# A global Arnoldi method for the model reduction of second-order structural dynamical systems

Thomas Bonin[b], Heike Faßbender[a], Andreas Soppa[*,a], Michael Zaeh[b]

[a] *TU Braunschweig, Carl-Friedrich-Gauß Fakultät, Institut Computational Mathematics, AG Numerik, 38092 Braunschweig, Germany*
[b] *TU Muenchen, Institut für Werkzeugmaschinen und Betriebswissenschaften, 85748 Garching, Germany*

**Abstract**

In this paper we consider the reduction of second-order dynamical systems with multiple inputs and multiple outputs (MIMO) arising in the numerical simulation of mechanical structures. Undamped systems as well as systems with proportional damping are considered. In commercial software for the kind of application considered here, modal reduction is commonly used to obtain a reduced system with good approximation abilities of the original transfer function in the lower frequency range. In recent years new methods to reduce dynamical systems based on (block) versions of Krylov subspace methods emerged. This work concentrates on the reduction of second-order MIMO systems by the global Arnoldi method, an efficient extension of the standard Arnoldi algorithm for MIMO systems. In particular, a new model reduction algorithm for second order MIMO systems is proposed which automatically generates a reduced system of given order approximating the transfer function in the lower range of frequencies. It is based on the global Arnoldi method, determines the expansion points iteratively and the number of moments matched per expansion point adaptively. Numerical examples comparing our results to modal reduction and reduction via the block version of the rational Arnoldi method are presented.

*Key words:* Model Order Reduction, Simulation, Krylov Subspace, Global Arnoldi Algorithm, Moment Matching
MSC[2009] 65F30, 70J50

## 1. Introduction

In the context of the numerical simulation of machine tools second-order dynamical systems of the form

$$M\ddot{x}(t) + D\dot{x}(t) + Kx(t) = Fu(t), \qquad y(t) = C_p x(t) + C_v \dot{x}(t) \tag{1}$$

arise, where $M, K, D \in \mathbb{R}^{n\times n}$, $F \in \mathbb{R}^{n\times m}$, $C_p, C_v \in \mathbb{R}^{q\times n}$, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^q$. Only undamped or proportional damped systems are considered, that is, either $D = 0$ or $D = \alpha M + \beta K$ for real $\alpha, \beta$.

The system matrices considered are large, sparse, and non-symmetric. The matrix $K$ is non-singular. The mass matrix $M$ may be singular. In that case one obtains a system of differential algebraic equations. In general, $m$ and $q$ will be larger than one, so that the system is multi-input multi-output (MIMO). All of this accounts for unacceptable computational and resource demands in simulation and control of these models. In order to reduce these demands to acceptable computational times, usually model order reduction techniques are employed which generate a reduced order model that captures the essential dynamics of the system and preserves its important properties. That is, one tries to find a second order system of reduced dimension $r \ll n$

$$\hat{M}\ddot{\hat{x}}(t) + \hat{D}\dot{\hat{x}}(t) + \hat{K}\hat{x}(t) = \hat{F}u(t), \qquad \hat{y}(t) = \hat{C}_p\hat{x}(t) + \hat{C}_v\dot{\hat{x}}(t), \tag{2}$$

*Corresponding author
*Email addresses:* thomas.bonin@iwb.tum.de (Thomas Bonin), h.fassbender@tu-braunschweig.de (Heike Faßbender), a.soppa@tu-braunschweig.de (Andreas Soppa), michael.zaeh@iwb.tum.de (Michael Zaeh)

which approximates the original system in some sense, where $\hat{M}, \hat{D}, \hat{K} \in \mathbb{R}^{r\times r}$, $\hat{F} \in \mathbb{R}^{r\times m}$, $\hat{C}_p, \hat{C}_v \in \mathbb{R}^{q\times r}$, $\hat{x}(t) \in \mathbb{R}^r$, $u(t) \in \mathbb{R}^m$, $\hat{y}(t) \in \mathbb{R}^q$.

In the last years various methods to reduce second-order dynamical systems have been proposed, see, e.g., [2, 4, 28]. As model reduction of linear first-order systems is much further developed and understood, it is tempting to transform the original second-order system (1) to a mathematically equivalent first-order system

$$\underbrace{\begin{bmatrix} -K & 0 \\ 0 & M \end{bmatrix}}_{E} \underbrace{\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \end{bmatrix}}_{\dot{z}(t)} = \underbrace{\begin{bmatrix} 0 & -K \\ -K & -D \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}}_{z(t)} + \underbrace{\begin{bmatrix} 0 \\ F \end{bmatrix}}_{B} u(t), \quad y(t) = \underbrace{\begin{bmatrix} C_p & C_v \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}}_{z(t)}, \tag{3}$$

where $E, A \in \mathbb{R}^{2n\times 2n}$, $B \in \mathbb{R}^{2n\times m}$, $C \in \mathbb{R}^{q\times 2n}$, $z(t) \in \mathbb{R}^{2n}$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^q$. Various other linearizations have been proposed in the literature, see, e.g., [19, 25, 31]. The linearization (3) is usually preferred as it is symmetry preserving in case $K, M, D$ are symmetric. The system considered here is non-symmetric, so one of the various other possible linearizations could be used instead. Note that the transformation process doubles the dimension of the system. The corresponding reduced system is of the form

$$\hat{E}\dot{\hat{z}}(t) = \hat{A}\hat{z}(t) + \hat{B}u(t) \qquad \hat{y}(t) = \hat{C}\hat{z}(t), \tag{4}$$

where $\hat{E}, \hat{A} \in \mathbb{R}^{r\times r}$, $\hat{B} \in \mathbb{R}^{r\times m}$, $\hat{C} \in \mathbb{R}^{q\times r}$, $\hat{z}(t) \in \mathbb{R}^r$, $u(t) \in \mathbb{R}^m$, $\hat{y}(t) \in \mathbb{R}^q$.

In the engineering context of our application modal reduction [3] is most common. Here we will consider projection based model reduction based on Krylov subspace methods. In the recent years various new Krylov subspace based methods to reduce first- and second-order systems have been proposed, see, e.g., [1, 10, 26] and the references therein. We will consider methods which generate matrices $V \in \mathbb{R}^{2n\times r}$ with $V^TV = I_r$ such that the reduced first-order system (4) is constructed by applying the Galerkin projection $\Pi = VV^T$ to (3)

$$\hat{E} = V^TEV, \quad \hat{A} = V^TAV, \quad \hat{B} = V^TB, \quad \text{and} \quad \hat{C} = CV. \tag{5}$$

Similarly, the reduced second-order system (2) is constructed by applying a Galerkin projection to (1) such that

$$\hat{M} = V^TMV, \quad \hat{D} = V^TDV, \quad \hat{K} = V^TKV, \quad \hat{F} = V^TF, \quad \hat{C}_p = C_pV, \quad \text{and} \quad \hat{C}_v = C_vV, \tag{6}$$

where $V \in \mathbb{R}^{n\times r}$ with $V^TV = I_r$. The matrix $V$ can be constructed iteratively by employing Krylov subspace algorithms, in particular the block Arnoldi algorithm. It is well-known that Krylov subspace based methods are not guaranteed to yield reduced order models with the best overall performance in the entire frequency domain; only local approximation around the expansion point can be expected. Therefore, multi point moment matching methods have been introduced [12, 14, 17], see Section 2 for a short review. In [15] the choice of expansion points is discussed, in [16] an algorithm choosing the expansion points iteratively, called Iterative Rational Krylov Algorithm (IRKA) and in [11, 24] adaptive multi point moment matching methods have been proposed. The global Arnoldi method [21] is similar to the standard Arnoldi method except that the standard inner product is replaced by the inner product $\langle Y, Z\rangle_F = \operatorname{trace}(Y^TZ)$ where $Y, Z \in \mathbb{R}^{n\times s}$. The associated norm is the Frobenius norm $\|\cdot\|_F$. The global Arnoldi algorithm constructs an $F$-orthonormal basis $V_1, V_2, \ldots, V_k$ of the Krylov subspace $\mathcal{K}_k(\Psi, \Upsilon), \Psi \in \mathbb{R}^{n\times n}, \Upsilon \in \mathbb{R}^{n\times s}$. Here a system of vectors (matrices in $\mathbb{R}^{n\times s}$) is said to be $F$-orthonormal if it is orthonormal with respect to $\langle\cdot,\cdot\rangle_F$. It has been used for model reduction of first-order systems (3), see [5, 6, 7, 8]. In Section 3 a short introduction of the global Arnoldi method is presented. Further its extension to model reduction of second order MIMO systems is discussed. In the context of the global Arnoldi algorithm, an adaptive-order global Arnoldi algorithm has been proposed [5, 24]. This algorithm adaptively determines the number of expansions for a fixed set of expansion points. Here we propose a combination of this algorithm and a modified version of IRKA [17] to reduce second-order MIMO systems. The algorithm is based on the global Arnoldi method, determines the expansion points iteratively and the number of moments matched per expansion point adaptively. Numerical experiments are given in Section 4.

