Max Planck Institute Magdeburg
Preprints

Peter Benner André Schneider

Balanced Truncation for Descriptor Systems with Many Terminals

Preprint submitted to Systems & Control Letters

MPIMD/13-17 **October 7, 2013**

Abstract

The model reduction method introduced in [Benner, P. and Schneider, A.; *Balanced Truncation Model Order Reduction for LTI Systems with many Inputs or Outputs*, in A. Edelmayer: Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems, 2010, ISBN/ISSN: 978-963-311-370-7] shows how to reduce linear time-invariant (LTI) continuous-time state space systems with either many inputs or many outputs using the well-known balanced truncation approach. We call this method balanced truncation for many terminals (BTMT). In this work we generalize BTMT to descriptor systems of the form

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), & A, E \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, \\ y(t) &= Cx(t) + Du(t), & C \in \mathbb{R}^{p\times n}, D \in \mathbb{R}^{p\times m}, \end{aligned}$$

where $m \in \mathcal{O}(n)$ and $p \ll n$, or vice versa. We show how to obtain a reduced order model by solving one Lyapunov equation and using the Gauss-Kronrod quadrature to compute the needed projection matrices. In particular, we discuss the case when E is singular and show numerical results.

Impressum:

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg

Publisher:
Max Planck Institute for
Dynamics of Complex Technical Systems

Address:
Max Planck Institute for
Dynamics of Complex Technical Systems
Sandtorstr. 1
39106 Magdeburg

`www.mpi-magdeburg.mpg.de/preprints`

1 Introduction

Throughout this work, we consider linear time-invariant continuous-time descriptor systems of the form

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \quad x(0) = x_0, \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{1}$$

Here, $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, $D \in \mathbb{R}^{p\times m}$, $x(t) \in \mathbb{R}^n$ contains internal state variables, $u(t) \in \mathbb{R}^m$ is the vector of input variables, $y(t) \in \mathbb{R}^p$ is the output vector, $x_0 \in \mathbb{R}^n$ is the initial value and n is the number of state variables, called the *order* of the system. W.l.o.g., we assume the so-called feed-through term $D \equiv 0$, as we can always re-write (1) as a system with nonzero feed-through term so that $D = 0$, see [1].

In model order reduction (MOR), the number of inputs m and the number of outputs p of descriptor systems are mostly assumed to be much smaller than the order of the system n itself [2]. Numerical problems in solving large-scale matrix equations or in building up Krylov subspaces efficiently are the reasons [3]. Especially if further calculations require the knowledge of the full state while the control of the system is bounded to a few inputs leads to a system structure with a number of outputs similar or equal to the number of states. An application is the thermal driven deformation of a workpiece which heats-up during the manufacturing process [4, 5]. To calculate this deformation, the solution of the heat equation in every point of the discretized domain is required. In contrast to these many outputs, a heat input is given only in sparse points of the domain, e.g., at the boundary or the tool center point of the workpiece. Another area of interest is flow simulation [6, 7]. Dealing with such problems, it is often of significance to know the fluid velocity or the pressure in every point of the domain. Mathematically, this means nothing else than the output matrix C of the system (1) is equal to the identity I_n. The numerical algorithms introduced here also perform if the system is subjected to distributed control, maybe even controlled everywhere, i.e. $B = I_n$, while C is of low rank. For example, such systems arise also in fluid flow control [8]. Domain decomposition approaches, e.g., in electromagnetic simulation [9] or flow problems [6, 7], may also lead to system structures described above.

The aim of this paper is to introduce a numerical methodology how to reduce many terminal descriptor systems by applying balanced truncation (BT) [10]. In Section 2 we review the basics of BT for descriptor systems [11]. We show which Lyapunov equations (LE) need to be solved and where the problems with high rank input or output matrices arise. In Section 3 we show how to overcome these problems by solving only the "cheap" LE in low rank factor form. This information is combined with an integral representation of the systems Gramians and the Gauss-Kronrod quadrature formula. Applying a Schur decomposition approach to the computed matrix product, the needed projection matrices are computed. In Section 4 we introduce two test problems. One model deals with the heat transfer within a hollow cylinder, which is simulated via a state space system, i.e., $E = I_n$. The second system is a mechanical damped mass-spring system, see [11]. This differential algebraic equation (DAE) has additional

constraints and is of index 3. We show difficulties and results of the reduction. We conclude in Section 5.

2 Balanced Truncation

A detailed knowledge of the BT MOR approach for descriptor systems is important to understand this work. Therefore, we repeat in detail basic knowledge based on previous work, e.g., [12, 13, 10, 11]. We assume the matrix pencil $\lambda E - A$ to be regular, i.e., $\det(\lambda E - A) \neq 0$ for some $\lambda \in \mathbb{C}$. Additional to regularity, we assume the pencil to be asymptotically stable, i.e., its finite eigenvalues have negative real parts. There exists a transformation of the pencil to *Weierstrass canonical form* [14] with

$$E = W \begin{bmatrix} I_{n_f} & 0 \\ 0 & N \end{bmatrix} T \quad \text{and} \quad A = W \begin{bmatrix} J & 0 \\ 0 & I_{n_\infty} \end{bmatrix} T. \tag{2}$$

The matrices J and N are of Jordan canonical form, N is nilpotent with nilpotency index ν, and the numbers n_f and n_∞ are the dimensions of the deflating subspaces of the pencil corresponding to the finite and infinite eigenvalues of the system. The spectral projections onto the right and the left deflating subspaces of the pencil corresponding to the finite eigenvalues have the form

$$P_r = T^{-1} \begin{bmatrix} I_{n_f} & 0 \\ 0 & 0 \end{bmatrix} T \quad \text{and} \quad P_l = W \begin{bmatrix} I_{n_f} & 0 \\ 0 & 0 \end{bmatrix} W^{-1}.$$

By means of this transformation it is possible to partition the system matrices following the block structure of (2) such that a coordinate transformation enables decoupling of the system in the so-called slow subsystem and the fast subsystem which contains the information about the dynamics. More details including solution theory can be found in [15, 11]. We apply the Laplace transform $\mathcal{L} : f(t) \longrightarrow \int_0^\infty e^{-st} f(t) dt$ to (1). With respect to $\mathcal{L}(\dot{x}(t)) = sx(s) - x(0)$, we get

$$y(s) = C(sE - A)^{-1} B u(s) + C(sE - A)^{-1} E x(0),$$

where $s \in \mathbb{C}$ and $x(s)$, $u(s)$, and $y(s)$ are the Laplace transforms of $x(t)$, $u(t)$, and $y(t)$, respectively. If $Ex(0) = 0$, the input to output rational matrix valued function

