bar{k}} = (\boldsymbol{L}_{\bar{k},\bar{k}} + s\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1} \\ &- \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}} + s_i f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}.\end{aligned}$$

Proof: Multiply (24) from the left by $\boldsymbol{V}_{\bar{k}}^H$ and from the right by $(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$ and rearrange the equation. □

The following lemma can be proved directly from lemma 3 by letting $s \to \infty$. We state the lemma and provide a separate proof, as we will refer to the lemma and the proof later on.

Lemma 4. *If $\boldsymbol{V}_{\bar{k}+1}$, $\boldsymbol{L}$ and $\boldsymbol{F}$ are the results of the rational Krylov algorithm, then*

$$\begin{aligned}&\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}} = \boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\\ &-\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{k+1}(l_{\bar{k}+1,\bar{k}}+s_i f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}.\end{aligned} \tag{35}$$

Proof: Add $s_i\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}$ to equation (15), and we get

$$(\boldsymbol{G}+s_i\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F} = \boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s_i\boldsymbol{F}). \tag{36}$$

Multiply from the left by $\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}$:

$$\boldsymbol{F}_{\bar{k},\bar{k}} = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s_i\boldsymbol{F}).$$

Multiply from the right by $(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$ and rearrange the equation, and we get equation (35). □

Lemma 5. *If $\boldsymbol{V}_{\bar{k}+1}$, $\boldsymbol{L}$ and $\boldsymbol{F}$ are the results of the rational Krylov algorithm, then*

$$\begin{aligned}&\boldsymbol{e}_{\bar{k}}^T(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H((\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b} = 0\\ &i = 1,\ldots,\bar{\imath}\quad j = 0,\ldots,\bar{\jmath}_i - 2.\end{aligned} \tag{37}$$

Proof: From the rational Krylov algorithm and section 3 we see that

$$\begin{aligned}&((\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b} \in \operatorname{span}\{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_{p_{i,j}}\}\\ &p_{i,j} = j+1+\sum_{r=1}^{i-1}\bar{\jmath}_r,\ i = 1,\ldots\bar{\imath}, j = 0,\ldots,\bar{\jmath}_i - 1.\end{aligned}$$

From the relation (16) we see that the matrix $\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}}$ is triangular in the columns:

$$k = \begin{cases}1,\ldots,\bar{\jmath}_1 - 1 & \text{if } i = 1\\ \sum_{r=1}^{i-1}\bar{\jmath}_r,\ldots,(\sum_{r=1}^{i}\bar{\jmath}_r) - 1 & \text{if } i \neq 1 \text{ and } i \neq \bar{\imath}\\ \sum_{r=1}^{\bar{\imath}-1}\bar{\jmath}_r,\ldots,(\sum_{r=1}^{\bar{\imath}}\bar{\jmath}_r) & \text{if } i = \bar{\imath}\end{cases} \tag{38}$$

The rest of $\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}}$ has Hessenberg structure. This means that

$$((\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b} = \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}$$

where $\boldsymbol{y}$ is the solution to

$$(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y} = \boldsymbol{V}_{\bar{k}}^H((\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$$

and further

$$y_k = 0,\quad k = j+2+\sum_{r=1}^{i-1}\bar{\jmath}_r,\ldots,\bar{k}\quad j = 0,\ldots,\bar{\jmath}_i - 2$$

and thus equation (37) holds. □

Lemma 6. *If $\boldsymbol{V}_{\bar{k}+1}$, $\boldsymbol{L}$ and $\boldsymbol{F}$ are the results of the rational Krylov algorithm, then for $i \neq q$*

$$\boldsymbol{e}_{\bar{k}}^T(\boldsymbol{L}_{\bar{k},\bar{k}} + s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b} = 0. \tag{39}$$

Proof:

The case $i > q$. Set $s = s_q$ and rewrite equation (24) as

$$\begin{aligned}
&(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s_q\boldsymbol{C})\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}}) \\
&+ (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s_q\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}} + s_i f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T \\
&= \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_q\boldsymbol{F}_{\bar{k},\bar{k}}) + \boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}} + s_q f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T.
\end{aligned} \tag{40}$$

From the rational Krylov algorithm and section 3 we see that

$$\begin{aligned}
&(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b} \in \text{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{p_q}\} \\
&p_q = \sum_{r=1}^{q-1} \bar{j}_r + 1.
\end{aligned} \tag{41}$$

Further, $(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ can be written as

$$(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b} = \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}, \tag{42}$$

where $\boldsymbol{y}$ is the solution to

$$(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y} = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b}.$$

From (16) we see that the matrix $\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}}$ is triangular in the columns (38). The rest of $\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}}$ has Hessenberg structure. Now we have $i > q$ and thus the index p_q in (41) is less than the lowest column index of the triangular part of $\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}}$ (38). This means that the elements $y_k = 0, \quad k = \sum_{r=1}^{i-1} \bar{j}_r + 2, \ldots, \bar{k}$ are equal to zero. Multiply equation (40) with the vector $\boldsymbol{y}$ and replace $\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}$ with $(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b}$, and we get the relation

$$(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b} = \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_q\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}. \tag{43}$$

Put the relation above into (39) and the proof is complete for the case $i > q$.

The case $i < q$. Using similar arguments as for equation (42) we get the relation (43) and at least the last element of $\boldsymbol{y}$ is zero. Substitute equation (43) into (39), and the proof is complete for the case $i < q$. □

The relation (39) can be proved to hold for $i = q$ and $\bar{j}_i > 1$.

Proof of Theorem 1. From the rational Krylov algorithm and section 3 we have

$$\begin{aligned}
&((\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b} \in \text{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}\} \\
&q = 1, \ldots, \bar{\imath}, \ j = 0, \ldots, \bar{j}_q - 1.
\end{aligned} \tag{44}$$

With equation (44), lemma 4 and lemma 5 we can express $((\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ for $q = 1, \ldots, \bar{\imath}, \ j = 1, \ldots, \bar{j}_q - 1$ as

$$\begin{aligned}
&((\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b} \\
&= \boldsymbol{V}_{\bar{k}}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}}\boldsymbol{V}_{\bar{k}}^H((\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C})^{j-1}(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b} \\
&= \boldsymbol{V}_{\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H((\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{C})^{j-1}(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b} \\
&= \boldsymbol{V}_{\bar{k}}(\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1})^j\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_q\boldsymbol{C})^{-1}\boldsymbol{b}
\end{aligned} \tag{45}$$

The first equality follows from (44), the second from lemma 4 and lemma 5. The third equality follows from repeated application of lemma 4, lemma 5 and (44). Let us express $\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ for $q=1,\ldots,\bar{i}$, $i=1,\ldots,\bar{i}$, $i\neq q$ as

$$\begin{aligned}
&\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b}\\
&=\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}(\boldsymbol{G}+s_i\boldsymbol{C})(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}\\
&=\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}(\boldsymbol{G}+s_i\boldsymbol{C})\boldsymbol{V}_{\bar{k}}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}\\
&=(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}
\end{aligned}\tag{46}$$

The second equality follows from (44). The third equality follows from lemma 3 and lemma 6. If $i=q$ then multiply $\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$ by the unit matrix

$$\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}=(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}.\tag{47}$$

If we put (45), (46) and (47) together we can express $((\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ for $q=1,\ldots,\bar{i}$, $i=1,\ldots,\bar{i}$, $j=1,\ldots,\bar{j}_q-1$ as

$$\begin{aligned}
&((\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b}\\
&=\boldsymbol{V}_{\bar{k}}(\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1})^j(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i\\
&=\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}+s_q\boldsymbol{I})^{-j}(\boldsymbol{L}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}+s_i\boldsymbol{I})\boldsymbol{F}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}(\boldsymbol{L}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}+s_q\boldsymbol{I})^{-1}\hat{\boldsymbol{b}}_i\\
&=\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}+s_i\boldsymbol{I})(\boldsymbol{L}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}+s_q\boldsymbol{I})^{-j}(\boldsymbol{L}_{\bar{k},\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}+s_q\boldsymbol{I})^{-1}\hat{\boldsymbol{b}}_i\\
&=\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}(\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1})^j\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i\\
&=\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})((\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{F}_{\bar{k},\bar{k}})^j(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i
\end{aligned}\tag{48}$$

The third equality is a consequence of the fact that rational and polynomial functions of the same matrix commute. By taking $\hat{\boldsymbol{b}}_i=\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$, $\hat{\boldsymbol{d}}_i^T=\boldsymbol{d}^T\boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})$, $\hat{\boldsymbol{G}}=\boldsymbol{L}_{\bar{k},\bar{k}}$ and $\hat{\boldsymbol{C}}=\boldsymbol{F}_{\bar{k},\bar{k}}$ as the reduced-order model, we see that the moments of the original and reduced-order systems about the interpolation point s_q are equal for $q=1,\ldots,\bar{i}$, $i=1,\ldots,\bar{i}$, $j=1,\ldots,\bar{j}_q-1$. □

6 Errors

In this section we will derive error expressions. They are not computable by quantities of the reduced-order model, but they give insight into the approximations, and into which quantities need to be estimated in order to get an error estimate. The error estimates are further discussed in section 10.

Theorem 2. *If $\hat{H}_i(s)$ is the transfer function of the reduced-order model defined in theorem 1, and the approximate solution $\tilde{\boldsymbol{x}}_i(s)$ to the premultiplied linear system of equations*

$$(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{x}(s)=(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$$

is given by

$$\tilde{\boldsymbol{x}}_i(s) = \boldsymbol{V}_{\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}} + s_i\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s) \tag{49}$$

where $\boldsymbol{y}_i(s)$ *is the solution to*

$$(\boldsymbol{L}_{\bar{k},\bar{k}} + s\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s) = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b},$$

and $\rho_i(s)$ *is defined by*

$$\rho_i(s) = f_{\bar{k}+1,\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s),$$

then

$$\begin{aligned}\Delta\tilde{\boldsymbol{x}}_i(s) =& \boldsymbol{x}(s) - \tilde{\boldsymbol{x}}_i(s)\\ =&(\boldsymbol{v}_{\bar{k}+1} + (\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1}(s_{\bar{\imath}} - s))(s_i - s)\rho_i(s)\end{aligned} \tag{50}$$

the error and the derivatives of the error at the interpolation points s_q, $q = 1,\ldots,\bar{\imath}$ *become zero:*

$$\frac{\partial^j\Delta\tilde{\boldsymbol{x}}_i(s)}{\partial s^j}\Big|_{s=s_q} = 0,\ j = 0,\ldots,\bar{j}_i - 1 \tag{51}$$

and the error for the approximate transfer function is

$$\begin{aligned}\epsilon(s) &= H(s) - \hat{H}(s)\\ &= \boldsymbol{d}^T\Delta\tilde{\boldsymbol{x}}_i(s)\end{aligned}$$

Proof:

$$\begin{aligned}\boldsymbol{x}(s) - \tilde{\boldsymbol{x}}_i(s) =&(\boldsymbol{G} + s\boldsymbol{C})^{-1}(\boldsymbol{G} + s_i\boldsymbol{C})\boldsymbol{r}_i(s)\\ =&\boldsymbol{v}_{\bar{k}+1}(s_i - s_{\bar{\imath}})\rho_i(s)\\ &-(\boldsymbol{G} + s\boldsymbol{C})^{-1}(\boldsymbol{G} + s_i\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(s - s_{\bar{\imath}})\rho_i(s)\end{aligned} \tag{52}$$

where $\boldsymbol{r}_i(s)$ is the residual. The second equality follows from the expression of the residual $\boldsymbol{r}_i(s)$ (31). We get (50) by using the relation

$$(\boldsymbol{G} + s\boldsymbol{C})^{-1}(\boldsymbol{G} + s_i\boldsymbol{C}) = \boldsymbol{I} + (s_i - s)(\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}.$$

By using theorem 1 we have

$$\frac{\partial^j\Delta\epsilon(s)}{\partial s^j}\Big|_{s=s_q} = 0,\ j = 0,\ldots,\bar{j}_i - 1 \tag{53}$$

where $\epsilon(s) = H(s) - \hat{H}(s) = \boldsymbol{d}^T\Delta\tilde{\boldsymbol{x}}_i(s)$. Now $\boldsymbol{d}$ is not used when the basis is built up; and therefore (53) must hold for $\boldsymbol{d}$ chosen arbitrarily, and thus (51) holds.