## 2. Model reduction using block Arnoldi type methods

A $k$-th order Krylov subspace is defined by

$$\mathcal{K}_k(P, q) = \operatorname{span}\{q, Pq, P^2q, \ldots, P^{k-1}q\}, \tag{7}$$

where $P \in \mathbb{R}^{n\times n}$ and $q \in \mathbb{R}^n$. The Arnoldi method [13, 29] applied to the pair $(P, q)$ produces a matrix $V$ with orthonormal columns which span the Krylov subspace $\mathcal{K}_k(P, q)$ (in case no breakdown occurs during the computation).

In order to be able to treat MIMO systems, we will need to consider block Krylov subspaces

$$\mathcal{K}_k(P, Q) = \text{span}\{Q, PQ, P^2Q, \ldots, P^{k-1}Q\}, \tag{8}$$

where $P \in \mathbb{R}^{n\times n}$ and the columns of $Q \in \mathbb{R}^{n\times \ell}$ are linearly independent. Such a block Krylov subspace with $\ell$ starting vectors (assembled in $Q$) can be considered as a union of $\ell$ Krylov subspaces defined for each starting vector. Usually, the computation of an orthonormal basis for the block Krylov subspace $\mathcal{K}_k(P, Q)$ can be achieved by employing a block Arnoldi algorithm, see Algorithm 1 [27].

Due to the fact that the matrix $P$ may be singular the computation of $Z = PV_j$ in line 5 of Algorithm 1 may yield in a matrix $Z$ with linear dependent column vectors. Therefore a deflation of linear dependent vectors of $Z$ is necessary. For that purpose in line 6 of Algorithm 1 a rank revealing $QR$ factorization of $Z = WR$ is computed. In case $Z$ has the numerical rank $s$, in line 7 the unnecessary trailing part of $W$ is deleted.

**Algorithm 1** Block Arnoldi method

**Input:** matrices $P, Q$; $k$
**Output:** transformation matrix $V$

```
1:  function [V] = Block_Arnoldi(P, Q, k)
2:  compute the (reduced) QR factorization V_1 R = Q
3:  V = [V_1]
4:  for j = 1 : k do
5:     Z = P V_j
6:     compute the rank revealing QR factorization WR = Z,  rank(Z) = s
7:     W = W(:, 1 : s)
8:     for i = 1 : j do
9:        H_{i,j} = V_i^T W
10:       W = W − V_i H_{i,j}
11:    end for
12:    compute the (reduced) QR factorization V_{j+1} H_{j+1,j} = W
13:    V = [V V_{j+1}]
14: end for
```

As in the standard Arnoldi algorithm re-orthogonalization is necessary in order to keep the computed columns of $V$ orthogonal. The following relation will hold

$$PV_{[k]} = V_{[k]}H_{[k]} + [0, \ldots, 0, V_{k+1}H_{k+1,k}]$$

where $H_{[k]}$ is a block upper Hessenberg matrix. The columns of $V_{[k]} = [V_1, V_2, \ldots, V_k] \in \mathbb{R}^{n\times r}$ with $r = k \cdot \ell$ and $V_j \in \mathbb{R}^{n\times \ell}$ are an orthogonal basis for the block Krylov subspace $\mathcal{K}_k(P, Q)$ provided none of the upper triangular matrices $H_{j+1,j}$ in Algorithm 1 are rank-deficient.

### *2.1. First order systems*

The transfer function of a first-order system (3) is the linear mapping of the Laplace transformation of the input to the output

$$H(s) = C(sE - A)^{-1}B.$$

After expansion in a Laurent expansion series around an expansion point $s_0$ one obtains the moments $h_j(s_0), \quad j = 0, .., \infty$ of the transfer function

$$\begin{aligned} H(s) &= \sum_{j=0}^{\infty} h_j(s_0)(s - s_0)^j, \\ \text{where} \quad h_j(s_0) &= -C\,[(A - s_0E)^{-1}E]^j(A - s_0E)^{-1}B, \end{aligned}$$

see, e.g., [1]. Consider the block Krylov subspace $\mathcal{K}_k(P, Q)$ (8) for

$$P = (A - s_0E)^{-1}E \in \mathbb{R}^{2n\times 2n} \text{ and } Q = -(A - s_0E)^{-1}B \in \mathbb{R}^{2n\times m}.$$

Assume that an orthogonal basis for this block Krylov subspace is generated using the block Arnoldi method. Here, and in the rest of the paper, we will assume that no breakdown occurred during the computations so that the column-space of the resulting matrix $V$ spans the block Krylov subspace $\mathcal{K}_k(P, Q)$. Applying the similarity transformation (5) (with $V = V_{[k]} \in \mathbb{R}^{2n\times r}$, $V^TV = I_r$ and $r = k \cdot m$) yields a reduced system whose transfer function matches at least the first $k$ moments of the transfer function of the original system [1]. That is, at least the first $k$ moments $\hat{h}_j(s_0)$, of the transfer function $\hat{H}(s)$ of the reduced system (4) equal the first moments $h_j(s_0)$, of the transfer function $H(s)$ of the original system (3) at expansion point $s_0$:

$$h_j(s_0) = \hat{h}_j(s_0), \quad j = 0, 1, \ldots, k-1.$$

An alternative is to use more than one expansion point, this leads to multi point moment matching methods called Rational Krylov methods [14]. Assume that $\hat{\imath}$ expansion points $s_i$, $i = 1, 2, \ldots, \hat{\imath}$ are considered. The column vectors of the matrix $V$ are determined from the $\hat{\imath}$ block Krylov subspaces generated by

$$P = (A - s_iE)^{-1}E \text{ and } Q = -(A - s_iE)^{-1}B, \quad i = 1, 2, \ldots, \hat{\imath}. \tag{9}$$

From each of these subspaces, $r_i = k_i \cdot m$ column vectors are used to generate $V \in \mathbb{R}^{2n\times r}$ (with $r = \sum_{i=1}^{\hat{\imath}} r_i$). Then at least $k_i$ moments are matched per expansion point $s_i$,

$$h_j(s_i) = \hat{h}_j(s_i), \quad j = 0, 1, \ldots, k_i - 1, \quad i = 1, 2, \ldots, \hat{\imath}, \tag{10}$$

if the reduced system is generated by (5).

In [17] the choice of expansion points $s_i, i = 1, \ldots, \hat{\imath}$ is discussed. Starting from an initial set of expansion points a reduced order system is determined. Then a new set of expansion points is chosen as $s_i = -\lambda_i, i = 1, \ldots, \hat{\imath}$ where $\lambda_i$ are the eigenvalues of the matrix pencil $\hat{E} - \lambda\hat{A}$ with $\hat{E}, \hat{A}$ as in (4), ordered such that $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_r|$. This algorithm is called Iterative Rational Krylov Algorithm (IRKA) [17]. Here a modified version of IRKA is proposed: A new set of expansion points is chosen from the set of eigenvalues ordered by their imaginary part such that $|\mathrm{Im}(\lambda_1)| \leq |\mathrm{Im}(\lambda_2)| \leq \ldots \leq |\mathrm{Im}(\lambda_r)|$. Starting from $s_1 = \mathrm{Im}(\lambda_1) \cdot \imath$ ($\imath = \sqrt{-1}$) the next expansion points $s_i, i = 2, \ldots, \hat{\imath}$ are chosen as $s_i = \mathrm{Im}(\lambda_i) \cdot \imath$. As expansion points lying a bit apart yield better approximation results, this choice of the expansion points is refined such that in addition we require $|s_{i-1} - s_i| > \epsilon$, where $\epsilon$ is chosen by the user and defines a (minimum) distance between two adjacent expansion points. Hence, if $|s_2 - s_1| \leq \epsilon$, we do not choose $s_2 = \mathrm{Im}(\lambda_2) \cdot \imath$, but test $|s_2 - s_1|$ for $s_2 = \mathrm{Im}(\lambda_3) \cdot \imath$. If this is still small than $\epsilon$, we next test for $s_2 = \mathrm{Im}(\lambda_4) \cdot \imath$, until we have found an $\lambda_k$ such that $s_2 = \mathrm{Im}(\lambda_k) \cdot \imath$ yields $|s_2 - s_1| > \epsilon$. Next we choose $s_3$ is the same fashion starting from $\lambda_{k+1}$ such that $|s_3 - s_2| > \epsilon$. Unlike IRKA, this method cannot be guaranteed to be $\mathcal{H}_2$-optimal but after a few iterations good approximation results of the transfer function, especially for low frequencies, are obtained. The approach described here is summarized as the Modified Iterative Rational Arnoldi algorithm (MIRA) in Algorithm 2. In each iteration, for each expansion point $s_i$ the corresponding projection matrix $V_i$ is constructed. The final projection matrix $V$ is build up from the $V_i$, $V = [V_1\ V_2 \ldots V_{\hat{\imath}}]$. As $V$ may have linear dependend colums and is not necessarily orthogonal, a $QR$ decomposition is used to orthogonalize $V$ and to remove linear dependend columns.