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

is called transfer function of (1). It is possible to derive a general resolvent of the pencil by means of the Weierstrass canonical form and a Laurent expansion at infinity

$$(\lambda E - A)^{-1} = \sum_{k=-\infty}^{\infty} F_k \lambda^{-k-1},$$

with the coefficients $F_k = T^{-1} \begin{bmatrix} J^k & 0 \\ 0 & 0 \end{bmatrix} W^{-1}$ for $k \in \mathbb{N}_0$ and

$F_k = T^{-1} \begin{bmatrix} 0 & 0 \\ 0 & -N^{-(k+1)} \end{bmatrix} W^{-1}$ for $k < 0$. Using this, the transfer function can be

expressed as

$$G(s) = \sum_{k=-\infty}^{\infty} CF_k Bs^{-(k+1)}. \tag{3}$$

Since N is nilpotent with index ν, i.e., $N^{\nu-1} \neq 0, N^{\nu} = 0$, it follows that $F_k = 0$, $\forall k < -\nu$. The number ν is called the (algebraic) index of the DAE. Consequently, the transfer function can be split into the strictly proper part

$$G_{sp}(s) = \sum_{k=0}^{\infty} CF_k Bs^{-(k+1)}$$

and the polynomial part

$$P(s) = \sum_{k=1}^{\nu} CF_{-k} Bs^{(k-1)}.$$

Furthermore, the proper observability Gramian $\mathcal{G}_{po}$, the proper controllability Gramian $\mathcal{G}_{pc}$, the improper observability Gramian $\mathcal{G}_{io}$, and the improper controllability Gramian $\mathcal{G}_{ic}$ are defined as

$$\begin{aligned}
\mathcal{G}_{po} &= \int_0^{\infty} \mathcal{F}^T(t) C^T C \mathcal{F}(t) dt, \\
\mathcal{G}_{pc} &= \int_0^{\infty} \mathcal{F}(t) B B^T \mathcal{F}^T(t) dt, \\
\mathcal{G}_{io} &= \sum_{k=-\nu}^{-1} F_k^T C^T C F_k, \quad \text{and} \\
\mathcal{G}_{ic} &= \sum_{k=-\nu}^{-1} F_k B B^T F_k^T,
\end{aligned}$$

where $\mathcal{F} = T^{-1} \begin{bmatrix} e^{tJ} & 0 \\ 0 & 0 \end{bmatrix} W^{-1}$ is the fundamental solution matrix of system (1).

As an example, we discuss the square root variant of balanced truncation. The way to reduce the system is to transform it into a so-called balanced form, which means that after a transformation

$$\mathcal{T} : [E, A, B, C] \to \left[\tilde{W} E \tilde{T}, \tilde{W} A \tilde{T}, \tilde{W} B, C \tilde{T}\right], \tag{4}$$

the systems proper Gramians as well as the improper Gramians satisfy

$$\begin{aligned}
\tilde{G}_{po} (:= E^T \mathcal{G}_{po} E) = \mathcal{G}_{pc} &= \begin{bmatrix} \Sigma_p & 0 \\ 0 & 0 \end{bmatrix} \qquad \text{and} \\
\tilde{G}_{io} (:= A^T \mathcal{G}_{io} A) = \mathcal{G}_{ic} &= \begin{bmatrix} 0 & 0 \\ 0 & \Sigma_i \end{bmatrix},
\end{aligned}$$

such that $\Sigma_p = \operatorname{diag}(\theta_1^p, \ldots, \theta_{n_f}^p)$ and $\Sigma_i = \operatorname{diag}(\theta_1^i, \ldots, \theta_{n_\infty}^i)$ consist of the so-called proper and improper Hankel singular values (HSVs). These HSVs can be interpreted as

a measure of how difficult to observe or how hard to reach a generalized state is. Similar to the truncation of the singular values performing a best low rank approximation, the proper Hankel singular values can also be truncated by means of the SVD. The improper ones need to be retained due to stability reasons [11]. The matrices $\tilde{W}$ and $\tilde{T}$ are not unique for descriptor systems. For example, in [11], these matrices are given as

$$\tilde{W} = \left[R_p U_p \Sigma_p^{-\frac{1}{2}}, R_i U_i \Sigma_i^{-\frac{1}{2}}\right], \quad \tilde{T} = \left[S_p V_p \Sigma_p^{-\frac{1}{2}}, S_i V_i \Sigma_i^{-\frac{1}{2}}\right], \tag{5}$$

with $\Sigma_p^{-1/2} = \mathrm{diag}(\frac{1}{\sqrt{\theta_1^p}}, ..., \frac{1}{\sqrt{\theta_{n_f}^p}})$, $\Sigma_i^{-1/2} = \mathrm{diag}(\frac{1}{\sqrt{\theta_1^i}}, ..., \frac{1}{\sqrt{\theta_{n_\infty}^i}})$. Here, θ^p and θ^i are the proper and improper HSVs respectively, and

$$\mathcal{G}_{pc} \approx S_p S_p^T, \qquad \mathcal{G}_{po} \approx R_p R_p^T, \tag{6a}$$

$$\mathcal{G}_{ic} \approx S_i S_i^T, \qquad \mathcal{G}_{io} \approx R_i R_i^T. \tag{6b}$$

Moreover, $R_p^T E S_p = U_p \Sigma_p V_p^T$ as well as $R_i^T A S_i = U_i \Sigma_i V_i^T$, such that the transformed realization (4) represents the system in balanced form. Please notice that, e.g., $S_p S_p^T$ with $S_p \in \mathbb{R}^{n \times n_{pc}}$ is a rank n_{pc} approximation to $\mathcal{G}_{pc}$. Equivalently, $R_p \in \mathbb{R}^{n \times n_{po}}$, $S_i \in \mathbb{R}^{n \times n_{ic}}$, and $R_i \in \mathbb{R}^{n \times n_{io}}$ with $n_{pc}, n_{po}, n_{ic}, n_{io} \ll n$.