The error for the transfer function of the reduced-order model follows from the error of $\tilde{\boldsymbol{x}}_i(s)$ (52) and from the expressions for the transfer function $H(s)$ (3) and the expression for the transfer function of the reduced-order model $\hat{H}(s)$ (34). □

In order to use theorem 2 in an error estimate, we need to estimate $\| (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|$ in a good and efficient way. This is discussed in section 10.

Proof of corollary 1 in section 4.3: The proof follows directly from (51) of theorem 2.

In theorem 3 we develop an expression for the error of the first non-matching moment of the reduced-order model.

Theorem 3. *If* $\hat{m}_{\bar{j}_q,i}(s_q)$, $q = 1, \ldots, \bar{i}$, $i = 1, \ldots, \bar{i}$ *are the moments of the reduced-order model defined in theorem 1, then*

$$\begin{aligned}\epsilon(s_q) =& m_{\bar{j}_q}(s_q) - \hat{m}_{\bar{j}_q,i}(s_q) \\ =&(-1)^{\bar{j}_q}\boldsymbol{d}^T(\boldsymbol{v}_{\bar{k}+1}f_{\bar{k}+1,\bar{k}}\boldsymbol{e}_{\bar{k}}^T - (\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+s_q f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T) \\ &(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})((\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{F}_{\bar{k},\bar{k}})^{\bar{j}_q-1} \\ &(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i.\end{aligned} \tag{54}$$

Proof: From (48) we have

$$\begin{aligned}m_{\bar{j}_q-1,i}(s_q) =&(-1)^{\bar{j}_q-1}\boldsymbol{d}^T((\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C})^{\bar{j}_q-1}(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b} \\ =&(-1)^{\bar{j}_q-1}\boldsymbol{d}^T\boldsymbol{V}_{\bar{k}}(\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1})^{\bar{j}_q-1}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}}) \\ &(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i \\ =&\hat{m}_{\bar{j}_q-1,i}(s_q),\end{aligned} \tag{55}$$

and

$$\begin{aligned}&\hat{m}_{\bar{j}_q,i}(s_q) = \\ &(-1)^{\bar{j}_q}\boldsymbol{d}^T\boldsymbol{V}_{\bar{k}}(\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1})^{\bar{j}_q}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i\end{aligned} \tag{56}$$

The vector $((\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ belongs to the subspace spanned by the basis vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}$ for $j = \bar{j}_q - 1$, but not for $j = \bar{j}_q$.

Replace s_i by s_q in (36), multiply from the right by $(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$ and from the left by $(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}$, and rearrange the equation:

$$\begin{aligned}&(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}} = \boldsymbol{V}_{\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1} \\ &+(\boldsymbol{v}_{\bar{k}+1}f_{\bar{k}+1,\bar{k}}\boldsymbol{e}_{\bar{k}}^T - (\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}}+s_q f_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T)(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\end{aligned} \tag{57}$$

By using (55) and (56) we can express the error of the moments as

$$\begin{aligned}m_{\bar{j}_q}(s_q) - \hat{m}_{\bar{j}_q,i}(s_q) = (-1)^{\bar{j}_q}\boldsymbol{d}^T((\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}} - \boldsymbol{V}_{\bar{k}}\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}) \\ (\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1})^{\bar{j}_q-1}(\boldsymbol{L}_{\bar{k},\bar{k}}+s_i\boldsymbol{F}_{\bar{k},\bar{k}})(\boldsymbol{L}_{\bar{k},\bar{k}}+s_q\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\hat{\boldsymbol{b}}_i.\end{aligned}$$

Substitute (57) into the equation above and use (48), and we get (54). □

7 Alternative Way of Calculating the Approximate Solution

7.1 Introduction

In section 4.3 we get a simple expression of the residual (31), only for $i = \bar{i}$, without the matrices $(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}$ and $(\boldsymbol{G}+s_i\boldsymbol{C})$.

In this section we will develop $\bar{i}$ methods such that for each i, $1 \le i \le \bar{i}$ gets a simple expression of the residual without the matrices $(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}$ and $(\boldsymbol{G}+s_i\boldsymbol{C})$. However, the methods will not match the maximum number of moments given the information in the basis, except for $i = \bar{i}$.

7.2 Approximate Solution

Now we solve the equation (26) for $\boldsymbol{y}_i(s)$ and take

$$\tilde{\boldsymbol{x}}_i(s) = \boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s_i\boldsymbol{F})\boldsymbol{y}_i(s)$$

as the approximate solution. It differs from the solution $\tilde{\boldsymbol{x}}_i(s)$ (27) of section (4) only in that we have retained the last row of $(\boldsymbol{L} + s_i\boldsymbol{F})$.

From (17) we have $(l_{\bar{k}+1,\bar{k}} + s_{\bar{\imath}}\, f_{\bar{k}+1,\bar{k}}) = 0$ and thus (27) and (7.2) are equivalent for $s_i = s_{\bar{\imath}}$. For $s_i = s_{\bar{\imath}}$ an approximate solution is found by using orthogonal projection of the system (21) onto the subspace spanned by the basis vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}$.

For $i \neq \bar{\imath}$, the solution (27) from the previous discussion belongs to the subspace span$(\boldsymbol{V}_{\bar{k}})$, while the alternative solution (7.2) is taken from the larger subspace span$(\boldsymbol{V}_{\bar{k}+1})$.

7.3 Residuals

We will derive an expression first for the residual to the premultiplied linear system of equations (21) and then for the original linear system of equations (20), where the approximate solution is calculated by (26) and (7.2). The residual of the premultiplied system (21) is

$$\begin{aligned}
\boldsymbol{r}_i(s) =& (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b} - (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s\boldsymbol{C})\tilde{\boldsymbol{x}}_i(s) \\
=& \boldsymbol{V}_{\bar{k}+1}\hat{\boldsymbol{b}}_i - (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s\boldsymbol{C})\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L} + s_i\boldsymbol{F})\boldsymbol{y}_i(s) \\
=& \boldsymbol{V}_{\bar{k}+1}(\hat{\boldsymbol{b}_i} - (\boldsymbol{L} + s\boldsymbol{F})\boldsymbol{y}_i(s)) \\
=& -\boldsymbol{v}_{\bar{k}+1}(l_{\bar{k}+1,\bar{k}} + sf_{\bar{k}+1,\bar{k}})\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s) \\
=& -\boldsymbol{v}_{\bar{k}+1}(s - s_{\bar{\imath}})\rho_i(s),
\end{aligned} \tag{58}$$

where $\boldsymbol{y}_i(s)$ is the solution to (26). The second equality follows from (7.2) and (28), the third equality from (24), the fourth from (26), and the fifth from (17) and (30).

By multiplying (58) by$(\boldsymbol{G} + s_i\boldsymbol{C})$ we see that the residual for the original system (20) is

$$\begin{aligned}
\boldsymbol{r}_i(s) =& \boldsymbol{b} - (\boldsymbol{G} + s\boldsymbol{C})\tilde{\boldsymbol{x}}_i(s) \\
=& -(\boldsymbol{G} + s_i\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(s - s_{\bar{\imath}})\rho_i(s)
\end{aligned}$$

Corollary 2. *If the approximate solution $\tilde{\boldsymbol{x}}_i(s)$, $1 \leq i \leq \bar{\imath}$ is calculated by (26) and (7.2), then the residual at each interpolation point s_q is*

$$\begin{aligned}
\boldsymbol{r}_i(s_q) &= (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{b} - (\boldsymbol{G} + s_i\boldsymbol{C})^{-1}(\boldsymbol{G} + s_q\boldsymbol{C})\tilde{\boldsymbol{x}}(s_q) \\
&= \begin{cases} 0 & q \neq i \\ 0 & q = i \text{ only for } \bar{\jmath}_i > 1 \end{cases}
\end{aligned}$$

The corollary can be proved in a similar way as Corollary 1.

7.4 Transfer Function

From (26) and (7.2) we see that the transfer function (33) can be approximated by

$$\begin{aligned}\hat{H}_i(s) &= \boldsymbol{d}^T \tilde{\boldsymbol{x}}_i(s) \\ &= \boldsymbol{d}^T \boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s_i\boldsymbol{F})(\boldsymbol{L}_{\bar{k},\bar{k}}+s\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}\boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}.\end{aligned} \tag{59}$$

7.5 Reduced-Order Model

From (33) and (59) we suggest the reduced model $\hat{\boldsymbol{b}}_i = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$, $\hat{\boldsymbol{d}}_i^T = \boldsymbol{d}^T\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s_i\boldsymbol{F})$, $\hat{\boldsymbol{G}} = \boldsymbol{L}_{\bar{k},\bar{k}}$ and $\hat{\boldsymbol{C}} = \boldsymbol{F}_{\bar{k},\bar{k}}$. The approximate transfer function $\hat{H}_i(s)$ can be written as

$$\hat{H}_i(s) = \hat{\boldsymbol{d}}_i^T(\hat{\boldsymbol{G}}+s\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{b}}_i.$$

Theorem 4. *Let the j^{th} moments of the original and reduced-order systems about the interpolation point s_q be $m_j(s_q) = (-1)^j\boldsymbol{d}^T((\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{C})^j(\boldsymbol{G}+s_q\boldsymbol{C})^{-1}\boldsymbol{b}$ and $\hat{m}_{j,i}(s_q) = (-1)^j\hat{\boldsymbol{d}}_i^T((\hat{\boldsymbol{G}}+s_q\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{C}})^j(\hat{\boldsymbol{G}}+s_q\hat{\boldsymbol{C}})^{-1}\hat{\boldsymbol{b}}_i$ respectively. If $\hat{\boldsymbol{b}}_i = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$, $\hat{\boldsymbol{d}}_i^T = \boldsymbol{d}^T\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s_i\boldsymbol{F})$, $\hat{\boldsymbol{G}} = \boldsymbol{L}_{\bar{k},\bar{k}}$ and $\hat{\boldsymbol{C}} = \boldsymbol{F}_{\bar{k},\bar{k}}$ where $\boldsymbol{L}_{\bar{k},\bar{k}}$, $\boldsymbol{F}_{\bar{k},\bar{k}}$ and $\boldsymbol{V}_{\bar{k}+1}$ are the result of the rational Krylov algorithm, then $m_j(s_q) = \hat{m}_{j,i}(s_q)$ for $i \neq q$, $j = 1,\ldots,\bar{j}_q - 1$ and for $i = q$, $j = 1,\ldots,\bar{j}_q - 2$.*

Proof: The theorem can be proved by using similar techniques as in the proof of theorem 1

From theorem 1 and section 3 we find that the reduced-order model of theorem 1 matches as many moments of the original model as one can hope for, whereas from theorem 4 and section 3 we see that the reduced model of theorem 4 matches one moment less than possible when $i = q$, i.e. when the expansion point s_q is used in the preconditioner $(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}$.

7.6 Errors

Theorem 5. *If $\hat{H}_i(s)$ is the transfer function of the reduced-order model defined in theorem 4, and the approximate solution $\tilde{\boldsymbol{x}}_i(s)$ to the linear system of equations*

$$(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}(\boldsymbol{G}+s\boldsymbol{C})\boldsymbol{x}(s) = (\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$$

is given by

$$\tilde{\boldsymbol{x}}_i(s) = \boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s_i\boldsymbol{F})\boldsymbol{y}_i(s) \tag{60}$$

where $\boldsymbol{y}_i(s)$ is the solution to

$$(\boldsymbol{L}_{\bar{k},\bar{k}}+s\boldsymbol{F}_{\bar{k},\bar{k}})\boldsymbol{y}_i(s) = \boldsymbol{V}_{\bar{k}}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b},$$

and $\rho_i(s)$ is defined by

$$\rho_i(s) = f_{\bar{k}+1,\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{y}_i(s),$$

then

$$\begin{aligned}\Delta\tilde{\boldsymbol{x}}_i(s) =& \boldsymbol{x}(s) - \tilde{\boldsymbol{x}}_i(s) \\ =& (\boldsymbol{I} + (s_i - s)(\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C})\boldsymbol{v}_{\bar{k}+1}(s_{\bar{i}} - s)\rho_i(s)\end{aligned}$$

The error and its derivatives at the interpolation points $s_q, q = 1, \ldots, \bar{i}$ *become zero for* $i \neq q$, $j = 1, \ldots, \bar{j}_q - 1$ *and for* $i = q$, $j = 1, \ldots, \bar{j}_q - 2$*:*

$$\frac{\partial^j \Delta\tilde{\boldsymbol{x}}_i(s)}{\partial s^j}\Big|_{s=s_q} = 0,$$

and the error of the approximate transfer function is

$$\begin{aligned}\epsilon(s) &= H(s) - \hat{H}_i(s) \\ &= \boldsymbol{d}^T \Delta\tilde{\boldsymbol{x}}_i(s)\end{aligned}$$

The proof of theorem 5 can be done in a similar way as the proof of theorem 2.