In [24] a strategy for an adaptive-order model reduction method based on the Arnoldi method is discussed. Given a fixed set of expansion points $s_i, i = 1, \ldots, \hat{\imath}$ and the reduced dimension $r$, an adaptive scheme for automatically choosing $r_i$ about each expansion point $s_i$ is proposed, see Chapter 3.2.

### *2.2. Second order systems*

The transfer function of a second-order system is given by the Laplace transformation of (1):

$$H(s) = (C_p + sC_v)(s^2M + sD + K)^{-1}F.$$

Shifting it with $s = s_0 + \sigma$, we have

$$H(s) = (\tilde{C}_p + \sigma C_v)(\sigma^2M + \sigma\tilde{D} + \tilde{K})^{-1}F,$$

**Algorithm 2** Modified Iterative Rational Arnoldi (MIRA)

**Input:** system matrices $E, A, B, C$ resp. $M, D, K, F, C_p$; initial expansion points $s_i$, $i = 1, \ldots, \hat{i}$; $k_i$, $i = 1, \ldots, \hat{i}$; tolerance $tol$; $\epsilon$

**Output:** reduced system of order $r = \sum_{i=1}^{\hat{i}} k_i \cdot m$

set $s_i^{old}$ so that $\max_{i \in \{1,2,\ldots \hat{i}\}} |s_i^{old} - s_i| > tol$
**while** $\max_{i \in \{1,2,\ldots \hat{i}\}} |s_i^{old} - s_i| > tol$ **do**
  $V = [\,]$
  **for** $i = 1 : \hat{i}$ **do**
    compute the $LU$ factorization: $LU = (A - s_i E)$ resp. $LU = -(s_i^2 M + s_i D + K)$
    $P = U \backslash (L \backslash E)$ resp. $P = U \backslash (L \backslash M)$
    $Q = -U \backslash (L \backslash B)$ resp. $Q = -U \backslash (L \backslash F)$
    $V_i = \text{Block\_Arnoldi}(P, Q, k_i, \tau)$
  **end for**
  $\tilde{V} = [\, V_1 \; V_2 \; \cdots \; V_{\hat{i}} \,]$
  compute the rank revealing $QR$ factorization $VR = \tilde{V}, \quad rank(\tilde{V}) = s$
  $V = V(:, 1 : s)$
  compute reduced system matrices with $V$ by (5) resp. (6)
  compute the eigenvalues $\lambda_j$, $j = 1 \ldots, r$ of the reduced system ordered such that $|\text{Im}(\lambda_1)| \leq |\text{Im}(\lambda_2)| \leq \ldots \leq |\text{Im}(\lambda_r)|$
  $s_i^{old} \leftarrow s_i$, for $i = 1, \ldots, \hat{i}$
  choose new expansion points $s_i$ as explained at the end of Section 2.1
**end while**
compute the congruence transformation with $V$ by (5) resp. (6).

where $\tilde{D} = 2s_0 M + D$, $\tilde{K} = s_0^2 M + s_0 D + K$ and $\tilde{C}_p = C_p + s_0 C_v$. After expansion in a Laurent expansion series around expansion point $s_0$ one obtains the moments $h_j(s_0)$, $j = 0, .., \infty$ of the transfer function

$$H(s) = \sum_{j=0}^{\infty} h_j(s_0)(s - s_0)^j,$$

where

$$\begin{aligned} h_0(s_0) &= \tilde{C}_p \, \xi_0(s_0) \\ h_j(s_0) &= C_v \, \xi_{j-1}(s_0) + \tilde{C}_p \, \xi_j(s_0), \; j = 1, 2, \ldots \end{aligned}$$

and

$$\begin{aligned} \xi_0(s_0) &= \tilde{K}^{-1} F, \\ \xi_1(s_0) &= \tilde{K}^{-1} (-\tilde{D} \, \xi_0(s_0)), \\ \xi_j(s_0) &= \tilde{K}^{-1} (-\tilde{D} \, \xi_{j-1}(s_0) - M \, \xi_{j-2}(s_0)), \; j = 2, 3, \ldots, \end{aligned}$$

see, e.g., [30]. (In an abuse of notation, we denote the transfer function (the moments) of the first- and the second-order system by $H$ ($h_j$). It will be clear from the context which one is refered to.) Recall, that here we consider only the special cases of undamped or proportional damped systems. For these cases, as proposed in [4], the block Krylov subspace $\mathcal{K}_k(P, Q)$ with

$$P = -(s_0^2 M + s_0 D + K)^{-1} M \;\text{ and }\; Q = (s_0^2 M + s_0 D + K)^{-1} F$$

is used to generate the transformation matrix $V$. In [4, 9] it was shown, that the transfer function of the system reduced by applying the congruence transformation (6) with $V$ matches at least the first $k$ moments of the transfer function of the original system.

If more than one expansion point is used, one is interested in matching at least $k_i$ moments for each expansion point $s_i, i = 1, \ldots, \hat{i}$. Therefore, the $r = \sum_{i=1}^{\hat{i}} r_i$, $r_i = k_i \cdot m$ column vectors of matrix $V$ are determined from the block Krylov subspaces generated by

$$P = -(s_i^2 M + s_i D + K)^{-1} M \quad \text{and} \quad Q = (s_i^2 M + s_i D + K)^{-1} F. \tag{11}$$

The transfer function of the system reduced by applying the congruence transformation (6) with $V$ matches at least the first $k_i$ moments of the transfer function of the original system per expansion point $s_i$ [4]. As in the case of first-order systems, the iterative approach MIRA for the choice of the expansion points $s_i$ can be used. The pseudo-code of MIRA is given as Algorithm 2.

## 3. Model Reduction using the Global Arnoldi Method

The global Krylov method was first proposed in [18, 21] for solving linear equations with multiple right hand sides and Lyapunov equations. Applications to model order reductions of first-order systems are studied in [5, 6, 7, 8]. It was also used for solving large Lyapunov matrix equations [22]. The global Krylov method is similar to the standard Krylov method except that the standard inner product is replaced by the inner product $\langle Y, Z \rangle_F = \text{trace}(Y^T Z), Y, Z \in \mathbb{R}^{n \times \ell}$. The associated norm is the Frobenius norm $\| \cdot \|_F$. A system of vectors (matrices) in $\mathbb{R}^{n \times \ell}$ is said to be $F$-orthonormal if it is orthonormal with respect to $\langle \cdot, \cdot \rangle_F$.

The global Arnoldi algorithm [21] (see Algorithm 3) constructs an $F$-orthonormal basis $V_1, V_2, \ldots, V_k$ with $V_j \in \mathbb{R}^{n \times \ell}$ of the Krylov subspace $\mathcal{K}_k(P, Q), P \in \mathbb{R}^{n \times n}, Q \in \mathbb{R}^{n \times \ell}$; i.e.,

$$\begin{aligned} \langle V_i, V_j \rangle_F &= 0 \qquad i \neq j,\ i, j = 1, \ldots, k, \\ \langle V_j, V_j \rangle_F &= 1. \end{aligned}$$

As in the case of the block Arnoldi method the matrix $P$ may be singular so that the computation of $Z = PV_j$ in line 5 of Algorithm 3 may yield in a matrix $Z$ with linear dependent column vectors. Therefore a deflation of linear dependent vectors of $Z$ is necessary.