Based on [16] and [11], the Gramians can alternatively be defined as

$$\begin{aligned}
\mathcal{G}_{po} &= \frac{1}{2\pi}\int_{-\infty}^{\infty} (-i\omega E - A)^{-T} P_r^T C^T \times \\
&\quad C P_r (i\omega E - A)^{-1} d\omega, && \text{(7a)}\\
\mathcal{G}_{pc} &= \frac{1}{2\pi}\int_{-\infty}^{\infty} (i\omega E - A)^{-1} P_l B \times \\
&\quad B^T P_l^T (-i\omega E - A)^{-T} d\omega, && \text{(7b)}\\
\mathcal{G}_{io} &= \frac{1}{2\pi}\int_{0}^{2\pi} (e^{-i\omega} E - A)^{-T} (I - P_r)^T C^T \times \\
&\quad C (I - P_r)(e^{i\omega} E - A)^{-1} d\omega, \quad \text{and} && \text{(7c)}\\
\mathcal{G}_{ic} &= \frac{1}{2\pi}\int_{0}^{2\pi} (e^{i\omega} E - A)^{-1} (I - P_l) B \times \\
&\quad B^T (I - P_l)^T (e^{-i\omega} E - A)^{-T} d\omega. && \text{(7d)}
\end{aligned}$$

For a numerical computation of the HSVs it is essential to know that $\mathcal{G}_{pc}$ and $\mathcal{G}_{po}$ are the unique, symmetric, positive semidefinite solutions of the projected generalized continuous-time algebraic Lyapunov equations

$$\begin{aligned}
E^T \mathcal{G}_{po} A + A^T \mathcal{G}_{po} E &= -P_r^T C^T C P_r, && \text{(8a)}\\
\mathcal{G}_{po} &= P_l^T \mathcal{G}_{po} P_l, \\
E \mathcal{G}_{pc} A^T + A \mathcal{G}_{pc} E^T &= -P_l B B^T P_l^T, && \text{(8b)}\\
\mathcal{G}_{pc} &= P_r \mathcal{G}_{pc} P_r^T,
\end{aligned}$$

see [17]. The improper Gramians $\mathcal{G}_{io}$ and $\mathcal{G}_{ic}$ are the unique, symmetric, positive semidefinite solutions of the projected generalized discrete-time algebraic Lyapunov equations

$$\begin{aligned} A^T\mathcal{G}_{io}A - E^T\mathcal{G}_{io}E &= (I-P_r)^TC^TC(I-P_r), && \text{(9a)}\\ 0 &= P_l^T\mathcal{G}_{io}P_l, \\ A\mathcal{G}_{ic}A^T - E\mathcal{G}_{ic}E^T &= (I-P_l)BB^T(I-P_l)^T, && \text{(9b)}\\ 0 &= P_r\mathcal{G}_{ic}P_r^T. \end{aligned}$$

If E is nonsingular, then $P_r = P_l = I$. Hence, it is sufficient to solve the dual Lyapunov equations

$$A\mathcal{G}_{pc}E^T + E\mathcal{G}_{pc}A^T + BB^T = 0$$

and

$$A^T\mathcal{G}_{po}E + E^T\mathcal{G}_{po}A + C^TC = 0.$$

If the index of the pencil is given by $\nu = 1$, the algorithms proposed in [18] can be used. Due to symmetry and positive semidefiniteness, we compute the approximate factors R_p, R_i, S_p, and S_i of the Gramians in (6). The low rank solution of LEs without projection is explained, e.g., in [19, 20, 21, 22]. In [23], the Bartels-Stewart algorithm and Hammarling's method generalized to projected LEs are presented. Recently, [24] introduces Krylov subspace methods to solve projected Lyapunov equations numerically and [25] presents solvers for projected generalized Lyapunov equations based on matrix equations subroutines that are available in the *Subroutine Library In COntrol Theory* (SLICOT)[1]. A generalization of the alternating direction implicit method and the Smith method for large-scale projected generalized Lyapunov equations is presented in detail in [26]. The proposed methods work iteratively and can be employed for large-scale problems with right hand sides of small and moderate rank. The numerical experiments presented in this work use these algorithms whenever a projected LE with a suitable right hand side is to be solved. In Section 3 we see that all methods become numerically unfeasible if the right hand sides of the LEs become too large.

After calculating the required low rank factors R_p, R_i, S_p, and S_i, the reduction process continues for the proper and improper parts separately. We calculate the balancing transformations as in (5) and the insignificant information of the proper part is truncated. Consequently, singular value decompositions of the form

$$R_p^TES_p = [U_{p_1}, U_{p_2}]\begin{bmatrix}\Sigma_{p_1} & 0\\ 0 & \Sigma_{p_2}\end{bmatrix}[V_{p_1}, V_{p_2}]^T \qquad (10)$$

with $\Sigma_{p_1} = \operatorname{diag}(\theta_1^p, \ldots, \theta_r^p)$, $\Sigma_{p_2} = \operatorname{diag}(\theta_{r+1}^p, \ldots, \theta_{r_p}^p)$,
$r_p = \operatorname{rank}(R_p^TES_p) \leq \max(n_{po}, n_{pc})$, and

$$R_i^TAS_i = U_i\Sigma_iV_i^T \qquad (11)$$

[1] `http://www.slicot.org/`

Algorithm 2.1 Balanced truncation generalized square root method, see [11, Algorithm 3.3.1.]

Input: Realization A, E, B, C, such that $\lambda E - A$ is regular and stable.
Output: Reduced order realization A_r, E_r, B_r, C_r.

1: Compute S_p and R_p of (6a) as solution of (8).
2: Compute S_i and R_i of (6b) as solution of (9).
3: Perform the SVDs in (10) and (11).
4: Compute truncated balancing transformation (12a) and (12b).
5: Calculate the reduced system by projection as in (13).

with $\Sigma_i = \operatorname{diag}(\theta_1^i, \ldots, \theta_{r_i}^i)$ and $r_i = \operatorname{rank}(R_i^T A S_i) \leq \max(n_{io}, n_{ic})$ are computed. Defining the balancing transformations as

$$\tilde{W}_r = \left[R_p U_{p_1} \Sigma_{p_1}^{-\frac{1}{2}}, R_i U_i \Sigma_i^{-\frac{1}{2}} \right] \quad \text{and} \tag{12a}$$

$$\tilde{T}_r = \left[S_p V_{p_1} \Sigma_{p_1}^{-\frac{1}{2}}, S_i V_i \Sigma_i^{-\frac{1}{2}} \right], \tag{12b}$$

we truncate everything belonging to Σ_{p_2}. The reduced realization of order $r + r_i$ is calculated via

$$[E_r, A_r, B_r, C_r] = \left[\tilde{W}_r^T E \tilde{T}_r, \tilde{W}_r^T A \tilde{T}_r, \tilde{W}_r^T B, C \tilde{T}_r \right]. \tag{13}$$

Summarizing, the square root method of balanced truncation for usual descriptor systems follows Algorithm 2.1, see [11]. Since balancing sometimes leads to ill-conditioned matrices $\tilde{W}_r$ and $\tilde{T}_r$, there, the authors also propose a balancing free method based on [27].