8 Stability and Passivity

It is often important that the reduced-order model inherit properties of the original dynamic system; two such properties are stability and passivity.

A system is *stable* if the output $y(t)$ is limited for a limited input $u(t)$. A system is *passive* if it does not generate energy internally. A passive system is stable. The connection of two passive systems is a passive system, but the connection of two stable systems is not necessarily a stable system.

The system (1) is stable if all poles of the transfer function (3) are in the left half of the complex plane. Recall that the poles of (3) are equivalent to the eigenvalues of (7), as was shown in section 1.3. The stability of the reduced-order model can be checked by calculating the corresponding eigenproblem. If the reduced-order model of a stable dynamic system is not stable, then by creating a new reduced-order model in a new basis of lower dimension than the old, in such a way that unwanted eigenvalues are discarded. This can be done by using implicit restarts similar to [21], and by using techniques developed in [35] and [5]. Further work needs to be done before this can be used successfully in connection with the reduced-order models discussed in this report.

A single-input single-output (SISO) system is *passive*, if the following conditions are fulfilled:

1. $H(s^H) = H^H(s)$
2. $H(s) + H^H(s) \geq 0$, $\mathrm{Re}(s) > 0$

A RLC network is a passive network. In the case of a RLC network, the state vector and matrices are partitioned as follows by the Modified Nodal Analysis (MNA):

$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{v} \\ \boldsymbol{i}\end{bmatrix}, \ \boldsymbol{C} = \begin{bmatrix}\boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{H}\end{bmatrix}, \ \boldsymbol{G} = \begin{bmatrix}\boldsymbol{N} & \boldsymbol{E} \\ -\boldsymbol{E}^T & \boldsymbol{0}\end{bmatrix} \tag{61}$$

Further, we will assume that $\boldsymbol{d} = \boldsymbol{b}$. The matrices $\boldsymbol{N}$, $\boldsymbol{Q}$ and $\boldsymbol{H}$ contain the contribution from resistors, capacitors and inductors respectively. The matrices $\boldsymbol{N}$, $\boldsymbol{Q}$ and $\boldsymbol{H}$ are symmetric positive definite. The matrix $\boldsymbol{E}$ is the incidence matrix of the inductor branches. The vectors $\boldsymbol{v}$ and $\boldsymbol{i}$ correspond to the nodal voltages and inductor currents respectively.

In [23] and [6] it is shown that the use of congruence transformation to create a reduced-order model preserves passivity of a multiport RLC nework with the $\boldsymbol{G}$ and $\boldsymbol{C}$ matrices composed as (61). The basis is generated by a block Arnoldi method and a block rational Arnoldi method, respectively. The reduced-order model is created through congruence transformation by

$$\hat{\boldsymbol{C}} = \boldsymbol{V}_{\bar{k}}^H \boldsymbol{C} \boldsymbol{V}_{\bar{k}}$$

$$\hat{\boldsymbol{G}} = \boldsymbol{V}_{\bar{k}}^H \boldsymbol{G} \boldsymbol{V}_{\bar{k}}$$

and

$$\hat{\boldsymbol{b}} = \boldsymbol{V}_{\bar{k}}^H \boldsymbol{b}$$

The approximate transfer function can now be written as

$$\hat{H}(s) = \hat{\boldsymbol{b}}^H (\hat{\boldsymbol{G}} + s\hat{\boldsymbol{C}})^{-1} \hat{\boldsymbol{b}}.$$

We will show in the following how to generate such a congruence transformation in a efficient way by using only a few extra matrix vector multiplications involving the matrices $\boldsymbol{G}$ and $\boldsymbol{C}$. If we choose the continuation vector $\boldsymbol{t}_k$ on line 8 of the rational Krylov algorithm be equal to $\boldsymbol{e}_k$, then we obtain the matrix vector product $\boldsymbol{C}\boldsymbol{v}_k$ without any additional work. The matrix $\hat{\boldsymbol{C}}$ can be computed as

$$\hat{\boldsymbol{C}} = \boldsymbol{V}_{\bar{k}}^H (\boldsymbol{C} \boldsymbol{V}_{\bar{k}})$$

where we have obtained $\boldsymbol{C}\boldsymbol{V}_{\bar{k}}$ from the rational Krylov algorithm. Rewrite (15)

$$\boldsymbol{G}\boldsymbol{V}_{\bar{k}} \boldsymbol{F}_{\bar{k},\bar{k}} + \boldsymbol{G}\boldsymbol{v}_{\bar{k}+1} f_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^T = \boldsymbol{C}\boldsymbol{V}_{\bar{k}} \boldsymbol{L}_{\bar{k},\bar{k}} + \boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} l_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^T$$

Multiply by $\boldsymbol{V}_{\bar{k}}^H$ from the left and $\boldsymbol{F}_{\bar{k},\bar{k}}^{-1}$ from the right:

$$\boldsymbol{V}_{\bar{k}}^H \boldsymbol{G}\boldsymbol{V}_{\bar{k}} = \boldsymbol{V}_{\bar{k}}^H \boldsymbol{C}\boldsymbol{V}_{\bar{k}} \boldsymbol{L}_{\bar{k},\bar{k}} \boldsymbol{F}_{\bar{k},\bar{k}}^{-1} - \boldsymbol{V}_{\bar{k}}^H \boldsymbol{G}\boldsymbol{v}_{\bar{k}+1} f_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^H \boldsymbol{F}_{\bar{k},\bar{k}}^{-1} + \boldsymbol{V}_{kk}^H \boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} l_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^T \boldsymbol{F}_{\bar{k},\bar{k}}^{-1}$$

which can be written as

$$\hat{\boldsymbol{G}} = \hat{\boldsymbol{C}} \boldsymbol{L}_{\bar{k},\bar{k}} \boldsymbol{F}_{\bar{k},\bar{k}}^{-1} - \boldsymbol{V}_{\bar{k}}^H \boldsymbol{G}\boldsymbol{v}_{\bar{k}+1} f_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^H \boldsymbol{F}_{\bar{k},\bar{k}}^{-1} + \boldsymbol{V}_{kk}^H \boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} l_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^T \boldsymbol{F}_{\bar{k},\bar{k}}^{-1}$$

In order to compute $\hat{\boldsymbol{G}}$, the only quantities of the original model we need to compute are $\boldsymbol{G}\boldsymbol{v}_{\bar{k}+1}$ and $\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1}$.

Another possible approach to create a passive reduced-order model is to rewrite the rational Krylov factorisation (15) as an Arnoldi factorisation [30] [31]. Further work needs to be done in this area before it can be used for creating passive reduced-order models.

9 Implementation Aspects

In this section we will discuss a variant of rational Krylov where the iterations are started directly on $\boldsymbol{b}$, instead of a linear combination of the basis vectors like the algorithm discussed in section 2 when a new interpolation point s_i is used. We will call this version the restarting rational Krylov algorithm.

Restarting Rational Krylov Algorithm

```
1   k = 0
2   for i = 1 : ī
3       for j = 1 : j̄_i
4           if j = 1 then
5               r = (G + s_i C)^{-1} b
6           else
7               r = V_k t_k, choose continuation combination
8               r = (G + s_i C)^{-1} C r
9           endif
10          [v_{k+1}, f_k] = GramSchmidt (V_k, r)
11          k = k + 1
12      end
13  end
14  r = V_k t_k
15  r = (G + s_ī C)^{-1} C r
16  [v_{k+1}, f_k] = GramSchmidt (V_k, r)
```

The case when $(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}\boldsymbol{C}$ operates on $\boldsymbol{V}_k\boldsymbol{t}_k$ in lines 8 and 9 of the algorithm above the rational Krylov relation (15) for column k is identical with equation (14) derived from the rational Krylov algorithm discussed in section 2.

We will now derive the rational Krylov relation (15) for column k for the case when $(\boldsymbol{G} + s_i\boldsymbol{C})^{-1}$ operates on $\boldsymbol{b}$ in line 6 of the restarting rational Krylov algorithm.

Let $\alpha = \| (\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{b} \|$, From lines 6 and 11 we have

$$\begin{aligned} \boldsymbol{V}_{k+1}\boldsymbol{f}_k &= (\boldsymbol{G} + s_{i_k}\boldsymbol{C})^{-1}\boldsymbol{b} \\ &= \alpha(\boldsymbol{G} + s_{i_k}\boldsymbol{C})^{-1}(\boldsymbol{G} + s_1\boldsymbol{C})\boldsymbol{V}_{\bar{k}}\boldsymbol{e}_1. \end{aligned} \tag{62}$$

The last equality comes from $\alpha\boldsymbol{v}_1 = (\boldsymbol{G} + s_1\boldsymbol{C})^{-1}\boldsymbol{b}$. Multiply equation (62) from the left by $(\boldsymbol{G} + s_{i_k}\boldsymbol{C})$, and we get

$$(\boldsymbol{G} + s_{i_k}\boldsymbol{C})\boldsymbol{V}_{k+1}\boldsymbol{f}_k = \alpha(\boldsymbol{G} + s_1\boldsymbol{C})\boldsymbol{V}_k\boldsymbol{e}_1. \tag{63}$$

Put a zero at the bottom of $\boldsymbol{e}_1$ and rearrange equation (63) to get

$$\boldsymbol{G}\boldsymbol{V}_{k+1}(\boldsymbol{f}_k - \alpha\boldsymbol{e}_1) = \boldsymbol{C}\boldsymbol{V}_{k+1}(\alpha s_1\boldsymbol{e}_1 - s_{i_k}\boldsymbol{f}_k). \tag{64}$$

Use (14) and (64), join the column and we finally get (15).

The block rational Arnoldi algorithm discussed by Elfadel and Ling [6] is a block version of the restarting rational Krylov algorithm. The rational Lanczos algorithm discussed by Gallivan, Grimme and Van Dooren [18] is similar to a two-sided version of the restarting rational Krylov algorithm.

Given the same conditions, the restarting rational Krylov algorithm and the rational Krylov algorithm discussed in section 2 generate basis vectors that span the same subspace. However, they are not equivalent with regard to floating-point arithmetic. In the rational Krylov algorithm of section 2 it is always possible to operate with the matrix $(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{C}$ on the last basis vector; this is a stable process. Whereas in the algorithm below, when the interpolation point s_i is changed, the matrix $(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}$ operates on $\boldsymbol{b}$, and then the result is orthogonalised against the basis vectors. If $(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$ is nearly linearly independent of the basis vectors, then this leads to numerical problems similar to those in the parallel rational Krylov algorithm [34].

The more stable algorithm discussed in section 2 comes at an extra cost. the calculation of $\hat{\boldsymbol{b}}_i = \boldsymbol{V}_k^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$ needs to be done explicitly, whereas it is included in the process of building a basis in the restarting rational Krylov algorithm.

10 Error Estimates

10.1 General Comments

In this section we will derive error estimates for the approximate solution $\tilde{\boldsymbol{x}}_i(s)$ and for the approximate transfer function $\hat{H}_i(s)$, both for the main method discussed and for the alternative method discussed in section 7. In all cases the quantity we need to estimate in the original model is $\| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|$; the other quantities of the error estimates can be computed with quantities of the reduced-order model.