**Algorithm 3** Global Arnoldi method

**Input:** matrices $P, Q$; $k$
**Output:** transformation matrix $V$

```
1:  function [V] = Global_Arnoldi( P, Q, k)
2:  V_1 = Q/||Q||_F
3:  V = [V_1]
4:  for j = 1 : k do
5:     Z = P V_j
6:     compute the rank revealing QR factorization WR = Z,  rank(Z) = s
7:     W = W(:, 1 : s)
8:     for i = 1 : j do
9:        h_ij = trace(V_i^T W)
10:       W = W − h_ij V_i
11:    end for
12:    h_{j+1,j} = ||W||_F,  V_{j+1} = W/h_{j+1,j}
13:    V = [V V_{j+1}]
14: end for
```

Comparing both algorithms, the block Arnoldi method requires a $QR$ factorization of $W$ in every step (line 2 and 12), while the global Arnoldi method only needs the division by the Frobenius norm of $W$. Moreover, the block Arnoldi method requires the computation of $V^T W$ in line 9, while the global Arnoldi method only needs the trace of that matrix. Finally, in line 10 of the algorithm, the block Arnoldi method requires a matrix-matrix-product, while the global Arnoldi method only needs a scalar-matrix-product. Table 1 gives an estimate for the flop count for $k$ iterations

Table 1: Flop count for $k$ steps of the block Arnoldi and global Arnoldi algorithm

| Flops | block Arnoldi | | global Arnoldi | |
|---|---|---|---|---|
| Matrix-matrix multiplication $n \times n$ with $n \times \ell$ matrices | $2 \cdot k \cdot n^2 \cdot \ell$ | [line 5] | $2 \cdot k \cdot n^2 \cdot \ell$ | [line 5] |
| Matrix-matrix multiplication, resp. the trace of $\ell \times n$ and $n \times \ell$ matrices | $\frac{k(k+1)}{2} \cdot 2 \cdot n \cdot \ell^2$ | [line 9] | $\frac{k(k+1)}{2}(2 \cdot n \cdot \ell + \ell)$ | [line 10] |
| Matrix-matrix multiplication $n \times \ell$ with $\ell \times \ell$ matrices, resp. matrix multiplication | $\frac{k(k+1)}{2} \cdot 2 \cdot n \cdot \ell^2$ | [line 10] | $\frac{k(k+1)}{2}(n \cdot \ell)$ | [line 11] |
| QR on $n \times \ell$ matrix, resp. Frobenius norm of $n \times \ell$ matrix | $3(k+1) \cdot \ell^3 \cdot (n - \frac{\ell}{3})$ | [line 2, 12] | $(k+1) \cdot (2 \cdot n \cdot \ell + \ell)$ | [line 2, 13] |

of the block Arnoldi resp. the global Arnoldi algorithm assuming that the input matrices $P$ and $Q$ are of the size $n \times n$ and $n \times \ell$ and using the flop count for the Givens QR algorithm as given in [13]. The global Arnoldi is advantageous if the number of iterations $k$ and the number of column vectors $\ell$ of matrix $Q$ are large, because the block Arnoldi algorithm becomes expensive as $\ell$ is large and $k$ increases.

If $\ell = 1$, the global Arnoldi algorithm reduces to the standard Arnoldi algorithm. Let $V_{(k)} = [V_1\ V_2\ \dots\ V_k] \in \mathbb{R}^{n \times r}$, $r = k \cdot \ell$ and $H_k$ the corresponding $k \times k$ upper Hessenberg matrix. The following relation will hold

$$V_{(k)} = V_{(k)}(H_k \otimes I_s) + h_{k+1,k}[0, \dots, 0, V_{k+1}].$$

Here $\otimes$ denotes the Kronecker product of two matrices $X \in \mathbb{R}^{u \times u}$ and $Y \in \mathbb{R}^{v \times v}$

$$X \otimes Y = \begin{pmatrix} x_{11}Y & x_{12}Y & \cdots & x_{1u}Y \\ x_{21}Y & x_{22}Y & \cdots & x_{2u}Y \\ \vdots & \vdots & \vdots & \vdots \\ x_{u1}Y & x_{u2}Y & \cdots & x_{uu}Y \end{pmatrix} = [x_{ij}Y]_{i,j=1}^{u} \in \mathbb{R}^{uv \times uv}.$$

Note that the Hessenberg matrix $H_k$ in the global Arnoldi algorithm is of dimension $k \times k$ while for the block Arnoldi algorithm $H_{[k]}$ is a block Hessenberg matrix of dimension $\ell k \times \ell k$. Moreover, as noted in [21], linear dependence between the column vectors of the generated matrices $V_i, i = 1, \dots, k$ has no effect on the global Arnoldi algorithm. The major difference between the global and the block Arnoldi algorithm lies in the computed basis of the Krylov subspace: the global Arnoldi algorithm allows to generate the $F$-orthonormal basis, while the block Arnoldi algorithm constructs an orthogonal basis. The matrices constructed by the block Arnoldi algorithm have their columns mutually orthogonal. Finally, note that the block Arnoldi algorithm constructs an orthonormal basis of the block Krylov subspace $\mathcal{K}_k(P, Q) \subset \mathbb{R}^n$ while the global Arnoldi algorithm generates an $F$-orthonormal basis of the matrix Krylov subspace $\mathcal{K}_k(P, Q) \subset \mathbb{R}^{n \times \ell}$.

If the global Arnoldi method and the block Arnoldi method are applied to the same matrix pair $(P, Q)$, the resulting matrices $V_{(k)}$ and $V_{[k]}$ both span the same Krylov subspace $\mathcal{K}_k(P, Q)$. The orthonormalization of the bases vectors of $\mathcal{K}_k(P, Q)$ is the only difference whether constructed by the the block- or the global-Arnoldi method. In [14, Chapter 3] it is shown that the moment matching property (10) does only depend on the fact that the columns of $V = V_{[k]}$ resp. $V = V_{(k)}$ span the Krylov subspace $\mathcal{K}_k(P, Q)$ with $(P, Q)$ as in (9) resp. in (11). It does not depend on the way $V$ is computed or whether its columns have a certain additional property. Hence, the moment matching property holds for reduction methods based on the global Arnoldi algorithm as well as for reduction methods based on the block Arnoldi algorithm.

### *3.1. First order systems*

The global Arnoldi method is the standard Arnoldi method applied to the matrix pair $((I \otimes P^j), \text{vec}\,(Q)) = (\Psi, \Upsilon)$, where the inner product is replaced by $\langle X, Y \rangle_F = \text{vec}\,(X)^T \text{vec}\,(Y) = \text{trace}(X^T Y)$. Here vec $(\cdot)$ denotes the usual vector

stacking operation [20]

$$\text{vec}\,(Z) = (Z_{*1}\, Z_{*2} \cdots Z_{*v})^T \in \mathbb{R}^{uv}, \qquad Z = [Z_{*1}\, Z_{*2} \cdots Z_{*v}] \in \mathbb{R}^{u\times v},\ Z_{*j} \in \mathbb{R}^u,\ j = 1, \ldots, v.$$

For vec $(P^j, Q)$ we have

$$\text{vec}\,(P^j, Q) = (I \otimes P^j)\,\text{vec}\,(Q).$$

Therefore, with $P$ and $Q$ as in (9), the moments of the system (3) can be associated with a vector-function.

Assume that an $F$-orthogonal basis for the Krylov subspace $\mathcal{K}_k(P, Q)$ for $(P, Q)$ as in (9) is generated using the global Arnoldi method. Applying the projection $\Pi = V_{(k)}V_{(k)}^\dagger$ to (3) leads to the reduced system with

$$\hat{E} = V_{(k)}^\dagger E V_{(k)}, \qquad \hat{A} = V_{(k)}^\dagger A V_{(k)}, \qquad \hat{B} = V_{(k)}^\dagger B, \qquad \text{and} \qquad \hat{C} = C V_{(k)}. \tag{12}$$

As $V_{(k)}$ is $F$-orthonormal, the pseudo-inverse $V_{(k)}^\dagger = (V_{(k)}^T V_{(k)})^{-1} V_{(k)}^T$ has to be used instead of $V_{(k)}^T$. The computation of the pseudo-inverse $V_{(k)}^\dagger$ may be numerically unstable. In order to avoid this, a $QR$ factorization $V_{(k)} = VR$ is computed such that $V^T V = I_r$ and

$$V_{(k)}^\dagger = (V_{(k)}^T V_{(k)})^{-1} V_{(k)}^T = ((VR)^T VR)^{-1}(VR)^T = R^{-1}V^T. \tag{13}$$

With this, the projection $\Pi = V_{(k)}V_{(k)}^\dagger$ leads to the reduced system

$$\hat{E}\dot{\hat{z}}(t) = \hat{A}\hat{z}(t) + \hat{B}u(t) \qquad \hat{y}(t) = \hat{C}\hat{z}(t), \tag{14}$$

with

$$\hat{E} = R^{-1}V^T E V, \qquad \hat{A} = R^{-1}V^T A V, \qquad \hat{B} = R^{-1}V^T B, \qquad \text{and} \qquad \hat{C} = CV. \tag{15}$$

Multiplying the first equation of the reduced system (14) on the left hand side by $R$ leads to the reduced system (5) constructed by applying the Galerkin Projection $\Pi = VV^T$.