Unfortunately, Step 1 and Step 2 of Algorithm 2.1 become computationally infeasible if either the system has many inputs or outputs as the complexity grows at least by an order of magnitude and the memory requirements become too high. This motivates the new BTMT approach.

3 Balanced Truncation for Many Terminals (BTMT)

The up to now established iterative approaches to solve large-scale Lyapunov equations assume that the rank of the right hand side is much smaller than the size of the large and sparse coefficient matrices E and A. The rank of the Gramian factors and hence the efficiency of the reduction process depend on the rank of the right hand side. In practice, a typical challenge is to observe the dynamical system in every possible generalized state. In this case $C = I_n$, where I_n is the identity matrix of size n. Consequently, $\operatorname{rank}(P_r^T C^T C P_r) \in \mathcal{O}(n)$ and/or $\operatorname{rank}((I - P_r)^T C^T C (I - P_r)) \in \mathcal{O}(n)$. W.l.o.g., we assume these rank properties in this article. Technically, the approach also holds if the assumption is violated regarding the inputs, i.e., the ranks of the right hand sides with matrix B involved might be of order n. As a result of this problem, a computation of the HSVs, and thus also the introduced BT approach, becomes ineffective.

3.1 Hankel singular value approximation

Assuming that C having high rank (e.g., $C = I_n$), $p \sim n$, and $m \ll n$, the observability Gramians as solutions of (8a) and (9a) are hardly computable. Returning to (7a) and (7c) based on the original formulation of Moore [12], $\tilde{W}_r$ and $\tilde{T}_r$ can also be computed by means of $S_p^T E^T \mathcal{G}_{po} E S_p$ or $S_i^T A^T \mathcal{G}_{io} A S_i$ via the proper and improper HSVs θ^p and θ^i, respectively. The nonzero positive proper HSVs θ_j^p are computable as

$$\begin{aligned} \theta_j^p = \sqrt{\lambda_j(\tilde{\mathcal{G}}_{po}\mathcal{G}_{pc})} &= \sqrt{\lambda_j(\tilde{\mathcal{G}}_{po} S_p S_p^T)} \\ &= \sqrt{\lambda_j(S_p^T E^T \mathcal{G}_{po} E S_p)}, \end{aligned}$$

where $\tilde{\mathcal{G}}_{po} = E^T \mathcal{G}_{po} E$ and $\lambda_j(\cdot)$ denotes the j-th eigenvalue of matrix $(\cdot)$. With $\tilde{\mathcal{G}}_{io} = A^T \mathcal{G}_{io} A$, the improper HSVs θ_j^i are available via

$$\begin{aligned} \theta_j^i = \sqrt{\lambda_j(\tilde{\mathcal{G}}_{io}\mathcal{G}_{ic})} &= \sqrt{\lambda_j(\tilde{\mathcal{G}}_{io} S_i S_i^T)} \\ &= \sqrt{\lambda_j(S_i^T A^T \mathcal{G}_{io} A S_i)}. \end{aligned}$$

Replacing $\mathcal{G}_{po}$ in (7a) by $S_p^T E^T \mathcal{G}_{po} E S_p$ and $\mathcal{G}_{io}$ in (7c) by $S_i^T A^T \mathcal{G}_{io} A S_i$ leads to

$$\begin{aligned} S_p^T \tilde{\mathcal{G}}_{po} S_p &= \frac{1}{2\pi} \int_{-\infty}^{\infty} S_p^T E^T (-i\omega E - A)^{-T} P_r^T C^T \times \\ &\quad C P_r (i\omega E - A)^{-1} E S_p d\omega \\ &= \frac{1}{\pi} \int_0^{\infty} S_p^T E^T (-i\omega E - A)^{-T} P_r^T C^T \times \\ &\quad C P_r (i\omega E - A)^{-1} E S_p d\omega, \end{aligned} \tag{14}$$

and

$$\begin{aligned} S_i^T \tilde{\mathcal{G}}_{io} S_i &= \frac{1}{2\pi} \int_0^{2\pi} S_i^T A^T (e^{-i\omega} E - A)^{-T} (I - P_r)^T C^T \times \\ &\quad C (I - P_r)(e^{i\omega} E - A)^{-1} A S_i d\omega. \end{aligned} \tag{15}$$

By means of a suitable quadrature rule, the approximation of the matrix products

$$S_p^T \tilde{\mathcal{G}}_{po} S_p \approx \frac{1}{2\pi} \sum_{l=0}^{N_p} \lambda_l f_p(\omega_l) \tag{16}$$

and

$$S_i^T \tilde{\mathcal{G}}_{io} S_i \approx \frac{1}{2\pi} \sum_{l=0}^{N_i} \lambda_l f_i(\omega_l) \tag{17}$$

with

$$f_p(\omega) = \underbrace{S_p^T E^T (-i\omega E - A)^{-T} P_r^T C^T \underbrace{C P_r \underbrace{(i\omega E - A)^{-1} E S_p}_{a.)}}_{b.)}}_{c.)} \tag{18}$$

and

$$f_i(\omega) = \underbrace{S_i^T A^T (e^{-i\omega} E - A)^{-T} \tilde{P}_r^{\,T} C^T \underbrace{C \tilde{P}_r \underbrace{(e^{i\omega} E - A)^{-1} A S_i}_{a.)}}_{b.)}}_{c.)} \tag{19}$$

with $\tilde{P}_r = I - P_r$ can be performed with reasonable numerical effort. In detail, the evaluation of (18) and (19) exploits symmetry of f_p and f_i and therefore only needs the 3 emphasized basic steps *a.), b.)* and *c.)*. Step a.) is equivalent to solving a large-scale but sparse linear system of equations with only a few right hand sides, recalling that we assume S_p and S_i to be low rank factors, which facilitates the numerical treatment of this step. Although the solution does not need to be sparse anymore (most likely sparsity is lost), step b.) consists only of a manageable number of matrix vector multiplications. If the matrix C is equal to the identity matrix, this step simplifies even more. If not explicitly given, step b.) in (19) requires matrix subtraction in $\tilde{P}_r$, so that we have to take care of possible catastrophic cancellation. Step c.) then uses the symmetry of the functions f_p and f_i, such that the temporary result just needs to be multiplied by its conjugate transpose. Using an efficient quadrature rule to approximate the matrix product, these three steps become sufficient. Efficiency in this spirit means the quadrature rule needs to be adaptive, highly accurate, and should provide an error estimation. We choose the Gauss-Kronrod quadrature formula explained in the next section. The goal to compute the reduced system requires an additional intermediate step during the evaluation of f_p and f_i which causes acceptable numerical extra costs. This additional step is explained in Section 3.3.