For the first reduced-order model computed from the square $\boldsymbol{F}_{\bar{k},\bar{k}}$ and $\boldsymbol{L}_{\bar{k},\bar{k}}$ matrices (theorem 1 and theorem 2), its corresponding approximate solution $\tilde{\boldsymbol{x}}_i(s)$ (49) has an error bounded by

$$\| \Delta\tilde{\boldsymbol{x}}_i(s) \| \leq (1+|s_{\bar{i}}-s| \, \| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|)|\nu_i(s)|,$$

and its transfer function has an error bounded by

$$\begin{aligned} |\epsilon(s)| &\leq (|\boldsymbol{d}^T\boldsymbol{v}_{\bar{k}+1}| + |s_{\bar{i}}-s| \, \| \boldsymbol{d}^T(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|)|\nu_i(s)| \\ &\leq (|\boldsymbol{d}^T\boldsymbol{v}_{\bar{k}+1}| + |s_{\bar{i}}-s| \, \| \boldsymbol{d} \| \| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|)|\nu_i(s)| \end{aligned}$$

where

$$\nu_i(s) = (s_i - s)\rho_i(s).$$

For the second reduced-order model computed from the rectangular $\boldsymbol{F}$ and $\boldsymbol{L}$ matrices (theorem 4 and theorem 5), its corresponding approximate solution $\tilde{\boldsymbol{x}}_i(s)$ (60) has an error bounded by

$$\| \Delta\tilde{\boldsymbol{x}}_i(s) \| \leq (1+|s_i-s| \, \| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|)|\gamma_i(s)| \tag{65}$$

and its transfer function has an error bounded by

$$\begin{aligned} |\epsilon(s)| &\leq (|\boldsymbol{d}^T\boldsymbol{v}_{\bar{k}+1}| + |s_i-s| \, \| \boldsymbol{d}^T(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|)|\gamma_i(s)| \\ &\leq (|\boldsymbol{d}^T\boldsymbol{v}_{\bar{k}+1}| + |s_i-s| \, \| \boldsymbol{d} \| \| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|)|\gamma_i(s)| \end{aligned} \tag{66}$$

where

$$\gamma_i(s) = (s_{\bar{i}} - s)\rho_i(s). \tag{67}$$

The only quantity in these expressions for the errors that involves computation of matrices of the original model is $\| (\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|$. We need to estimate it in a good and efficient way, in order to get a good error estimate. The rest of this section deals with how to estimate $\| (\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|$.

In our approach, we will estimate the norm $\| (\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|$, but not the vector $(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1}$, so we cannot use the quantity $\| \boldsymbol{d}^T(\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \|$ to get a closer error bound of the approximate transfer functions.

10.2 Eigenvalue Distribution

The error estimates are mainly intended for the frequency response function $H(s), s = j\omega$. Their use for more general s needs further investigations. In a typical model-order reduction problem, some eigenvalues are close to the imaginary axis in the desired frequency range, but most eigenvalues are located further down in the left half-plane. In order to get a good approximation, the eigenvalues with corresponding right eigenvectors need to converge in the desired region in the complex plane, as discussed in section 10.3. In order to get a cheaply computable error estimate, we need some convergence of the left eigenvectors in the desired region, as discussed in section 10.4.

10.3 Estimating the Error by Calculating the Left Eigenvectors

Consider the eigenvalue problem

$$\boldsymbol{G}\boldsymbol{u}_j = -\lambda_j\boldsymbol{C}\boldsymbol{u}_j, \tag{68}$$

and the shifted and inverted eigenproblem

$$(\boldsymbol{G} + s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{u}_j = \frac{1}{s - \lambda_j}\boldsymbol{u}_j, \tag{69}$$

where λ_j is an eigenvalue of (68). We assume that $\boldsymbol{v}_{\bar{k}+1}$ can be expressed as a linear combination of the eigenvectors $\boldsymbol{u}_j$ normalised to unit length,

$$\begin{aligned} \boldsymbol{v}_{\bar{k}+1} &= \sum_{j=1}^{n} \alpha_j \boldsymbol{u}_j \\ &= \boldsymbol{U}\boldsymbol{\alpha}, \end{aligned} \tag{70}$$

and $\boldsymbol{\alpha}$ can be computed by

$$\boldsymbol{\alpha} = \boldsymbol{U}^{-1}\boldsymbol{v}_{\bar{k}+1},$$

but it would be far too expensive to actually compute it.

Assume for the moment that the matrix $\boldsymbol{C}$ is nonsingular, which will not be necessary in the general context. Multiply (68) from the left by $\boldsymbol{C}^{-1}$:

$$\boldsymbol{C}^{-1}\boldsymbol{G}\boldsymbol{u}_j = -\lambda_j\boldsymbol{u}_j.$$

Each column of $\boldsymbol{U}^{-H}$ is a left eigenvector of the shifted and inverted eigenproblem (69), and an eigenvector of the eigenproblem

$$\boldsymbol{G}^H\boldsymbol{C}^{-H}\boldsymbol{w}_j = -\bar{\lambda}_j\boldsymbol{w}_j \tag{71}$$

and also to

$$\boldsymbol{C}^{-H}\boldsymbol{w}_j = -\bar{\lambda}_j\boldsymbol{G}^{-H}\boldsymbol{w}_j, \tag{72}$$

but it is not a left eigenvector of (68) in general. If we compute the left eigenvectors $\boldsymbol{w}_j$ and normalise $\boldsymbol{w}_j$ such that

$$\boldsymbol{w}_j^H\boldsymbol{u}_j = 1$$

then we will get

$$\alpha_j = \boldsymbol{w}_j^H\boldsymbol{v}_{\bar{k}+1}. \tag{73}$$

Operate with the matrix $(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}$ on the vector $\boldsymbol{v}_{\bar{k}+1}$:

$$(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\sum_{j=1}^{n}\alpha_j\boldsymbol{u}_j = \sum_{j=1}^{n}\frac{\alpha_j}{s-\lambda_j}\boldsymbol{u}_j \tag{74}$$

and thus

$$\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \parallel \leq \sum_{j=1}^{n}|\frac{\alpha_j}{s-\lambda_j}|. \tag{75}$$

If we assume that none of the α_j are too big, then it is only the terms $|\frac{\alpha_j}{s-\lambda_j}|$ with eigenvalues λ_j close to s that contribute significantly to the error.

Let us now look at how the approximate solution $\tilde{\boldsymbol{x}}_i(s)$ depends on the convergence of eigenvalues with corresponding right eigenvectors around s. For the exact solution $\boldsymbol{x}(s)$ we have

$$\begin{aligned}\boldsymbol{x}(s) &= (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{b}\\ &= (\boldsymbol{G}+s\boldsymbol{C})^{-1}(\boldsymbol{G}+s_i\boldsymbol{C})(\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}\end{aligned}$$

Set

$$\boldsymbol{v} = (\boldsymbol{G}+s_i\boldsymbol{C})^{-1}\boldsymbol{b}$$

and we get

$$\boldsymbol{x}(s) = (\boldsymbol{I}+(s_i-s)(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C})\boldsymbol{v}.$$

If not $\parallel (s_i-s)(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v} \parallel \ll \parallel \boldsymbol{v} \parallel$ then it is necessary that $(\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}$ can be approximated in the basis in a good way. Assume that $\boldsymbol{v}$ can be expressed as a linear combination of the right eigenvectors similar to (70). From (74) we find that, if none of the α_j are too big, then it is only the terms $|\frac{\alpha_j}{s-\lambda_j}|$ with eigenvalues λ_j close to s that contribute significantly to the solution. So a necessary condition seems to be that the eigenvalues close to s with corresponding right eigenvectors need to converge in order to get a good approximation of the exact solution $\boldsymbol{x}(s)$.

One possibility would be first to run a rational Krylov method with shifts $s_1, \ldots, s_{\bar{i}}$ until all eigenvalues with corresponding right eigenvectors have converged around the region in the complex plane where we want a good approximation of $\boldsymbol{x}(s)$; and then to calculate the corresponding left eigenvectors of $\boldsymbol{C}^{-1}\boldsymbol{G}$ and thus the α_j by (73). To see how this can be done efficiently, we may run a rational Krylov algorithm on the eigenproblem (72) with the matrix operators $\boldsymbol{C}^H(\boldsymbol{G}+s_i\boldsymbol{C})^{-H}$, $i=1,\ldots,\bar{i}$ with the same shifts s_i, $i=1,\ldots,\bar{i}$ as the original rational Krylov run, and calculate the left eigenvectors $\boldsymbol{w}_j$ for the region where we have convergent right eigenvectors, and thus calculate the corresponding α_j by (73). We could reduce the computation time by using the same LU factorisations of the matrices $(\boldsymbol{G}+s_i\boldsymbol{C})$, $i=1,\ldots,\bar{i}$ in the two rational Krylov runs, but this would still be costly.

10.4 Estimating the Error Without Calculating the Left Eigenvectors

Replace s_i by s in (36), multiply from the left by $(\boldsymbol{G}+s\boldsymbol{C})^{-1}$ and we get

$$\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F} = (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s\boldsymbol{F}).$$

Multiply from the right by the pseudo inverse of $(\boldsymbol{L}+s\boldsymbol{F})$:

$$\boldsymbol{V}_{\bar{k}+1}\boldsymbol{F}(\boldsymbol{L}+s\boldsymbol{F})^{+} = (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}(\boldsymbol{L}+s\boldsymbol{F})(\boldsymbol{L}+s\boldsymbol{F})^{+}. \qquad (76)$$

Set

$$\boldsymbol{P}(s) = (\boldsymbol{L}+s\boldsymbol{F})(\boldsymbol{L}+s\boldsymbol{F})^{+}$$

where $\boldsymbol{P}(s)$ is an orthogonal projector from $\mathcal{C}^{\bar{k}+1}$ onto the subspace $\mathrm{span}(\boldsymbol{L}+s\boldsymbol{F})$. Multiply (76) by a random vector $\boldsymbol{y}_r$ and normalise:

$$\frac{\parallel \boldsymbol{F}(\boldsymbol{L}+s\boldsymbol{F})^{+}\boldsymbol{y}_r \parallel}{\parallel \boldsymbol{P}(s)\boldsymbol{y}_r \parallel} = \frac{\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{P}(s)\boldsymbol{y}_r \parallel}{\parallel \boldsymbol{P}(s)\boldsymbol{y}_r \parallel}$$

Set $\boldsymbol{q}(s)$ to be the vector

$$\boldsymbol{q}(s) = \boldsymbol{P}(s)\boldsymbol{y}_r / \parallel \boldsymbol{P}(s)\boldsymbol{y}_r \parallel . \qquad (77)$$

Under what conditions is $\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{q}(s) \parallel$ a good estimate of $\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \parallel$?

If some right eigenvectors have converged in the basis $\boldsymbol{V}_{\bar{k}}$, then those eigenvectors are orthogonal to $\boldsymbol{v}_{\bar{k}+1}$. This will cause the corresponding α_j (70) to be equal to zero. From (75) we see that $\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \parallel$ tends to be smaller than $\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{q}(s) \parallel$.

We have observed that in some cases the left eigenvectors converge to some extent in the basis $\boldsymbol{V}_{\bar{k}}$. In this lies another explanation of why the error estimate should work in those cases. If the left eigenvectors corresponding to converged right eigenvectors have converged to some extent in the original rational Krylov basis $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}$, then $\boldsymbol{v}_{\bar{k}+1}$ is close to being orthogonal to the partly converged left eigenvectors $\boldsymbol{w}_j$. Set $\alpha_j^l = \boldsymbol{w}_j^H\boldsymbol{v}_l$ and its absolute value will decrease for

increasing l for those j that correspond to partly converged left eigenvectors $\boldsymbol{w}_j$. Thus for any vector $\boldsymbol{z} \in \text{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}\}$ we have

$$|\boldsymbol{w}_j^H \boldsymbol{z}| \gtrsim |\boldsymbol{w}_j^H \boldsymbol{v}_{\bar{k}+1}|.$$

Finally we can conclude that

$$\begin{aligned} \frac{\| \boldsymbol{F}(\boldsymbol{L}+s\boldsymbol{F})^+ \boldsymbol{y}_r \|}{\| \boldsymbol{P}(s)\boldsymbol{y}_r \|} &= \| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{V}_{\bar{k}+1}\boldsymbol{q}(s) \| \\ &\gtrsim \| (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \| \end{aligned} \tag{78}$$

for the s that belongs to the region of converged right eigenvectors and partly converged left eigenvectors.