The algorithm MIRA (see Algorithm 2) is easily modified to make use of the global Arnoldi method instead of the block Arnoldi algorithm. For that in line 8 of Algorithm 2 the block Arnoldi algorithm (Algorithm 1) is replaced by the global Arnoldi algorithm (Algorithm 3). The resulting algorithm is called Iterative Rational Global Arnoldi algorithm (IRGA).

The only difference between the algorithm IRGA and the algorithm MIRA is the use of the global Arnoldi method instead of the block Arnoldi method. All other computational steps of MIRA and IRGA cause the same costs. With the observations in [5] the number of flops and the cost of IRGA are less compared with MIRA. So algorithm IRGA can be considered as an efficient alternative to algorithm MIRA.

If applied to the same matrix pair $(P, Q)$, the first moments $\hat{h}_j(s_i)$ of the transfer function of the reduced system (4) are the same as those of the original system (3) no matter whether MIRA or IRGA was used. But even if the same initial expansion points are used, both algorithms may match different moments due to the iterative choice of the expansion points $s_i$.

### *3.2. Second order systems*

Assume that an $F$-orthogonal basis $V_{(k)}$ for the Krylov subspace $\mathcal{K}_k(P, Q)$ for $(P, Q)$ as in (11) is generated using the global Arnoldi method. The reduced system is then given by applying the projection $\Pi = V_{(k)}V_{(k)}^\dagger$ to (1) such that

$$\hat{M} = V_{(k)}^\dagger M V_{(k)}, \quad \hat{D} = V_{(k)}^\dagger D V_{(k)}, \quad \hat{K} = V_{(k)}^\dagger K V_{(k)}, \quad \hat{F} = V_{(k)}^\dagger F, \quad \hat{C}_p = C_p V_{(k)}, \quad \text{and} \quad \hat{C}_v = C_v V_{(k)}. \tag{16}$$

As in the case of first order systems the computation of the pseudo-inverse $V_{(k)}^\dagger$ may be numerically unstable. In order to avoid this, the $QR$ factorization $V_{(k)} = VR$ is computed such that $V^T V = I_r$ and $V_{(k)}^\dagger = R^{-1}V^T$. With $V$ the projection (6) can be used to obtain the reduced second order system.

If the global Arnoldi method and the block Arnoldi method are applied to the matrix pair $(P, Q)$ as in (11) the moment matching property discussed for the block Arnoldi based model reduction is here still valid. The first moments $\hat{h}_j(s_i)$ at the expansion points $s_i$, $i = 1, \ldots, \hat{i}$, of the transfer function of the reduced system (16) are the same as those of the original system (1).

In the previous section, the $r_i$ were chosen as $m \cdot k_i$ so that for each expansion point at least $k_i$ moments are matched. Here a different approach as suggested in [5, 23] is used which adaptively determines the $r_i$. The Adaptive Order Rational Global Arnoldi (AORGA) algorithm describes an adaptive scheme for automatically choosing $r_i$ about each expansion point $s_i$ given a fixed set of expansion points $s_i$, $i = 1, \ldots, \hat{i}$ and the reduced dimension $r$. In the $j$-th iteration of AORGA an expansion point from the set of fixed expansion points corresponding to the maximum moment error will be chosen to compute $V_j$. Consequently, the corresponding reduced system will yield the greatest moment improvement among all reduced systems of the same order and the same set of expansion points.

Here an algorithm to reduce second-order systems combining this approach and IRGA is proposed. This algorithm, called Adaptive Iterative Rational Global Arnoldi algorithm for second order systems (AIRGA_2o), computes a reduced system by determining the expansion points iteratively and the number of matched moments per expansion point adaptively. The method is given in pseudo-code as Algorithm 4.

Assume that in the $j-1$-th iteration of AIRGA_2o the first $j_i - 1$ moments are matched at expansion point $s_i$. Here the results from [24] for first-order systems are adapted for second-order systems. In the $j$-th iteration of AIRGA_2o the $j_i$-th moment error $\hat{h}_{j_i}(s_i)$ at expansion point $s_i$, $i = 1, \ldots, \hat{i}$ of the second-order system is given by

$$\| h_{j_i}(s_i) - \hat{h}_{j_i}(s_i) \|_F = \| h_\pi^{(j-2)}(s_i)\, C_v\, R^{(j-2)}(s_i) + h_\pi^{(j-1)}(s_i)\, (C_p + s_i C_v)\, R^{(j-1)}(s_i) \|_F,$$

where $h_{j_i}(s_i)$ and $\hat{h}_{j_i}(s_i)$ are the $j_i$-th moments of the original resp. of the reduced second-order system at expansion point $s_i$. Moreover, $R^{(0)}(s_i) = -(s_i^2 M + s_i D + K)^{-1} F$, $h_\pi^{(0)}(s_i) = 1$ and $R^{(t)}(s_i)$, $h_\pi^{(t)}(s_i)$ are as stated in the $t$-th iteration of Algorithm 4 (lines 14 - 26) for $t = 1, \ldots, j-1$. In the $j$-th iteration of Algorithm 4 (line 11) this approach is used to determine the expansion point $\sigma_j$ corresponding to the maximum $j_i$-th moment error of the reduced second-order system at the expansion points $s_i$ by

$$\sigma_j = \max_{s_i} \| h_\pi^{(j-2)}(s_i)\, C_v\, R^{(j-2)}(s_i) + h_\pi^{(j-1)}(s_i)\, (C_p + s_i C_v)\, R^{(j-1)}(s_i) \|_F.$$

As in AORGA, in the $j$-th iteration of AIRGA_2o an expansion point from the set of iteratively determined expansion points corresponding to the maximum moment error will be chosen to compute $V_j$. Consequently, the corresponding reduced system will yield the greatest moment improvement among all reduced systems of the same order and the same set of iteratively determined expansion points.

## 4. Numerical Results

Our test model is a simplified, abstract mechanical structure of a machine tool modeled using the FEM environment MSC.PATRAN/MSC.NASTRAN© (see Figure 1, here TCP denotes the tool center point). The test model is of order $n = 51.816$, it has four inputs ($m = 4$) and eight outputs ($q = 8$). As the proposed algorithms make use of a special Krylov subspace, the damping matrix $D$ had to be chosen either to be zero or as Rayleigh damping

$$D = \alpha \cdot M + \beta \cdot K,$$

i.e. $D$ is proportional to the mass matrix $M$ and the stiffness matrix $K$. The parameters for the proportional damping matrix were chosen as $\alpha = 0.02$ and $\beta = \alpha/1500$.

The algorithms were implemented in MATLAB[1] version 7.7.0 (R2008b) and the computations were performed on a AMD Athlon(tm) 64 X2 Dual Core Processor 4400+ and 2 GB RAM. The second-order system was reduced by the following methods:

1. Algorithm MIRA for systems without damping matrix which generates a Galerkin projection (6) from $P$ and $Q$ as in (11) by the block Arnoldi method (RA).
2. Algorithm IRGA for systems without damping matrix which generates a projection (16) from $P$ and $Q$ as in (11) by the global Arnoldi method (GA).

[1] MATLAB is a trademark of The MathWorks, Inc.