3.2 The Gauss-Kronrod quadrature formula

We want to approximately calculate the value of the integrals (14) and (15) by means of the Gauss-Kronrod quadrature formula. A basis is the n-point Gaussian quadrature rule for the integration over a domain $[a, b]$

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} \alpha_i f(x_i^G) =: \mathbf{G}_n$$

with n quadrature points x_i^G (roots of the n-th Legendre polynomial) and nonnegative weights α_i. The Gauss quadrature approximates polynomials of degree $2n - 1$ or less exactly. The Gauss-Kronrod quadrature (GK) is an enlargement of the n-point Gauss

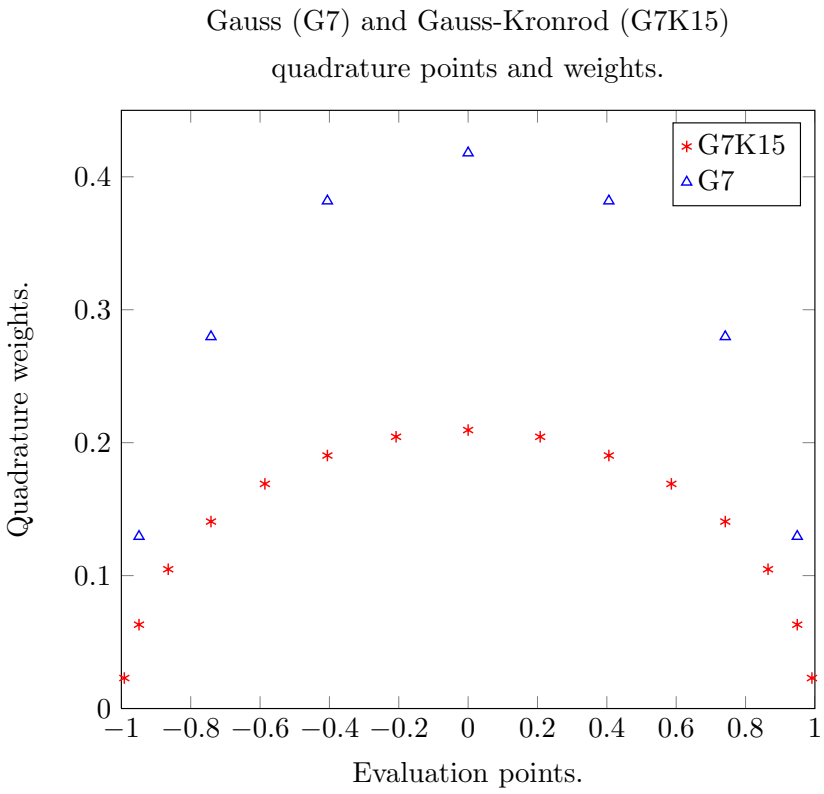


Figure 1: Quadrature points and weights of G7K15.

quadrature with $n+1$ new quadrature points and different weights a_i and b_i such that

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} a_i f(x_i^G) + \sum_{i=1}^{n+1} b_i f(x_i^K) =: \mathbf{K}_{2n+1}.$$

GK is a nested quadrature rule because it uses the same evaluation points as the Gauss quadrature, which is embedded in GK but of lower order. The Gauss points do not require additional function evaluations. GK leads to exact results for polynomials up to degree $3n+1$. Figure 1 shows $n = 7$ Gauss and $2n+1 = 15$ Gauss-Kronrod evaluation points and quadrature weights in the interval $[-1, 1]$. This so-called *Gauss-Kronrod pair* $(\mathbf{G}_7, \mathbf{K}_{15})$ is a standard choice referring to the number of supporting points and hence the method we apply. The difference between the Gauss approximation and the GK quadrature gives an error estimation. Here, $err = (200|\mathbf{G}_n - \mathbf{K}_{2n+1}|)^{1.5}$ is the standard choice how to compute this estimation. Since this choice is based on experiences and there is no guarantee for correctness, we consider $err = |\mathbf{G}_7 - \mathbf{K}_{15}|$, which mostly overestimates the error, as sufficient. For details see [28, Section 5.5].

We combine this very efficient method to approximate general integrals with an adaptive refinement of the integration domain, such that in each subdomain $[\alpha^k, \beta^k]$ of $[0, \infty]$ or $[0, 2\pi]$, respectively, the estimated approximation error is smaller then a user defined threshold. Alternatively, the sum of the errors over all subintervals is smaller than a given threshold. Both guarantees a certain accuracy within the entire interval. The refinement itself happens within the interval $[0, 1]$. Therefor, we transform the evaluation points $\{x_i^G\} \bigcup \{x_i^K\} =: x_i^{GK} \in [-1, 1]$, $i = 1, \ldots, 15$ to elements of the particular k-th subdomain $y_i^k \in [a^k, b^k] \subset [0, 1]$ via

$$y_i^k = \phi(x_i^{GK}) = \frac{b^k - a^k}{2} x_i^{GK} + \frac{b^k + a^k}{2},$$

such that

$$\int_{a^k}^{b^k} f(y)dy = \frac{b^k - a^k}{2} \int_{-1}^{1} f\left(\frac{b^k - a^k}{2} x + \frac{b^k + a^k}{2}\right) dx.$$

Next, we map these points to the respective integration domain. In the proper case, $y_i^k \in [a^k, b^k]$ is mapped to $\omega_i^k \in [\alpha^k, \beta^k] \subset [0, \infty)$ by means of $\omega_i^k = \psi(y_i^k) = \frac{y_i^k}{1 - y_i^{k\,2}}$, such that

$$\begin{aligned} \int_0^\infty f_p(\omega)d\omega &= \sum_k \int_{\alpha^k}^{\beta^k} f_p(\omega)d\omega \\ &= \sum_k \int_{a^k}^{b^k} \frac{1 + y^2}{(1 - y^2)^2} f_p\left(\frac{y}{1 - y^2}\right) dy. \end{aligned}$$

The improper case with $[\alpha^k, \beta^k] \subset [0, 2\pi]$ leads by means of $\omega_i^k = \xi(y_i^k) = 2\pi y_i^k$ to

$$\int_0^{2\pi} f_i(\omega)d\omega = \sum_k \int_{\alpha^k}^{\beta^k} f_i(\omega)d\omega = \sum_k 2 \int_{a^k}^{b^k} f_i\left(2\pi y\right) dy.$$

In this way, we apply the GK quadrature scheme adaptively. We check by means of the error estimate if the values are below the given threshold. If this holds, we are done. If not, we bisect the corresponding interval $[a^k, b^k]$ and repeat the whole process, including the mapping of the evaluation points for each refined subdomain. We iterate this procedure until the approximation satisfies the desired error tolerance. Numerical results can be found in Section 4. For more details of the computation of the Gauss-Kronrod quadrature, see [29, 30].