10.5 Estimating the Convergence of Right and Left Eigenvectors

In order for the error estimate (78) to be valid we need convergence of the right eigenvectors, and in some cases some convergence of the left eigenvectors corresponding to eigenvalues in the approximation region.

We will derive an alternative way of calculating the eigenvalues and corresponding right eigenvectors, rather than as discussed in [28, 29, 30]. In the process we get an efficient way of calculating the residuals of the eigenproblem. Then we will describe how to calculate the left eigenvectors and estimate the convergence. The subspace that the rational Krylov method is building up for the solution of the eigenproblem (68) gives generally much better estimates of the right eigenvectors than the left. Consider the eigenproblem (72) We have assumed that the pencil is regular, which means that $(\boldsymbol{G}+s\boldsymbol{C})$ is invertible for most s. It is then natural to calculate the spectrally transformed left eigenproblem.

$$\frac{1}{\bar{s}-\bar{\lambda}_j}\boldsymbol{w}_j = \boldsymbol{C}^H(\boldsymbol{G}+s\boldsymbol{C})^{-H}\boldsymbol{w}_j,$$

where $\bar{\lambda}_j$ is an eigenvalue to (71). From the above, and from the relation between the eigenproblems (68) and (69) for the right eigenvector, we get the following approach.

Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}, \boldsymbol{v}_{\bar{k}+1}, \ldots, \boldsymbol{v}_n$ be an orthonormal basis that spans $\mathcal{C}^n$ in such a way that the first $\bar{k}+1$ basis vectors are generated by the rational Krylov algorithm. Set

$$\boldsymbol{A} = (\boldsymbol{G}+s_{\bar{i}}\boldsymbol{C})^{-1}\boldsymbol{C}, \tag{79}$$

$$\boldsymbol{V}_a = \boldsymbol{V}_{\bar{k}},$$

$$\boldsymbol{V}_b = \boldsymbol{V}_{\bar{k}+1:n},$$

and we get

$$\boldsymbol{A}[\boldsymbol{V}_a \boldsymbol{V}_b] = [\boldsymbol{V}_a \boldsymbol{V}_b] \begin{bmatrix} \boldsymbol{H}_{a,a} & \boldsymbol{H}_{a,b} \\ \boldsymbol{H}_{b,a} & \boldsymbol{H}_{b,b}. \end{bmatrix} \tag{80}$$

The matrices $\boldsymbol{H}_{a,a}$ and $\boldsymbol{H}_{b,a}$ are known, and we get them from (79) through (80), and by multiplying (36) from the right by $(\boldsymbol{L}_{\bar{k},\bar{k}} + s_{\bar{\imath}}\, \boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$ and from the left by $(\boldsymbol{G} + s_i \boldsymbol{C})^{-1}$. Note that $(l_{\bar{k}+1,\bar{k}} + s_{\bar{\imath}}\, f_{\bar{k}+1,\bar{k}}) = 0$. Thus

$$\boldsymbol{H}_{a,a} = \boldsymbol{F}_{\bar{k},\bar{k}} (\boldsymbol{L}_{\bar{k},\bar{k}} + s_{\bar{\imath}}\, \boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$$

and

$$\boldsymbol{H}_{b,a} = \boldsymbol{e}_1 f_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^T (\boldsymbol{L}_{\bar{k},\bar{k}} + s_{\bar{\imath}}\, \boldsymbol{F}_{\bar{k},\bar{k}})^{-1}. \tag{81}$$

Consider the eigenproblem

$$\boldsymbol{A}\boldsymbol{u}_j = \eta_j \boldsymbol{u}_j \tag{82}$$

Let $(\tilde{\eta}_j, \boldsymbol{y}_j)$ be an eigenpair of

$$\boldsymbol{H}_{a,a}\boldsymbol{y}_j = \tilde{\eta}_j \boldsymbol{y}_j \tag{83}$$

and take

$$\tilde{\boldsymbol{u}}_j = \boldsymbol{V}_a \boldsymbol{y}_j \tag{84}$$

as the approximate eigenvector and $\tilde{\eta}_j$ as the approximate eigenvalue. The relation between the eigenvalue η_j of (82) and the eigenvalue λ_j of (68) is

$$\frac{1}{s_{\bar{\imath}} - \lambda_j} = \eta_j$$

From (80) we get the relation

$$\boldsymbol{A}\boldsymbol{V}_a = \boldsymbol{V}_a \boldsymbol{H}_{a,a} + \boldsymbol{V}_b \boldsymbol{H}_{b,a} \tag{85}$$

and the residual will be

$$\begin{aligned}
\boldsymbol{A}\tilde{\boldsymbol{u}}_j - \tilde{\eta}_j \tilde{\boldsymbol{u}}_j &= \boldsymbol{A}\boldsymbol{V}_a \boldsymbol{y}_j - \tilde{\eta}_j \boldsymbol{V}_a \boldsymbol{y}_j \\
&= \boldsymbol{A}\boldsymbol{V}_a \boldsymbol{y}_j - \boldsymbol{V}_a \boldsymbol{H}_{a,a} \boldsymbol{y}_j \\
&= \boldsymbol{V}_b \boldsymbol{H}_{b,a} \boldsymbol{y}_j \\
&= \boldsymbol{v}_{\bar{k}+1} f_{\bar{k}+1,\bar{k}} \boldsymbol{e}_{\bar{k}}^T (\boldsymbol{L}_{\bar{k},\bar{k}} + s_{\bar{\imath}}\, \boldsymbol{F}_{\bar{k},\bar{k}})^{-1} \boldsymbol{y}_j.
\end{aligned}$$

The first equality follows from (84), the second from (83), the third from (85), and the fourth from (81).

Multiply (80) from the left by $\boldsymbol{V}^H = [\boldsymbol{V}_a \boldsymbol{V}_b]^H$:

$$\boldsymbol{V}^H \boldsymbol{A} \boldsymbol{V} = \boldsymbol{H} \tag{86}$$

Complex transpose (86) and multiply by $\boldsymbol{V}$ to get

$$\boldsymbol{A}^H [\boldsymbol{V}_a \boldsymbol{V}_b] = [\boldsymbol{V}_a \boldsymbol{V}_b] \begin{bmatrix} \boldsymbol{H}_{a,a}^H & \boldsymbol{H}_{b,a}^H \\ \boldsymbol{H}_{a,b}^H & \boldsymbol{H}_{b,b}^H \end{bmatrix} \tag{87}$$

Consider the eigenproblem

$$\boldsymbol{A}^H \boldsymbol{w}_j = \bar{\eta}_j \boldsymbol{w}_j$$

Let $(\bar{\tilde{\eta}}_j, \boldsymbol{y}_j)$ be an eigenpair of

$$\boldsymbol{H}_{a,a}^H \boldsymbol{y}_j = \bar{\tilde{\eta}}_j \boldsymbol{y}_j \tag{88}$$

and take

$$\tilde{\boldsymbol{w}}_j = \boldsymbol{V}_a \boldsymbol{y}_j \tag{89}$$

as the approximate eigenvector and $\bar{\tilde{\eta}}_j$ as the approximate eigenvalue. From (87) we get the relation

$$\boldsymbol{A}^H \boldsymbol{V}_a = \boldsymbol{V}_a \boldsymbol{H}_{a,a}^H + \boldsymbol{V}_b \boldsymbol{H}_{a,b}^H \tag{90}$$

and the residual will be

$$\begin{aligned} \boldsymbol{A}^H \tilde{\boldsymbol{w}}_j - \bar{\tilde{\eta}}_j \tilde{\boldsymbol{w}}_j &= \boldsymbol{A}^H \boldsymbol{V}_a \boldsymbol{y}_j - \bar{\tilde{\eta}}_j \boldsymbol{V}_a \boldsymbol{y}_j \\ &= \boldsymbol{A}^H \boldsymbol{V}_a \boldsymbol{y}_j - \boldsymbol{V}_a \boldsymbol{H}_{a,a}^H \boldsymbol{y}_j \\ &= \boldsymbol{V}_b \boldsymbol{H}_{a,b}^H \boldsymbol{y}_j. \end{aligned} \tag{91}$$

The first equality follows from (89), the second from (88), and the third from (90).

The matrix $\boldsymbol{H}_{a,b}^H$ is unknown, but we may get a crude estimate of its norm by multiplying (90) by a normalised random vector $\boldsymbol{y}_r$:

$$\begin{aligned} \| \boldsymbol{H}_{a,b}^H \boldsymbol{y}_r \| &= \| \boldsymbol{A}^H \boldsymbol{V}_a \boldsymbol{y}_r - \boldsymbol{V}_a \boldsymbol{H}_{a,a}^H \boldsymbol{y}_r \| \\ &\leq \| \boldsymbol{H}_{a,b}^H \| . \end{aligned} \tag{92}$$

A crude estimate of the convergence of the left eigenvectors would be to calculate the norm of the normalised residual:

$$\frac{1}{\| \boldsymbol{H}_{a,b}^H \boldsymbol{y}_r \|} \| \boldsymbol{A}^H \boldsymbol{V}_a \boldsymbol{y}_j - \boldsymbol{V}_a \boldsymbol{H}_{a,a}^H \boldsymbol{y}_j \| = \frac{\| \boldsymbol{H}_{a,b}^H \boldsymbol{y}_j \|}{\| \boldsymbol{H}_{a,b}^H \boldsymbol{y}_r \|}$$

10.6 Invariant Subspaces

In the following we will describe a connection between a right invariant subspace and the corresponding left subspace, which is not invariant in general. It will give insight into why the left eigenvectors sometimes converge to some extent in the basis $\boldsymbol{V}_{\bar{k}}$.

We will assume that the subspace span$\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}\}$ builds up an invariant subspace under $\boldsymbol{A}$, and thus $\boldsymbol{H}_{b,a} = 0$. From (80) we get

$$\boldsymbol{A}[\boldsymbol{V}_a \boldsymbol{V}_b] = [\boldsymbol{V}_a \boldsymbol{V}_b] \begin{bmatrix} \boldsymbol{H}_{a,a} & \boldsymbol{H}_{a,b} \\ 0 & \boldsymbol{H}_{b,b} \end{bmatrix}$$

If we solve the eigenproblem

$$\boldsymbol{H}_{a,a} \boldsymbol{y}_j = \eta_j \boldsymbol{y}_j$$

then η_j will be an eigenvalue of $\boldsymbol{A}$ and

$$\boldsymbol{u}_j = \boldsymbol{V}_a \boldsymbol{y}_j$$

will be the corresponding eigenvector of $\boldsymbol{A}$.

The following theorem gives a connection between a right invariant subspace and a left subspace.

Theorem 6. *Let $\boldsymbol{A} \in \mathcal{C}^{n\times n}$ be a matrix with only simple eigenvalues. Furthermore let, $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}, \boldsymbol{v}_{\bar{k}+1}, \ldots, \boldsymbol{v}_n$ be an orthonormal basis of $\mathcal{C}^n$ in such a way that $\boldsymbol{A}$ is invariant under the subspace span$\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}\}$. If $\boldsymbol{V}_a = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\bar{k}}]$, $\boldsymbol{V}_b = [\boldsymbol{v}_{\bar{k}+1}, \ldots, \boldsymbol{v}_n]$ and $\boldsymbol{H}$ is defined by the relation*

$$\boldsymbol{A}[\boldsymbol{V}_a \boldsymbol{V}_b] = [\boldsymbol{V}_a \boldsymbol{V}_b] \begin{bmatrix} \boldsymbol{H}_{a,a} & \boldsymbol{H}_{a,b} \\ 0 & \boldsymbol{H}_{b,b} \end{bmatrix},$$

then there exists a similarity transformation such that

$$\begin{bmatrix} \boldsymbol{H}_{a,a} & 0 \\ 0 & \boldsymbol{H}_{b,b} \end{bmatrix} = \boldsymbol{X}^{-1} \boldsymbol{A} \boldsymbol{X},$$

$\boldsymbol{X}$ and $\boldsymbol{X}^{-1}$ being given by

$$\boldsymbol{X} = [\boldsymbol{V}_a, \boldsymbol{V}_a \boldsymbol{Q} + \boldsymbol{V}_b],$$

$$\boldsymbol{X}^{-1} = [\boldsymbol{V}_a - \boldsymbol{V}_b \boldsymbol{Q}^H, \boldsymbol{V}_b]^H$$

where $\boldsymbol{Q}$ is the unique solution to the equation

$$\boldsymbol{H}_{a,a} \boldsymbol{Q} - \boldsymbol{Q} \boldsymbol{H}_{b,b} = -\boldsymbol{H}_{a,b}. \tag{93}$$

Proof: For proof of Theorem 6 see the proof of theorem 1.5 in Chapter V, on invariant subspaces, in the book by Stewart and Sun [36].