**Algorithm 4** Adaptive Iterative Rational Global Arnoldi for second order systems (AIRGA_2o)

**Input:** matrices $M, D, K, F, C_p, C_v$; number of columns of matrix $F$ $m$; initial expansion points $s_i$, $i = 1, \ldots, \hat{i}$; maximal reduced dimension $r$; tolerance $tol$; $\epsilon$

**Output:** reduced matrices $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$; number of expansions $r_i$, $i = 1, \ldots, \hat{i}$; sequence of used expansion points $\sigma_j$, $j = 1, \ldots, \lceil r/m \rceil$

function $[\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v, r_i, \sigma_j] = \text{AIRGA\_2o}(M, D, K, F, C_p, C_v, m, s_i, r, tol, \epsilon)$
set $s_i^{old}$, $i = 1, \ldots, \hat{i}$ so that $\max_{i \in \{1,2,\ldots \hat{i}\}} |s_i^{old} - s_i| > tol$
$J = \lceil r/m \rceil$
**while** $\max_{i \in \{1,2,\ldots \hat{i}\}} |s_i^{old} - s_i| > tol$ **do**
  $V = [\,]$
  **for** $i = 1 : \hat{i}$ **do**
    $R^{(-1)}(s_i) = 0, \quad h_\pi^{(-1)}(s_i) = 0$
    $R^{(0)}(s_i) = (s_i^2 M + s_i D + K)^{-1} F, \quad h_\pi^{(0)}(s_i) = 1$
  **end for**
  **for** $j = 1 : J$ **do**
    $\sigma_j = \max_{s_i} \| h_\pi^{(j-2)}(s_i)\, C_v\, R^{(j-2)}(s_i) + h_\pi^{(j-1)}(s_i)\, (C_p + s_i C_v)\, R^{(j-1)}(s_i) \|_F$ ▷ Expansion point selection
    $h_{j,j-1}(\sigma_j) = \|R^{(j-1)}(\sigma_j)\|_F$
    $V_j = R^{(j-1)}(\sigma_j)/h_{j,j-1}(\sigma_j)$ ▷ New block of $m$ $F$-orthonormal vectors
    **for** $i = 1 : \hat{i}$ **do**
      **if** $(s_i == \sigma_j)$ **then**
        $Z = -(s_i^2 M + s_i D + K)^{-1} M V_j, \quad h_\pi^{(j)}(s_i) = h_\pi^{(j-1)}(s_i) \cdot h_{j,j-1}(\sigma_i)$ ▷ New block for the next iteration
        compute the rank revealing $QR$ factorization $WR = Z, \quad rank(Z) = s$
        $R^{(j)}(s_i) = W(:, 1 : s)$
      **else**
        $R^{(j)}(s_i) = R^{(j-1)}(s_i), \quad h_\pi^{(j)}(s_i) = h_\pi^{(j-1)}(s_i)$
      **end if**
      **for** $t = 1 : j$ **do**
        $\epsilon_{t,j}(s_i) = \text{trace}(\, V_t\, R^{(j)}(s_i)\,), \quad \delta_{t,j}(s_i) = \text{trace}(\, V_t\, R^{(j-1)}(s_i)\,)$ ▷ $F$-orthogonalization
        $R^{(j)}(s_i) = R^{(j)}(s_i) - \epsilon_{t,j}(s_i)\, V_t, \quad R^{(j-1)}(s_i) = R^{(j-1)}(s_i) - \delta_{t,j}(s_i)\, V_t$
      **end for**
    **end for**
  **end for**
  $\check{V} = [\, V_1\, V_2 \cdots V_J\,]$
  compute the rank revealing $QR$ factorization $\tilde{V}R = \check{V}, \quad rank(\check{V}) = s$
  $V = \tilde{V}(:, 1 : \min\{r, s\})$
  compute $\hat{M} = V^T M V$, $\hat{D} = V^T D V$ and $\hat{K} = V^T K V$ as in (6) ▷ Reduce matrices M, D and K
  compute the eigenvalues $\lambda_j$, $j = 1 \ldots, r$ of the reduced system,
  ordered such that $|\text{Im}(\lambda_1)| \le |\text{Im}(\lambda_2)| \le \ldots \le |\text{Im}(\lambda_r)|$ ▷ Determine a new set of expansion points
  $s_i^{old} \leftarrow s_i$, for $i = 1, \ldots, \hat{i}$
  choose new expansion points $s_i$ as explained at the end of Section 2.1
**end while**
compute $\hat{M}, \hat{D}, \hat{K}, \hat{F}, \hat{C}_p, \hat{C}_v$ as in (6) ▷ Yield reduced system matrices by congruence transformation

3. Algorithm AIRGA_2o for systems without damping matrix which generates a projection (16) from $P$ and $Q$ as in (11) by the adaptive global Arnoldi method (AGA).
4. Algorithm MIRA for systems with proportional damping which generates a Galerkin projection (6) from $P$ and $Q$ as in (11) by the block Arnoldi method (RA_PD).
5. Algorithm IRGA for systems with proportional damping which generates a projection (16) from $P$ and $Q$ as in (11) by the global Arnoldi method (GA_PD).
6. Algorithm AIRGA_2o for systems with proportional damping which generates a projection (16) from $P$ and $Q$

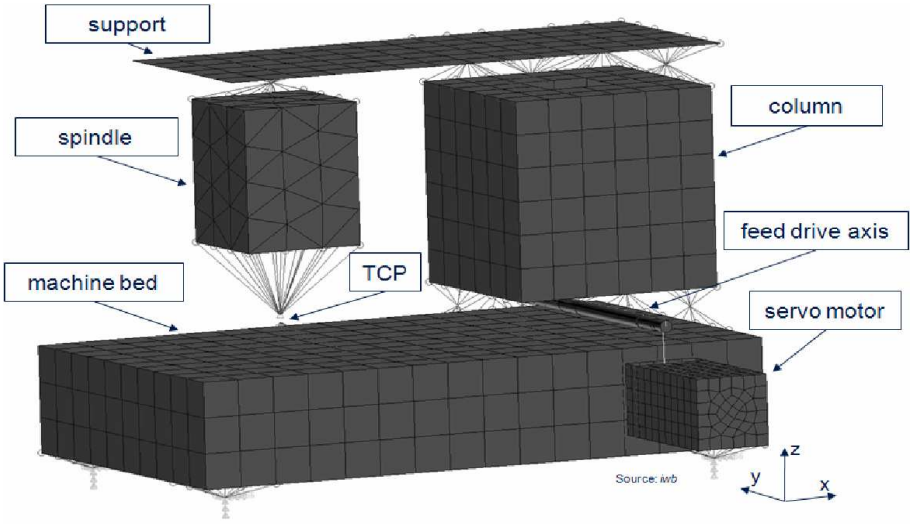


Figure 1: FE model of a simplified, abstract mechanical structure.

as in (11) by the adaptive global Arnoldi method (AGA_PD).

The first three methods are modified versions of the Rational-Arnoldi method for first-order systems. They reduce second-order systems without damping matrix [9, 14]. That is, they assume $D = 0$ in (11) and compute a reduced system (2) with $\tilde{D} = 0$. A damped reduced system is obtained by adding the proportional damping matrix $\hat{D} = \alpha\hat{M} + \beta\hat{K}$. The last three methods exploit the special structure of the proportional damping matrix [4]. In case complex valued expansion points are used, the last three of the above algorithms generate complex valued matrices $V$. The methods RA, GA and AGA generate real matrices even in case of complex valued expansion points.

Once a complex expansion point is used, in the algorithms to reduce systems with proportional damping matrix, all further computations involve complex arithmetic. As the reduced systems are commonly used for further simulation in NASTRAN or SIMULINK which requires real systems, the following considerations had to be taken into account. Before computing the congruence transformation the transformation matrix $V \in \mathbb{C}^{n\times r}$ has to be transformed back to a real matrix. This can be done, e.g., as follows

$$[V, R] = \mathrm{qr}(\,[\mathrm{Re}(V(:, 1 : \lceil r/2 \rceil))\ \mathrm{Im}(V(:, 1 : \lfloor r/2 \rfloor))]\,). \tag{17}$$

In our implementation a rank-revealing $QR$ decomposition was used to compute (17). By this process the number of columns of $V$ doubles. Therefore, in setting up the real transformation matrices only the first $r/2$ columns of $V$ were used, so that the resulting system is of the desired order $r$. Note that this process halves the number of matched moments.

The methods were started with the four initial expansion points $2\pi\imath$, $500\pi\imath$, $1000\pi\imath$ and $1500\pi\imath$. To match at least the first two moments (methods RA, GA and AGA) resp. at least the first moment (methods RA_PD, GA_PD and AGA_PD) of the original systems in our computations, the reduction methods where used with $k_i = 2$, $i = 1, \ldots, \hat{\imath}$, $\hat{\imath} = 4$, $tol = 0.1$ and $\epsilon = 750$. With these parameters the reduced dimension was $r = k_i \cdot \hat{\imath} \cdot m = 2 \cdot 4 \cdot 4 = 32$. With the procedure to choose the expansion points $s_i$ as explained at the end of section 2.1 the last expansion point is located at frequencies higher than $\hat{\imath} \cdot \epsilon/(2\pi) = 358$ Hz. Hence, we expect a good approximation of the frequency range from 0 Hz to 358 Hz at least. Besides the methods already mentioned, a second-order modal reduced system of dimension 32 (MODAL_2o) was generated by NASTRAN in order to compare the approximation results of the various reduction methods. In Table 2 essential information about the results obtained with the various methods is summarized. The sequence of expansion points adaptively determined in the last iteration by the method AGA was $s_1, s_2, s_1, s_3, s_4, s_2, s_3, s_4$ resp. $s_1, s_2, s_3, s_4, s_1, s_2, s_3, s_4$ by the method AGA_PD. The numerical results demonstrate the applicability of the proposed methods based on the global Arnoldi method. The approximation of the transfer function and the time response by reduced systems obtained with global Arnoldi methods are comparable to those

obtained with block Arnoldi reduced systems. All Krylov based reduced systems approximate the original system transfer function more accurate than the modal reduced system of the same order.