3.3 Getting the reduced order model

Although we know approximations to $S_p^T \tilde{\mathcal{G}}_{po} S_p$ and $S_i^T \tilde{\mathcal{G}}_{io} S_i$, obtaining a reduced order model requires additional calculation steps. The goal is to obtain the projection matrices $\tilde{W}_r$ and $\tilde{T}_r$ of (13). In the remainder of this subsection, we concentrate on the proper part. All techniques explained hold for the improper part, too.

Following the BT Schur method approach in [13] and its application in [31], $\tilde{W}_r$ and $\tilde{T}_r$ are the result of a Schur decomposition of

$$S_p^T \tilde{\mathcal{G}}_{po} S_p \in \mathbb{R}^{n_{pc} \times n_{pc}} \quad \text{and} \quad S_i^T \mathcal{G}_{io} S_i \in \mathbb{R}^{n_{ic} \times n_{ic}} \tag{20}$$

with $n_{pc}, n_{ic} \ll n$, see (6). This means we solve a small-size eigenvalue problem from whose solution $\tilde{W}_r$ and $\tilde{T}_r$ can be recovered. Assuming, we want to reduce the proper part of the system to order r_p, we need to compute the left and right invariant subspaces corresponding to the r_p largest (magnitude) eigenvalues by means of both Gramians, i.e.,

$$(\mathcal{G}_{pc} \tilde{\mathcal{G}}_{po}) V_r = V_r \Lambda_p, \tag{21}$$

with $\Lambda_p = \operatorname{diag}(\lambda_1, \lambda_2 \ldots, \lambda_{r_p})$, $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_{r_p}| > 0$ and

$$V_l^T (\mathcal{G}_{pc} \tilde{\mathcal{G}}_{po}) = \Lambda_p V_l^T. \tag{22}$$

The order r_p of the reduced proper part of the system can, for example suggested in [31], be determined by the ordered eigenvalues λ_i of $S_P^T \tilde{\mathcal{G}}_{po} S_P$ (only of size $n_{pc} \times n_{pc}$ with $n_{pc} \ll n$ and therefor cheaply computable) and a given tolerance, such that

$$\min_{r_p \in \mathbb{N}} \left(2 \sum_{i=r_p+1}^{n} \lambda_i \leq tol \right). \tag{23}$$

To derive the small eigenvalue problems belonging to the matrices in (20), considering (6), we study in detail the underlying right and left eigenvalue problems (21) and (22). We multiply (21) with $S_P^T \tilde{\mathcal{G}}_{po}$ from the left

$$S_P^T \tilde{\mathcal{G}}_{po} S_p S_P^T \tilde{\mathcal{G}}_{po} V_r = S_P^T \tilde{\mathcal{G}}_{po} V_r \Lambda_p$$

and additionally substitute $\tilde{V}_r := S_P^T \tilde{\mathcal{G}}_{po} V_r$, such that

$$(S_P^T \tilde{\mathcal{G}}_{po} S_p) \tilde{V}_r = \tilde{V}_r \Lambda_p. \tag{24}$$

We know an approximation to $S_P^T \tilde{\mathcal{G}}_{po} S_p$, see (16), so we can easily solve this problem. The goal is to compute V_r, the required right dominant invariant subspace of the system. Consequently, multiplying (24) with S_p from the left

$$S_p S_P^T \tilde{\mathcal{G}}_{po} S_p \tilde{V}_r = S_p \tilde{V}_r \Lambda_p \tag{25}$$

leads by

$$V_r := S_p \tilde{V}_r \tag{26}$$

to (21). That means we solve (24) and compute V_r via (26). It needs to be clarified that, as $z \neq 0$ in (24) with $z \in \operatorname{colspan}(\tilde{V}_r)$, the statement $x \neq 0$ with $x \in \operatorname{colspan}(V_r)$ holds in (21). Assume $x = 0$ in (26), i. e.,

$$S_p z = 0.$$

Algorithm 3.1 Evaluation of $f_p(\omega)$ including additional step

1: **for** $l \leftarrow 1, N_p$ **do**
2: $\quad f := (i\omega_l E - A)^{-1}(ES_p)$;
3: $\quad f := CP_r f$;
4: $\quad g := (-i\omega_l E - A)^{-T}(P_r^T C^T f)$;
5: $\quad f := f^T f$;
6: **end for**

It follows, that

$$S_p S_p^T \tilde{\mathcal{G}}_{po} S_p z = 0,$$

and by (25) we get $\lambda z = 0$. Due to $\lambda \neq 0$, the only option is $z = 0$.

The analog arguments are valid for the calculation of the left dominant invariant subspace. Consider the large-size left eigenvalue problem (22). By multiplication with S_p from the right and again a substitution $\tilde{V}_l^T := V_l^T S_p$, we derive a small eigenvalue problem, i.e.,

$$\tilde{V}_l^T (S_P^T \tilde{\mathcal{G}}_{po} S_P) = \Lambda_p \tilde{V}_l^T. \tag{27}$$

After calculating $\tilde{V}_l^T$, we determine V_l^T by multiplying $S_P^T \tilde{\mathcal{G}}_{po}$ from the right

$$\underbrace{\tilde{V}_l^T S_P^T \tilde{\mathcal{G}}_{po}}_{V_l^T} S_P S_P^T \tilde{\mathcal{G}}_{po} \quad = \quad \Lambda_p \underbrace{\tilde{V}_l^T S_P^T \tilde{\mathcal{G}}_{po}}_{V_l^T},$$

such that

$$V_l := \tilde{\mathcal{G}}_{po} S_P \tilde{V}_l = E^T \mathcal{G}_{po} E S_P \tilde{V}_l. \tag{28}$$

Here, it is also easy to show, that $0 \neq x \in \operatorname{colspan}(V_l)$ is guaranteed if $0 \neq z \in \operatorname{colspan}(\tilde{V}_l)$, but the knowledge of $E^T \mathcal{G}_{po} E S_P$ is required in (28). Fortunately, this information is cheaply available by an additional computational step. We approximate $\mathcal{G}_{po} E S_P$ (and if needed later multiply E^T from the left), such that we can calculate $\hat{V}_l = \mathcal{G}_{po} E S_P \tilde{V}_l$. We simply add Step 4 referring to function g in Algorithm 3.1 while evaluating $f_p(\omega)$, see (18).