From theorem 6 we get

$$\begin{bmatrix} \boldsymbol{H}_{a,a}^H & 0 \\ 0 & \boldsymbol{H}_{b,b}^H \end{bmatrix} = \boldsymbol{X}^H \boldsymbol{A}^H \boldsymbol{X}^{-H}$$

If we solve the eigenproblem

$$\boldsymbol{H}_{a,a}^H \boldsymbol{y}_j = \bar{\eta}_j \boldsymbol{y}_j$$

then $\bar{\eta}_j$ will be an eigenvalue of $\boldsymbol{A}^H$ and

$$\boldsymbol{w}_j = \boldsymbol{V}_a \boldsymbol{y}_j - \boldsymbol{V}_b \boldsymbol{Q}^H \boldsymbol{y}_j$$

will be the corresponding eigenvector; the matrix $\boldsymbol{Q}$ is the solution to the equation (93).

Let α be a scalar such that

$$\alpha = \max_{\bar{\eta}_j \in \lambda(\boldsymbol{H}_{a,a}^H)} \frac{\| \boldsymbol{Q}^H \boldsymbol{y}_j \|}{\| \boldsymbol{y}_j \|}, \tag{94}$$

where $\boldsymbol{Q}$ is the solution of (93). The scalar α gives a measure of the convergence of the left eigenvectors corresponding to converged right eigenvectors in the basis generated by the rational Krylov algorithm. If α is large, then some left eigenvalues have not yet converged.

The scalar α (94) is not computable with known quantities, but the magnitude of α can be estimated by residual calculations. If we assume that $\frac{\|\boldsymbol{Q}^H \boldsymbol{y}_j\|}{\|\boldsymbol{y}_j\|}$ is of the same order of magnitude corresponding to converged eigenvalues and right eigenvectors, then it is sufficient to test the convergence of a few left eigenvectors by calculating the residual (91). The matrix $\boldsymbol{H}_{a,b}$ is not known, but its norm can be estimated by (92).

11 Testing

11.1 Introduction

If not otherwise stated, the tests are with regard to the reduced-order model presented in theorem 1. The version of the rational Krylov algorithm presented in section 2 is used in all tests.

The objective of these tests is to test the rational Krylov algorithm, the error estimate, and the quantities that are needed for the error estimate to work.

We will test several different strategies to choose the vector $\boldsymbol{y}_r$ of the error estimate (78). One strategy is to choose the vector $\boldsymbol{y}_r$ as a normalised random vector; this will in many cases be an overestimate of the error as discussed in section 10.4. We will refer to this strategy as random-vector error estimation

One way to come a bit closer to the real error is to choose the initial parts of the vector $\boldsymbol{y}_r$ to be zero. The idea behind this strategy is that the vector $\boldsymbol{q}(s)$ (77) becomes closer to the vector $\boldsymbol{e}_{\bar{k}+1}$ in some sense, and thus the estimate becomes closer to $\parallel (\boldsymbol{G}+s\boldsymbol{C})^{-1}\boldsymbol{C}\boldsymbol{v}_{\bar{k}+1} \parallel$. We will refer to this strategy as random-vector error estimation with initial zeros.

Another way to come closer to the real error is to initialise the vector $\boldsymbol{y}_r$ as a random vector, and then discard the eigenvectors of the matrix

$$\boldsymbol{F}_{\bar{k},\bar{k}}(\boldsymbol{L}_{\bar{k},\bar{k}}+s\boldsymbol{F}_{\bar{k},\bar{k}})^{-1}$$

that corresponds to converged eigenvalues of (7). We will refer to this strategy as discarded-eigenvectors error estimation.

By choosing the second smallest right singular vector of the matrix

$$\boldsymbol{F}(\boldsymbol{L}+s\boldsymbol{F})^{+} \tag{95}$$

of the error estimate (78) as the vector $\boldsymbol{y}_r$, we will in many cases get an underestimate of the error. Note that the matrix (95) generally has a null space of dimension one, so the smallest singular value is equal to zero. The norm of the vector $\boldsymbol{P}(s)\boldsymbol{y}_r$ of (78) is usually close to 1, and its upper bound is 1. We will refer to this strategy as small-singular-values error estimation.

We will test the convergence of both right and left eigenvectors of the matrix $\boldsymbol{A}$ (79) in the subspace $\mathcal{S}_{\bar{k}}$. The residuals corresponding to the right eigenvectors are normalised by $\parallel \boldsymbol{H}_{b,a} \parallel$ and the residuals corresponding to the left eigenvectors are normalised by an estimate of $\parallel \boldsymbol{H}_{a,b} \parallel$ (92).

Another way to test the convergence is to compute the exact right and left eigenvectors of the matrix $\boldsymbol{A}$ (79), and then check how much they have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). One measure is to compute

$$1- \parallel \boldsymbol{V}_{\bar{k}}^{H}\boldsymbol{u}_k \parallel \tag{96}$$

for the normalised right eigenvectors of $\boldsymbol{A}$ (79), and a corresponding measure for the normalised left eigenvectors. Of course this is useful in evaluating methods, but not for practical use in error estimates. If an eigenvector has converged in the subspace span($\boldsymbol{V}_{\bar{k}}$), then (96) is equal to zero. We will refer to the measure (96) as a projected-eigenvector measure.

In several of the plots we plot the amplitude characteristic of the approximate transfer function $|\hat{H}(jf)|$ together with the error $|H(jf)-\hat{H}(jf)|$ and error estimates, where $j=\sqrt{-1}$ and f is the frequency.

11.2 Crosstalk Via PEEC

This test circuit is a full wave analysis of three conducting strips modelling a crosstalk problem via PEEC. It is provided by Zhaojun Bai and is of dimension $n = 256$. The rational Krylov algorithm is applied with the shifts $s_1 = j3 \times 10^8$, $s_2 = j2 \times 10^9$ and $s_3 = j8 \times 10^9$, where $j = \sqrt{-1}$. The shifts are applied 3, 6 and 8 times respectively, so the dimension of the reduced-order model is $\bar{k} = 17$.

In Figure 1, $|\hat{H}_3(j2\pi f)|$ is plotted together with the error $|H(j2\pi f) - \hat{H}_3(j2\pi f)|$ and error estimates, where we have used the interpolation point s_3 in the approximate transfer function $\hat{H}_3(s)$ (34). The error estimates differ only in how the vector $\boldsymbol{y}_r$ of (78) is chosen. The two error estimates based on choosing the vector $\boldsymbol{y}_r$ as a random vector are overestimates almost all the way. The variant where we have chosen the initial elements to be zero (initial 80%) gets closer to the real error. The discarded-eigenvectors error estimate gets closest to the real error of all estimates, in some parts an overestimate, and in other parts an underestimate. The eigenvalues corresponding to discarded eigenvectors are plotted in Figure (3) together with eigenvalues in the approximation region. The small-singular-values error estimate is an underestimate of the error all the way.

In Figure 2, $|\hat{H}_1(j2\pi f)|$ is plotted together with the error $|H(j2\pi f) - \hat{H}_1(j2\pi f)|$ and error estimates. Note that the error estimate based on choosing the vector $\boldsymbol{y}_r$ as a random vector gets closer to the real error, and that the small-singular-value error estimate gets even more of an underestimate than the corresponding estimates to $\hat{H}_3(s)$.

The eigenvalue distribution is given in Figure 3. Compare the eigenvalue distribution with the plots for the transfer function and error estimates, and note that the transfer function, and the error and error estimates, have a top near the imaginary part of each eigenvalue.

In Figure 4, we show the normalised residual corresponding to approximations of the right and left eigenvectors in the subspace span($\boldsymbol{V}_{\bar{k}}$) of the matrix $\boldsymbol{A}$ (79). The norm of the residuals is plotted against the imaginary part of the original eigenvalue (68). To get a measure of how much the left eigenvectors have converged, we have plotted the normalised residual where the left eigenvector is replaced by a normalised random vector. We have also plotted the approximate transfer function. The residual test indicates that the right eigenvectors have converged in the approximation region and some of the left eigenvectors have converged to some extent.

In Figure 5, we show how much the right and left eigenvectors have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). We have plotted the projected-eigenvector measure (96) for the right and left eigenvectors. We have also plotted the corresponding measure for normalised random vectors, to get a measure of how much the left eigenvectors have converged. The projected-eigenvector measure (96) is plotted against the imaginary part of the original eigenvalue (68). We have also plotted the approximate transfer function. Similar to the residual test, this test indicates that the right eigenvectors have converged in the approximation region and some of the left eigenvectors have converged to some extent.

11.3 The RF Circuit

This test circuit models the pin package of an RF circuit; it is provided by Peter Feldmann [14] and is of dimension $n = 1841$. The rational Krylov algorithm is applied with the shifts $s_1 = j2\pi \times 10^8$, $s_2 = j8\pi \times 10^9$ $s_3 = j10\pi \times 10^9$ and $s_3 = j12\pi \times 10^9$, where $j = \sqrt{-1}$. All shifts are applied 7 times, so the dimension of the reduced-order model is $\bar{k} = 28$.

In Figure 6 $|\hat{H}_1(jf)|$ is plotted together with the error $|H(jf) - \hat{H}_1(jf)|$ and error estimates, where we have used the interpolation point s_1 in the approximate transfer function $\hat{H}_1(s)$ (34). The error estimates differ only in how the vector $\boldsymbol{y}_r$ of (78) is chosen. We have chosen the same error estimates as in the previous test problem, except that the initial 95% of the vector $\boldsymbol{y}_r$ is zero in the random-vector error estimate with initial zeros.

The trend for the error estimates is similar to the previous test problem.

The eigenvalue distribution is given in Figure 7.

In Figure 8, we show how much the right and left eigenvectors have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). We have plotted the projected-eigenvector measure (96) for the right and left eigenvectors. The projected-eigenvector measure (96) is plotted against the imaginary part of the original eigenvalue (68). The test indicates that the right eigenvectors have converged in the approximation region; in contrast to the cross talk via PEEC test problem, the left eigenvectors have not converged.

11.4 The PEEC Circuit

This test circuit is generated from the PEEC discretisation [25] of an electromagnetic problem. The dimension of the problem is $n = 306$. The rational Krylov algorithm is applied with the shifts $s_i = j2\pi \times 10^9 \times [1,\ 1.5,\ 2,\ 2.5,\ 3,\ 3.5,\ 4,\ 4.5,\ 5]$; they are applied [2, 5, 5, 5, 5, 5, 5, 5, 2] times respectively, so the dimension of the reduced-order model is $\bar{k} = 39$.

In Figure 9, $|\hat{H}_3(jf)|$ is plotted, and in Figure 10 $|\hat{H}_3(jf)|$ is plotted together with the error $|H(jf) - \hat{H}_3(jf)|$ and error estimates, where we have used the interpolation point s_3 in the approximate transfer function $\hat{H}_3(s)$ (34). The error estimates differ only in how the vector $\boldsymbol{y}_r$ of (78) is chosen. We have chosen the initial 80% of the vector $\boldsymbol{y}_r$ to be zero for the random-vector error estimate with initial zeros. Here it gets more of an overestimate than in previous test problems. We have also chosen the small-singular-values error estimate. This estimate comes rather close to the real error in this case.

In Figure 11, we show how much the right and left eigenvectors have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). For the right and left eigenvector we have plotted the projected-eigenvector measure (96). The projected-eigenvector measure (96) is plotted against the imaginary part of the original eigenvalue (68). The test indicates that the right eigenvectors have converged in the approximation region; similar to the RF circuit test problem and differently from the crosstalk via PEEC test problem the left eigenvectors have not converged.