Table 2: Information about the results obtained with the various Krylov subspace methods

| methods for systems without damping matrix | | | | |
|---|---|---|---|---|
| | $V, W \in$ | expansion points $s_i$ in the last iteration | time to reduce the system [s] | number of iterations |
| RA | $\mathbb{R}^{32\times32}$ | $2\pi(74.28\imath, 250.05\imath, 435.59\imath, 588, 76\imath)$ | 182,55 | 2 |
| GA | $\mathbb{R}^{32\times32}$ | $2\pi(74.28\imath, 250.05\imath, 435.59\imath, 588, 76\imath)$ | 166,17 | 2 |
| AGA | $\mathbb{R}^{32\times32}$ | $2\pi(74.28\imath, 250.05\imath, 435.59\imath, 588, 76\imath)$ | 180,62 | 2 |

| methods for systems with proportional damping | | | | |
|---|---|---|---|---|
| | $V, W \in$ | expansion points $s_i$ in the last iteration | time to reduce the system [s] | number of iterations |
| RA_PD | $\mathbb{C}^{32\times32}$ | $2\pi(74.28\imath, 250.05\imath, 435.59\imath, 588, 76\imath)$ | 426,62 | 2 |
| GA_PD | $\mathbb{C}^{32\times32}$ | $2\pi(74.28\imath, 250.05\imath, 435.59\imath, 588, 76\imath)$ | 370,64 | 2 |
| AGA_PD | $\mathbb{C}^{32\times32}$ | $2\pi(74.28\imath, 250.05\imath, 435.59\imath, 588, 76\imath)$ | 445,84 | 2 |

In order to compare the different methods the approximation of the original transfer function and the time response of the reduced systems were analyzed. To assess the quality of the reduced systems the following errors were used:

- The relative error of the transfer function at frequency $f$ from the $k$-th input to the $l$-th output of a reduced system was computed by
$$\epsilon_{rel}(f) = \frac{|H_{k,l}(f) - \hat{H}_{k,l}(f)|}{|H_{k,l}(f)|}.$$
Here $\epsilon_{rel}(f)$ is the relative error, $H_{k,l}$ and $\hat{H}_{k,l}$ are the transfer functions from the $k$-th input to the $l$-th output of the original resp. of the reduced system.

- The absolute time response error at time $t$ from the $k$-th input to the $l$-th output of a reduced system was computed by
$$\epsilon_{abs}(t) = |y_{k,l}(t) - \hat{y}_{k,l}(t)|.$$
Here $\epsilon_{abs}(t)$ is the absolute error, $y_{k,l}$ and $\hat{y}_{k,l}$ are the time responses from the $k$-th input to the $l$-th output of the original resp. of the reduced system.

### *4.1. Approximation of the transfer function*

In Figure 2 the transfer function and the time response of the TCP's relative motion (5'th output) against the motor torque (1'st input) and in Figure 3 the relative approximation errors $\epsilon_{rel}(f)$ of the reduced systems are displayed. The relevant frequency interval is from 0 to 750 Hz because this frequency range is most important to simulate the behavior of mechanical structures. The results of the proposed methods for the system reduced without damping matrix $D$ are displayed on the left in each of the figures, while the results for the system reduced with proportional damping matrix are given on the right hand side. Besides the relative error for the different Krylov subspace reduced models all figures also include the relative error for the second-order modal reduced system (MODAL_2o) of dimension 32.

Clearly, all Krylov subspace reduced systems approximate the original system more accurate than the modal reduced system in the frequency interval considered here. All methods achieve reduced systems with lower approximation errors than the modal reduced system of the same order. As expected all reduced systems approximate the original transfer function up to a frequency of 358 Hz and higher very accurately.

All Krylov subspace reduced systems obtained by reduction of a system without damping matrix have an error smaller than $5 \cdot 10^{-9}$ for frequencies up to 600 Hz. For higher frequencies the error increases. The maximum errors of the global Arnoldi methods GA and AGA ($10^{-5}$) are slightly higher than of the block Arnoldi method RA ($10^{-6}$).

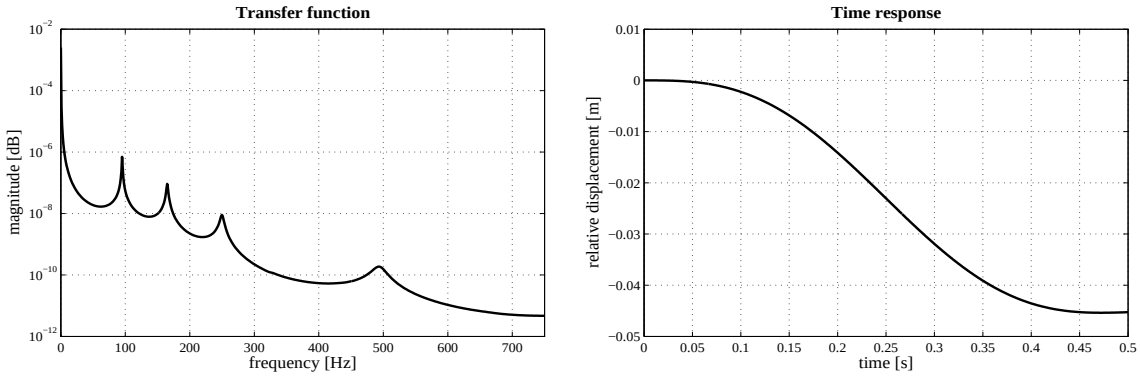


Figure 2: Transfer function and time response from the 1'st input to the 5'th output of the original system.

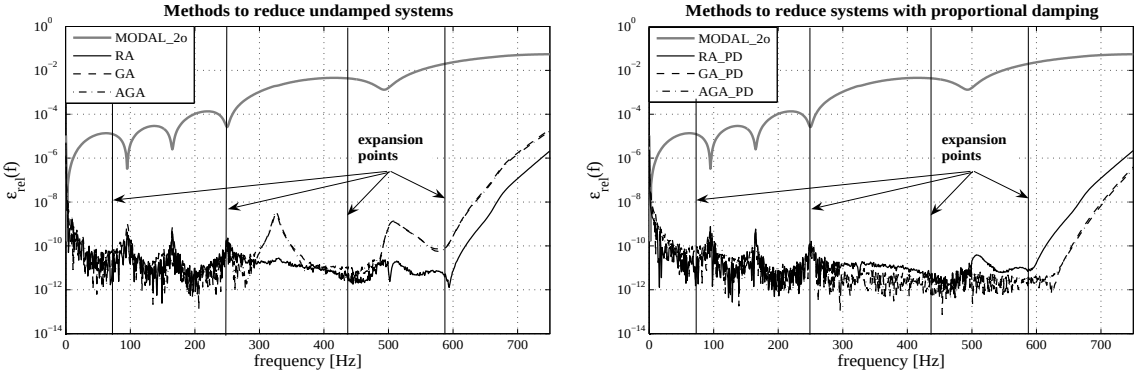


Figure 3: Relative error $\epsilon_{rel}(f)$.

All Krylov subspace reduced systems obtained by reduction of a system with proportional damping matrix have a maximal error of $10^{-9}$ for frequencies up to 600 Hz. Here the maximum errors of the block Arnoldi method RA_PD ($10^{-6}$) are slightly higher than that of the global Arnoldi methods GA and AGA ($5 \cdot 10^{-7}$). In the whole frequency range considered here the approximation errors of this systems are very similar or smaller than those of systems achieved by the methods without damping matrix.

### *4.2. Approximation of the time response*

To analyze the approximation abilities of the response behavior in time domain an input signal $u(t) = 1000 \cdot sin(4\pi t)$ was used as torque on the motor shaft. In Figure 4 the absolute approximation errors $\epsilon_{abs}(t)$ from the 1'st input to the 5'th output of the reduced systems are displayed. The relevant time interval is from 0 to 0,5 sec. In the entire time interval considered here all reduced systems approximate the time response of the original system very accurately with maximal absolute error smaller than $2 \cdot 10^{-4}$. The errors of all Krylov subspace reduced systems and the error of the modal reduced system are nearly the same.