Knowing the bases of the left and right dominant invariant subspaces, we continue following the approach of [13] generalized to descriptor systems. We apply the SVD to the product of V_l^T and V_r, such that

$$V_l^T V_r = U_p \Sigma_p V_p^T. \tag{29}$$

We show that, in comparison to the BT square root approach, the matrices in (10) and (29) span the same subspaces. We assume $\mathcal{G}_{pc} = S_p S_p^T$ and $\mathcal{G}_{po} = R_p R_p^T$ with $S_p, R_p \in \mathbb{R}^{n \times n_f}$ to be exact low rank factorizations. If the pencil $\lambda E - A$ is regular and stable, it follows by [11, Thm. 3.2.8.] that

$$\begin{aligned} \operatorname{rank}(\mathcal{G}_{pc} \tilde{\mathcal{G}}_{po}) &= \operatorname{rank}(\mathcal{G}_{pc} E^T \mathcal{G}_{po} E^T) \\ &= \operatorname{rank}(S_p^T E^T \mathcal{G}_{po} E^T S_p) = n_f. \end{aligned} \tag{30}$$

Algorithm 3.2 BTMT for $m \ll n$ and $\mathcal{O}(p) = \mathcal{O}(n)$.

Input: Realization A, E, B, C, such that $\lambda E - A$ is regular and stable with $m \ll n$ and $\mathcal{O}(p) = \mathcal{O}(n)$.

Output: Reduced order realization A_r, E_r, B_r, C_r.

1: Compute S_p and S_i of (6) as solution of (8b) and (9b).
2: Approximate $S_p^T \tilde{\mathcal{G}}_{po} S_p$ and $S_i^T \tilde{\mathcal{G}}_{io} S_i$ by (16) – (19).
3: Compute $\tilde{W}_r$ and $\tilde{T}_r$ in (31) by means of the explained Schur and singular value decompositions.
4: Calculate the reduced system by projection in (13).

Using that, from (24) and (27) we know $\tilde{V}_l, \tilde{V}_r \in \mathbb{R}^{n_f \times n_f}$ are of full rank. Regarding (26) and using that $\tilde{V}_r$ is invertible, we get $\operatorname{span}(V_r) = \operatorname{span}(S_p)$. Due to (30), $\operatorname{rank}(R_p^T E S_p) = n_f$, such that $R_p^T E S_p \tilde{V}_l$ is of full rank. From (28) it follows, that $\operatorname{span}(V_l) = \operatorname{span}(E^T R_p)$, which implies that $\operatorname{span}(\hat{V}_l) = \operatorname{span}(R_p)$.

Analogously, for the improper part we calculate

$$\begin{aligned} (S_i^T \tilde{G}_{io} S_i)\tilde{V}_r^i = \tilde{V}_r^i \Lambda_i \quad &\text{with} \quad V_r^i = S_i \tilde{V}_r^i, \\ \tilde{V^i}_l^T (S_i^T \tilde{\mathcal{G}}_{io} S_i) = \Lambda_i \tilde{V^i}_l^T \quad &\text{with} \quad V_l^i = A^T \mathcal{G}_{io} A S_i \tilde{V}_l^i, \end{aligned}$$

and also $\hat{V}_l^i = \mathcal{G}_{io} A S_i \tilde{V}_l^i$. Please note that there is no truncation in the improper part. We keep all r_i improper eigenvalues $\lambda_i \neq 0$ and their corresponding eigenvectors. Again, applying the SVD leads to

$$\tilde{V^i}_l^T V_r^i = U_i \Sigma_i V_i^T.$$

Similar to (12), the projectors W_r and T_r are defined as

$$\tilde{W}_r = \left[\hat{V}_l U_p \Sigma_p^{-\frac{1}{2}}, \hat{V}_l^i U_i \Sigma_i^{-\frac{1}{2}}\right] \quad \text{and} \tag{31a}$$

$$\tilde{T}_r = \left[V_r V_p \Sigma_p^{-\frac{1}{2}}, V_r^i V_i \Sigma_i^{-\frac{1}{2}}\right]. \tag{31b}$$

The reduced model of order $r := r_p + r_i$ is determined by (13). The complete approach is summarized in Algorithm 3.2. The case when we have many inputs and only a few outputs can be treated in the same way.

4 Numerical Investigations

4.1 Heat transfer within a hollow cylinder

In [5], the solution of an inverse heat transfer problem is investigated. One of the introduced models is a cylinder in which, due to a drilling process, a heat source exists. The goal is to find out the initial temperature by means of measurements on

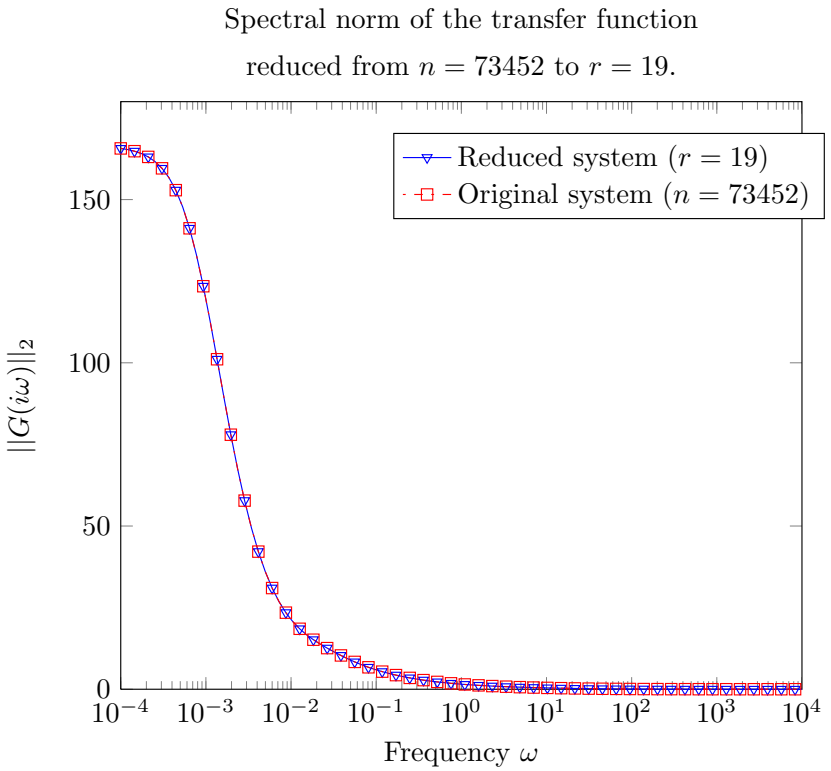


Figure 2: Transfer function of the hollow cylinder..