11.5 RC Circuit

The RC circuit is of dimension $n = 13875$, it models the interconnect on a chip. The test problem is provided by Roland Freund. The matrices $\boldsymbol{G}$ and $\boldsymbol{C}$ are

symmetric and positive definite and thus a stable reduced-order model can be created through congruence transformation. The rational Krylov algorithm is applied with the shifts $s_i = [10^1,\ 10^7,\ 10^8,\ 10^9,\ 10^{10}] * 2 * \pi$ one time each, so the dimension of the reduced order model is $\bar{k} = 5$, further $\boldsymbol{d} = \boldsymbol{b}$. In figure 12 we have plotted $|\hat{H}(jf)|$ of the reduced order model created through congruence transformations together with the error $|H(jf) - \hat{H}(jf)|$. We have also plotted the error $|H(jf) - \hat{H}_3(jf)|$ as a comparison.

By inspection the matrices $\hat{\boldsymbol{G}}$ and $\hat{\boldsymbol{C}}$ of the reduced-order model created through congruence transformation are symmetric and positive definite, which they should be according to theory, and thus we have a passive reduced-order model.

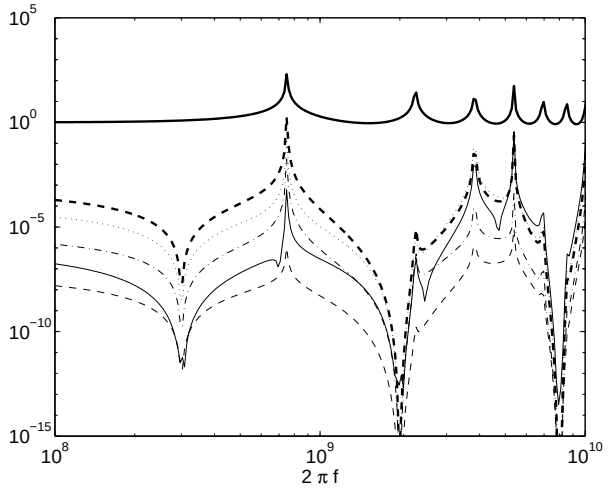

Figure 1: $|\hat{H}_3(j2\pi f)|$ of the crosstalk via PEEC is plotted by a thick solid line $-$, and the error $|H(j2\pi f) - \hat{H}_3(j2\pi f)|$ by a thin solid line $-$. The random-vector error estimate is plotted by a thick dashed line $- -$, the random-vector error estimate with initial zeros by a dotted line $\cdot\,\cdot$, the discarded eigenvectors error estimate by a dash dot line $-\,\cdot$, and the small singular value error estimate by a dashed line $- -$.

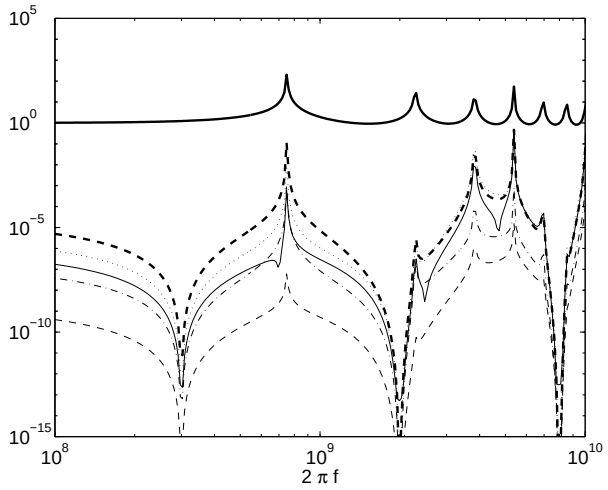

Figure 2: $|\hat{H}_1(j2\pi f)|$ corresponding to $|\hat{H}_3(j2\pi f)|$ of Figure 1 is plotted together with the error $|H(j2\pi f) - \hat{H}_1(j2\pi f)|$ and error estimates.

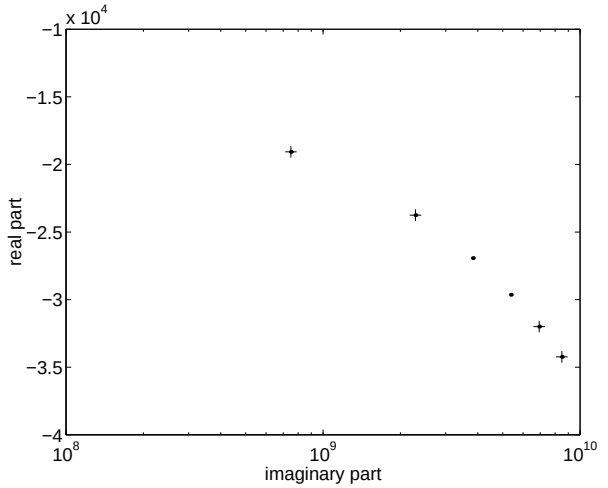

Figure 3: The eigenvalue distribution of the test matrices of the crosstalk via PEEC. The eigenvalues are plotted with a dot $\cdot$. The eigenvalues corresponding to discarded eigenvectors for the discarded-eigenvectors error estimate are plotted with a $+$. Note that the real parts are plotted with a linear scaling on the y-axis, and the imaginary parts with a logarithmic scaling on the x-axis.

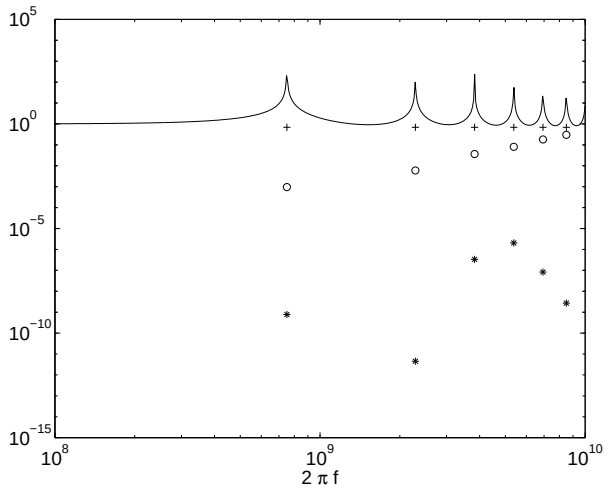

Figure 4: Residual-based test for convergence of left and right eigenvectors of the test matrices of the crosstalk via PEEC. The normalised residuals corresponding to approximations of the right and left eigenvectors in the subspace span($\boldsymbol{V}_{\bar{k}}$) are plotted by a star $*$ and a circle $\circ$ respectively. The normalised residuals for the left eigenvector replaced by a random vector are plotted by a plus $+$. $|H(j2\pi f)$ is plotted by a solid line $-$.

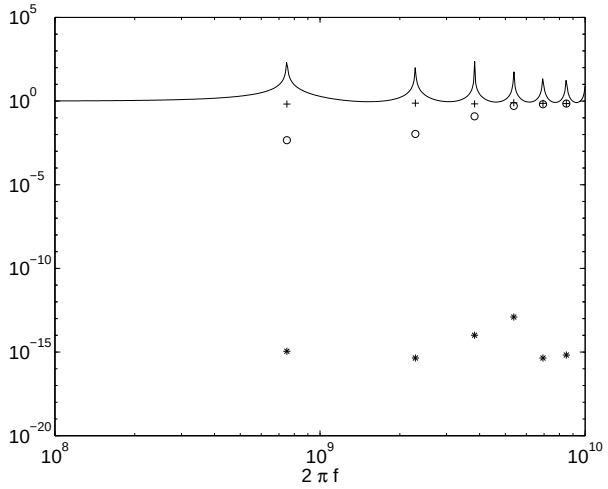

Figure 5: This plot shows how much the right and left eigenvectors of the test matrices of the crosstalk via PEEC have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). The measure $1- \parallel \boldsymbol{V}_{\bar{k}}^{H} \boldsymbol{u} \parallel$ corresponding to the right eigenvector, left eigenvector and a normalised random vector are plotted by a star $*$, a circle $\circ$ and a plus $+$ respectively. $|H(j2\pi f)|$ is plotted by a solid line $-$.

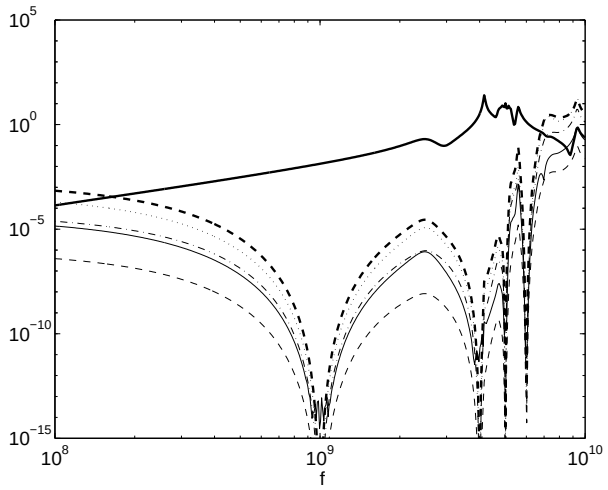

Figure 6: $|\hat{H}_1(jf)|$ of the RF circuit is plotted by a thick solid line $-$, and the error $|H(jf) - \hat{H}_1(jf)|$ is plotted by thin solid line $-$. The random-vector error estimate is plotted by a thick dashed line$-\ -$, the random vector error estimate with initial zeros by a dotted line $\cdot\ \cdot$. the discarded-eigenvectors error estimate by a dash dot line $-\ \cdot$, and the small-singular-value error estimate by a dashed line $-\ -$.

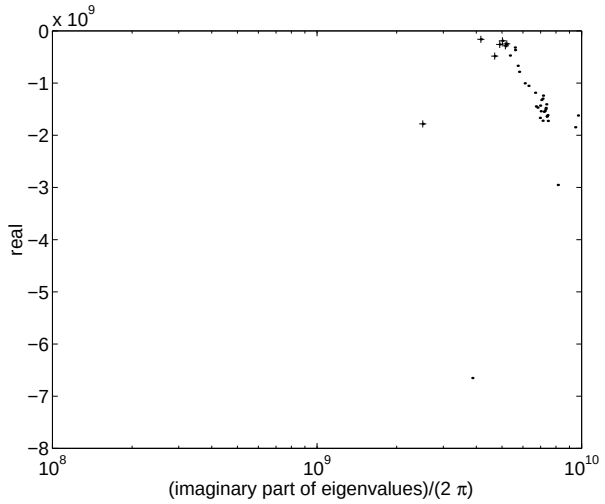

Figure 7: The eigenvalue distribution for the test matrices of the RF circuit. The eigenvalues are plotted with a dot $\cdot$. The eigenvalues corresponding to discarded eigenvectors for the discarded-eigenvectors error estimate are plotted with a $+$. Note that the real parts are plotted with a linear scaling on the y-axis, and the imaginary parts with a logarithmic scaling on the x-axis.

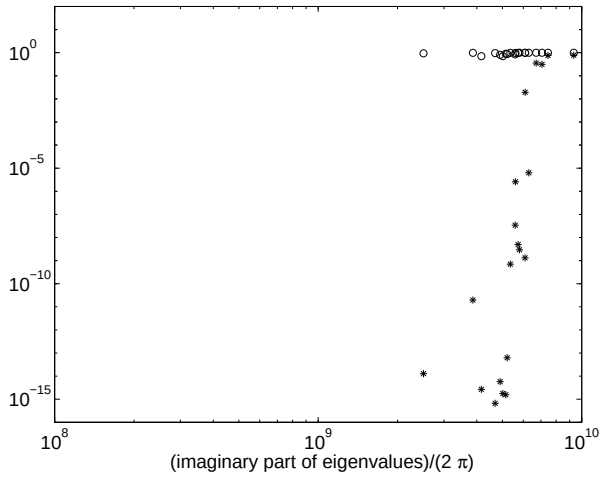

Figure 8: This plot shows how much the right and left eigenvectors for the test matrices of the RF circuit have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). The measure $1- \parallel \boldsymbol{V}_{\bar{k}}^{H} \boldsymbol{u} \parallel$ corresponding to the right eigenvector and left eigenvector and left eigenvector is plotted by a star $*$ and a circle $\circ$ respectively.