## 5. Conclusions

In this paper we propose a novel multi point adaptive method combining the Adaptive Order Rational Global Arnoldi (AORGA) algorithm and a modified Iterative Rational Krylov Algorithm (IRKA) for application in model order reduction of second-order systems. Undamped systems as well as systems with proportional damping can be treated. Starting from a arbitrary initial set of expansion points the method determines the expansion points iteratively and the expansions per expansion point adaptively. Numerical experiments show good approximation results of the time response and the transfer function, especially for low frequencies. This frequency range is most important to

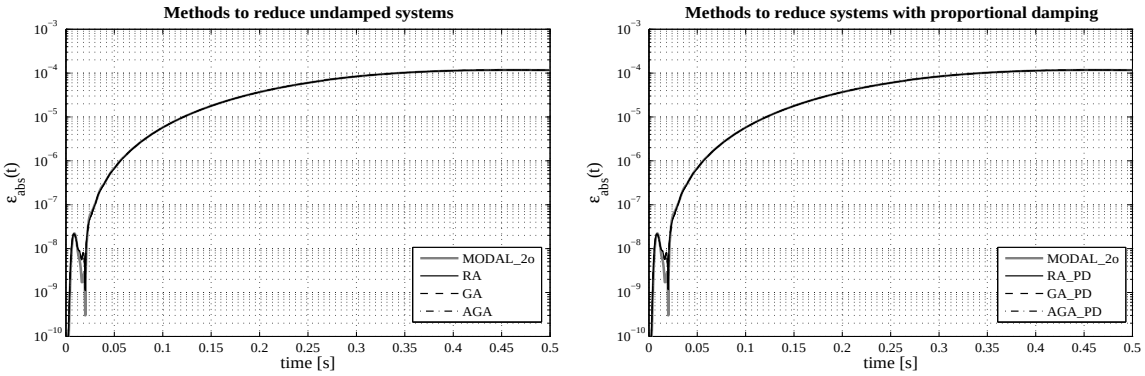


Figure 4: Absolute error $\epsilon_{abs}(t)$ of the reduced systems.

simulate the behavior of mechanical structures. Further investigations by adoption this method to the global Lanczos algorithm and for Krylov subspaces of second kind are still in work.

## Acknowledgment

The authors would like to thank the German science foundation (Deutsche Forschungsgemeinschaft (DFG)) for funding of the project "WAZorFEM" in 2007 and 2008 which was the starting point for this work. The authors thank Peter Benner (TU Chemnitz) for many enlightening discussions.

## References

[1] A. Antoulas, Approximation of Large-Scale Dynamical Systems, SIAM, Philadelphia, USA, 2005.
[2] Z. Bai, Y. Su, SOAR: A Second-order Arnoldi Method for the Solution of the Quadratic Eigenvalue Problem, SIAM Journal on Matrix Analysis and Applications 26 (3) (2005) 640–659.
[3] M. Bampton, R. Craig, Coupling of Substructures for Dynamic Analysis, AIAA Journal 6 (1968) 1313–1319.
[4] C. Beattie, S. Gugercin, Krylov-based model reduction of second-order systems with proportional damping, in: Proceedings of the 44th IEEE Conference on Decision and Control, 2005, pp. 2278–2283.
[5] C.-C. Chu, M. Lai, W. Feng, MIMO Interconnects Order Reductions by Using the Multiple Point Adaptive-Order Rational Global Arnoldi Algorithm, IEICE Trans. Electron. E89-C(6) (2006) 792–802.
[6] C.-C. Chu, M. Lai, W. Feng, The multiple point global Lanczos method for multiple-inputs multiple-outputs interconnect order reduction, IEICE Trans. Fundam. E89-A(10) (2006) 2706–2716.
[7] C.-C. Chu, M. Lai, W. Feng, Lyapunov-based error estimations of MIMO interconnect reductions by using the global Arnoldi algorithm, IEICE Trans. Electron. E90-A(2) (2007) 415–418.
[8] C.-C. Chu, M. Lai, W. Feng, Model-order reductions for MIMO systems using global Krylov subspace methods, Mathematics and Computers in Simulation 79 (2008) 1153–1164.
[9] R. Eid, B. Salimbahrami, B. Lohmann, E. Rudnyi, J. Korvink, Parametric Order Reduction of Proportionally Damped Second Order Systems, Technical Report on Automatic Control TRAC-1, Technische Universität München, Lehrstuhl für Regelungstechnik, 2006.
[10] P. Feldmann, R. W. Freund, Efficient linear circuit analysis by Padé approximation via the Lanczos process, IEEE Trans. Computer-Aided Design 14 (1995) 639–649.
[11] M. Frangos, I. M. Jaimoukha, Adaptive rational interpolation: Arnoldi and Lanczos-like equations, European Journal of Control (2008) 342–354.
[12] K. Gallivan, E. Grimme, P. van Dooren, A rational Lanczos algorithm for model reduction, Numerical Algorithms 12 (1996) 33–63.
[13] G.H.Golub, C. van Loan, Matrix Computations, John Hopkins University Press, 1996.
[14] E. Grimme, Krylov projection methods for model reduction, Ph.D. thesis, University of Illinois at Urbana-Champaign (1997).
[15] E. Grimme, K. Gallivan, A rational lanczos algorithm for model reduction ii: Interpolation point selection, Numerical Algorithms 12 (1998) 33–63.
[16] S. Gugercin, A. Antoulas, An $\mathcal{H}_2$ error expression for the Lanczos procedure, in: Proceedings of the 2003 IEEE Conference on Decision and Control, 2003, pp. 1869–1872.
[17] S. Gugercin, A. Antoulas, C. Beattie, $\mathcal{H}_2$-Model Reduction for Large-Scale Linear Systems, SIAM Journal on Matrix Analysis and Applications 30 (2) (2008) 609–638.
[18] M. Heyouni, The global Hessenberg and CMRH methods for linear systems with multiple right-hand sides, Numer. Algorithms 26 (4) (2001) 317–332.

[19] N. Higham, D. Mackey, N. Mackey, F. Tisseur, Symmetric linearizations for matrix polynomials, SIAM J. Matrix Anal. Appl. 29 (1) (2006) 143–159.
[20] R. Horn, C. Johnson, Topics in Matrix Analysis, Cambridge University Press, 1991.
[21] K. Jbilou, A. Messaoudi, H. Sadok, Global FOM and GMRES algorithms for matrix equations, Appl. Numer. Math. 31 (1999) 49–63.
[22] K. Jbilou, A. Riquet, Projection methods for large Lyapunov matrix equations, Linear Algebra and its Applications 415 (2-3) (2006) 344–358.
[23] M.-H. Lai, C.-C. Chu, W.-S. Feng, Applications of AOGL Model-Order Reduction Techniques in Interconnect Analysis, in: Proceedings of the 2007 IEEE International Symposium on Circuits and Systems, 2007, pp. 1133–1136.
[24] H.-J. Lee, C.-C. Chu, W. Feng, An adaptive-order rational Arnoldi method for model-order reductions of linear time-invariant systems, Linear Algebra Appl. 415 (2006) 235–261.
[25] D. Mackey, N. Mackey, C. Mehl, V. Mehrmann, Vector spaces of linearizations for matrix polynomials, SIAM J. Matrix Anal. Appl. 28 (4) (2006) 971–1004.
[26] M. Odabasioglu, M. Celik, T. Pileggi, Prima: passive reduced-order interconnect macromodeling algorithm, IEEE Trans. on CAD of Integrated Circuits and Systems 17 (8) (1998) 645–654.
[27] M. Sadkane, Block Arnoldi and Davidson methods for unsymmetric large eigenvalue problems, Numer. Math. 64 (1993) 195–211.
[28] S. Salimbahrami, Structure Preserving Order Reduction of Large Scale Second Order Models, Ph.D. thesis, Technical University of Munich, Germany (2005).
[29] G. Stewart, Matrix Algorithms, Volume II: Eigensystems, SIAM, Philadelphia, USA, 2001.
[30] Y. Su, J. Wang, X. Zeng, Z. Bai, C. Chiang, D. Zhou, SAPOR: Second-order Arnoldi method for passive order reduction of RCS circuits, in: Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design, 2004, pp. 74–79.
[31] F. Tisseur, K. Meerbergen, A survey of the quadratic eigenvalue problem, SIAM Review 43 (2001) 234–286.