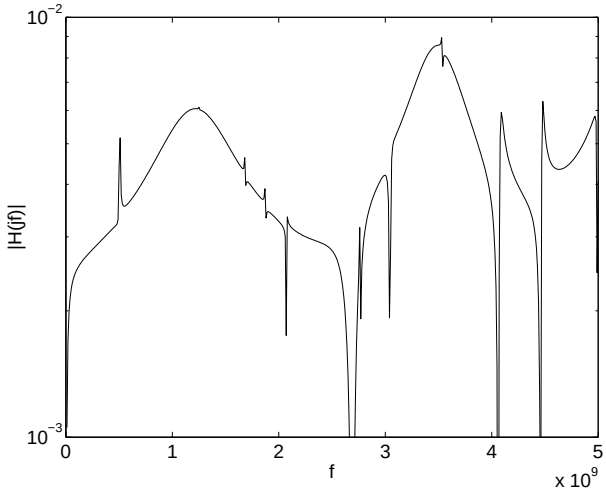

Figure 9: $|\hat{H}_3(jf)|$ of the PEEC circuit is plotted by a solid line $-$.

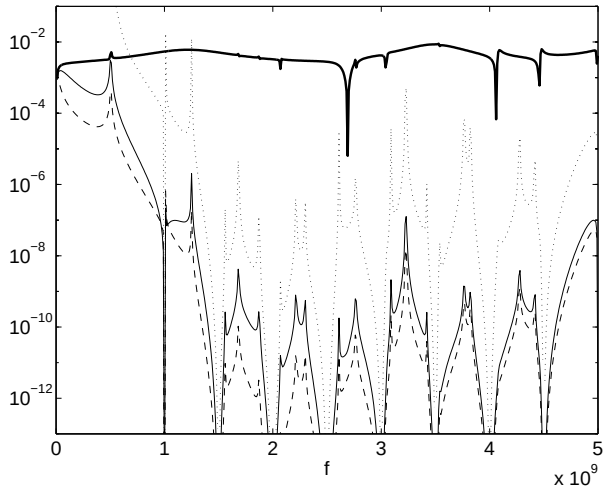

Figure 10: $|\hat{H}_3(jf)|$ of the PEEC circuit is plotted by a thick solid line $-$, the error $|H(jf) - \hat{H}_3(jf)|$ by thin solid line $-$, the random-vector error estimate with initial zeros by a dotted line $\cdot\,\cdot$, and the small singular value error estimate by a dashed line $-\,-$.

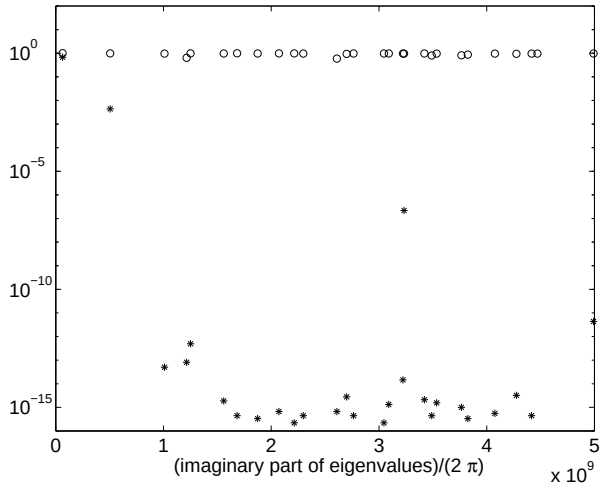

Figure 11: This plot shows how much the right and left eigenvectors of the test matrices of the PEEC circuit have converged in the subspace span($\boldsymbol{V}_{\bar{k}}$). The measure $1 - \| \boldsymbol{V}_{\bar{k}}^H \boldsymbol{u} \|$ corresponding to the right eigenvector and left eigenvector is plotted by a star $*$ and a circle $\circ$ respectively.

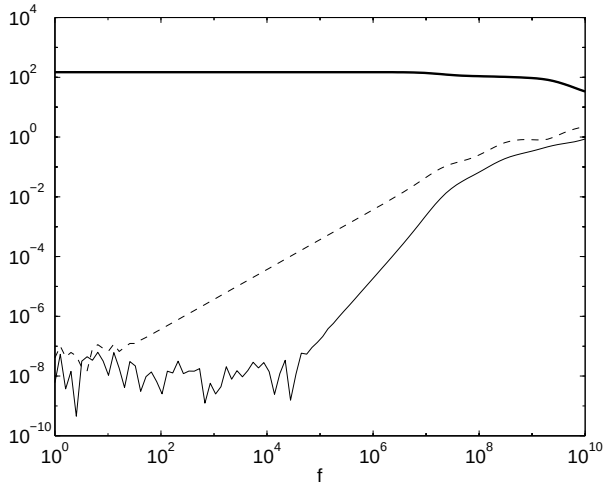

Figure 12: $|\hat{H}(jf)|$ of the reduced-order model created through congruence transformation of the RC circuit is plotted by a thick solid line $-$, the error $|H(jf) - \hat{H}(jf)|$ by thin solid line $-$. The error $|H(jf) - \hat{H}_3(jf)$ is plotted by a dashed line $- -$.

References

[1] Z. Bai, P. Feldmann, and R. W. Freund. Stable and passive reduced-order models based on partial Padé approximation via the Lanczos process. Numerical Analysis Manuscript 97–3–10, Bell Laboratories, 1997.

[2] Zhaojun Bai and Qiang Ye. Error Estimation of the Padé Approximation of Transfer Functions Via the Lanczos Process. Manuscript, July 1998.

[3] G. A. Baker, Jr. and P. Graves-Morris. *Padé Approximants part I and II in Encyclopedia of mathematics and its applications.* Addison-Wesley, 1981.

[4] M. Celik, O. Ocali, M. A. Tan, and A. Atalar. Pole-zero computations in microwave circuits using multipoint Padé approximation. *IEEE Trans. Circuits and System-I: Fundamental Theory and Applications*, 42(1):6–13, Jan. 1995.

[5] G. De Samblanx, K. Meerbergen, and A. Bultheel. The implicit application of a rational filter in the rks method. *BIT*, 37:925–947, 1997.

[6] I. M. Elfadel and D. D. Ling. A block rational Arnoldi algorithm for multipoint passive model-order reduction of multiport RLC networks. In *ICCAD' 97. Proceedings of the 1997 IEEE/ACM International Conference on Computer-Aided Design*, pages 66–71, 1997.

[7] I. M. Elfadel and David D. Ling. Zeros and passivity of Arnoldi-reduced-order models for interconnect networks. In *Proc. 34nd ACM/IEEE Design Automation Conf.*, pages 28–33. Association for Computing Machinery Inc., New York, New York, 1997.

[8] P. Feldmann and R. W. Freund. Efficient linear circuit analysis by Padé approximation via the Lanczos process. In *Proceedings of EURO-DAC '94 with EURO-VHDL '94, Grenoble, France*, pages 170–175. IEEE Computer Society Press, 1994.

[9] 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*, 14:639–649, 1995.

[10] P. Feldmann and R. W. Freund. Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm. In *Proceedings of the 32nd Design Automation Conference, San Francisco, California*, pages 474–479. Association for Computing Machinery, Inc., 1995.

[11] R. W. Freund. Reduced-order modeling techniques based on Krylov subspaces and their use in circuit simulation. *Applied and Computational Control, Signals, and Circuits*, 1998, to appear.

[12] R. W. Freund and P. Feldmann. Reduced-order modeling of large passive linear circuits by means of the SyPVL algorithm. In *Technical Digest of the 1996 IEEE/ACM International Conference on Computer-Aided Design*, pages 280–287. IEEE Computer Society Press, 1996.

[13] R. W. Freund and P. Feldmann. Small-signal circuit analysis and sensitivity computations with the PVL algorithm. *IEEE Transactions on Circuits and and Systems—II: Analog and Digital Signal Processing*, 43:577–585, 1996.

[14] R. W. Freund and P. Feldmann. The SyMPVL algorithm and its applications to interconnect simulation. In *Proceedings of the 1997 International Conference on Simulation of Semiconductor Processes and Devices*, pages 113–116. IEEE, 1997.

[15] R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal. An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices. *SIAM J. Sci. Comp.*, 14:137–158, 1993.

[16] R. W. Freund and H. Zha. A look-ahead algorithm for the solution of general Hankel systems. *Numerische Mathematik*, 64:295–321, 1993.

[17] K. Gallivan, E. J. Grimme, and P. Van Dooren. Asymptotic waveform evaluation via a Lanczos method. *Appl. Math. Lett.*, 7:75–80, 1994.

[18] K. Gallivan, E. J. Grimme, and P. Van Dooren. A rational Lanczos algorithm for model reduction. *Numerical Algorithms*, 12:33–63, 1996.

[19] W. B. Gragg. Matrix interpretations and applications of the continued fraction algorithm. *Rocky Mountain Journal of Mathematics*, 4:213–225, 1974.

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

[21] E. J. Grimme, D. C. Sorensen, and P. Van Dooren. Model reduction of state space systems via an implicitly restarted Lanczos method. *Numerical Algorithms*, 12:1–32, 1996.

[22] T. V. Nguyen and Jing Li. Multipoint Pade' approximation using a rational block Lanczos algorithm. In *Technical Digest of the 1997 IEEE/ACM International Conference on Computer-Aided Design*, pages 72–75. IEEE Computer Society Press, Los Alamitos, California, 1997.

[23] A. Odabasioglu, M. Celik, and L. T. Pileggi. Prima: passive reduced-order interconnect macromodeling algorithm. In *Technical Digest of the 1997 IEEE/ACM International Conference on Computer-Aided Design*, pages 58–65. IEEE Computer Society Press, Los Alamitos, California, 1997.

[24] L. T. Pillage and R. A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Transactioins on Computer-Aided Design*, 9:352–366, 1990.

[25] A. E. Ruehli. Equivalent circuit models for three-dimensional multiconductor systems. *IEEE Trans. Microwave Theory and Tech.*, 22:216–221, 1974.

[26] Axel Ruhe. The two-sided Arnoldi algorithm for nonsymmetric eigenvalue problems. In B. Kågström and A. Ruhe, editors, *Matrix Pencils, LNM 973*, pages 104–120. Springer-Verlag, Berlin Heidelberg New York, 1983.

[27] Axel Ruhe. Rational Krylov sequence methods for eigenvalue computation. *Lin. Alg. Appl.*, 58:391–405, 1984.

[28] Axel Ruhe. Rational Krylov algorithms for nonsymmetric eigenvalue problems, II: Matrix pairs. *Lin. Alg. Appl.*, 197/198:283–296, 1994.

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

[30] Axel Ruhe. Rational Krylov, a practical algorithm for large sparse nonsymmetric matrix pencils. *SIAM J. Sci. Comp.*, 19:1535–1551, 1998.

[31] Axel Ruhe and Daniel Skoogh. Rational Krylov algorithms for eigenvalue computations and model reduction. In *B. Kågström et al (eds),* Applied Parallel Computing PARA'98*, Lecture Notes in Computer Science, No. 1541*, pages 491–502. Springer-Verlag, 1998. To be published in December 1998.

[32] Y. Saad. *Numerical Methods For Large Eigenvalue Problems.* Manchester University Press, 1992.

[33] L. M. Silveira, M. Kamon, I. Elfadel, and J. White. A coordinate-transformed Arnoldi algorithm for generating guaranteed stable reduced-order models of RLC circuits. In *Technical Digest of the 1996 IEEE/ACM International Conference on Computer-Aided Design*, pages 288–294. IEEE Computer Society Press, Los Alamitos, California, 1996.

[34] Daniel Skoogh. An implementation of a parallel rational Krylov algorithm. Licentiate Thesis, Chalmers University of Technology, Göteborg, Sweden, 1996.

[35] D.C. Sorensen. Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM J. Matrix Anal. Applic.*, 13:357–385, 1992.

[36] G. W. Stewart and Ji Guang Sun. *Matrix Perturbation Theory.* Academic Press, 1990.

[37] D. R. Taylor. *Analysis of the Look Ahead Lanczos Algorithm.* PhD thesis, University of California at Berkeley, 1982.