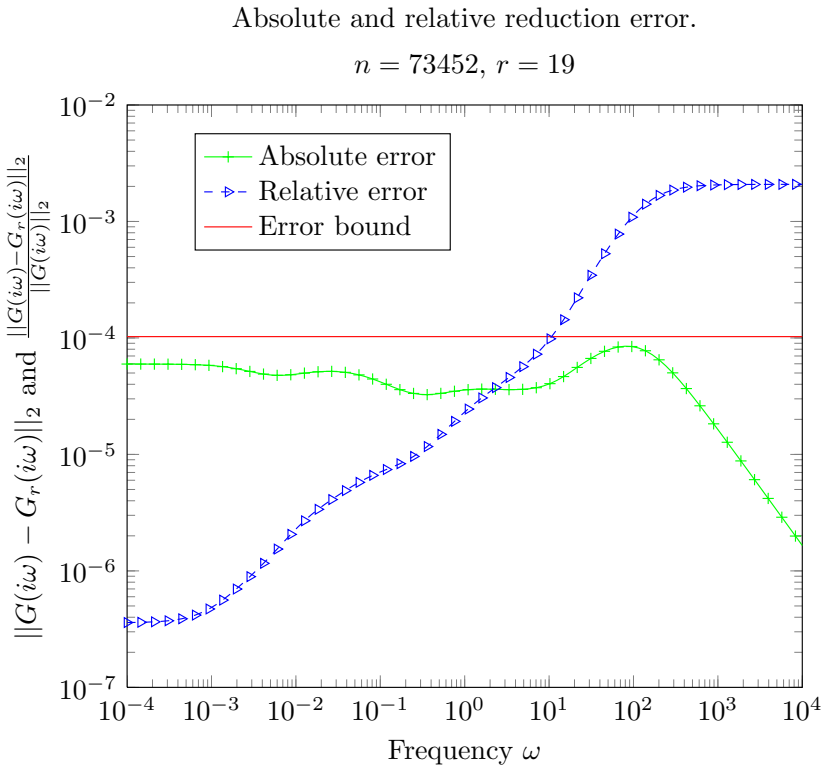


Figure 3: Error plot of the reduction of the hollow cylinder.

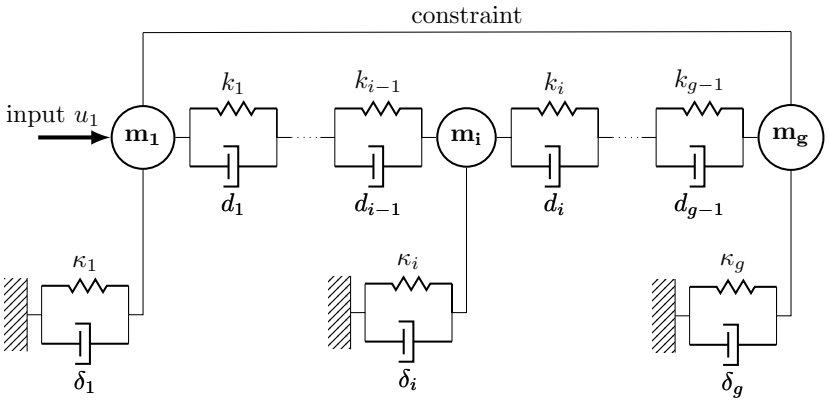


Figure 4: A damped mass-spring system with flexible number of masses and a holonomic constraint [11].

the outside. The underlying equations can be written as

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

where f is an additional linear perturbation term which has no influence on the reduction process of the coefficient matrices. It contains for example the influence of the external temperature and only needs to be adapted to the reduced state space as a final step. For more details see [5]. Let us now assume we know the initial temperature at the inner boundary of the hollow cylinder but want to know how the material expands over time with changing temperatures. Therefore, it is necessary to know the temperature at each point of the discretization grid. Consequently, the output matrix is given by $C = I_n$. We now reduce by means of the BTMT approach. Figure 2 shows the spectral norm of the original ($n = 73452$) and the reduced system ($r = 19$) along the imaginary axis. In Figure 3, the absolute and relative approximation error is shown. The BT error bound computed as $2\sum_{i=r+1}^{n}\sqrt{\lambda_i}$ with λ_i as in (23) holds for all frequencies. Depending on the quality of the approximation of the Gramians information this is not necessarily always the case in BTMT. We see that the reduced model is much better than typically needed in mechanical heat transfer applications.

4.2 Constrained damped mass-spring system of index 3

As a second example, we consider a constrained damped mass-spring system introduced., e.g., in [11, eq. (3.34)], see Figure 4. In contrast to [11], the wish to observe every mass in terms of position and velocity leads to the following system:

$$\begin{aligned} \dot{p}(t) &= v(t), \\ M\dot{v}(t) &= Kp(t) + Dv(t) - G^T\lambda(t) + B_2u(t), \\ 0 &= Gp(t), \\ y_1(t) &= C_1p(t), \quad \text{and} \quad y_2(t) = C_2v(t). \end{aligned} \tag{32}$$

Here, $p(t) \in \mathbb{R}^g$ denotes the position and $v(t) \in \mathbb{R}^g$ the velocity of the masses m_i, $i = 1, \ldots, g$. Furthermore, $\lambda(t) \in \mathbb{R}^2$ is the Lagrange multiplier, M the mass matrix, D the damping matrix, K the stiffness matrix, and $G = [1, 0, ..., 0, -1]$ the matrix containing the constraint of the fixed bonding of the first and the last mass. The projectors P_l and P_r can be computed explicitly. For details see [11]. The input matrix is $B_2 = e_1$, where e_1 denotes the first column of I_g. This means, as input we can influence the velocity of the first mass. The output matrices $C_1 = C_2 = I_g$ give the velocity and the position of every mass. In form of (1), i.e.,

$$x = \begin{bmatrix} p(t) \\ v(t) \\ \lambda(t) \end{bmatrix}, \quad E = \begin{bmatrix} I_g & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \text{and} \quad A = \begin{bmatrix} 0 & I_g & 0 \\ K & D & -G^T \\ G & 0 & 0 \end{bmatrix},$$

system (32) is an index 3 descriptor system with one input but $2g$ outputs. To get comparable results we take the same setting for the masses and the spring and damping

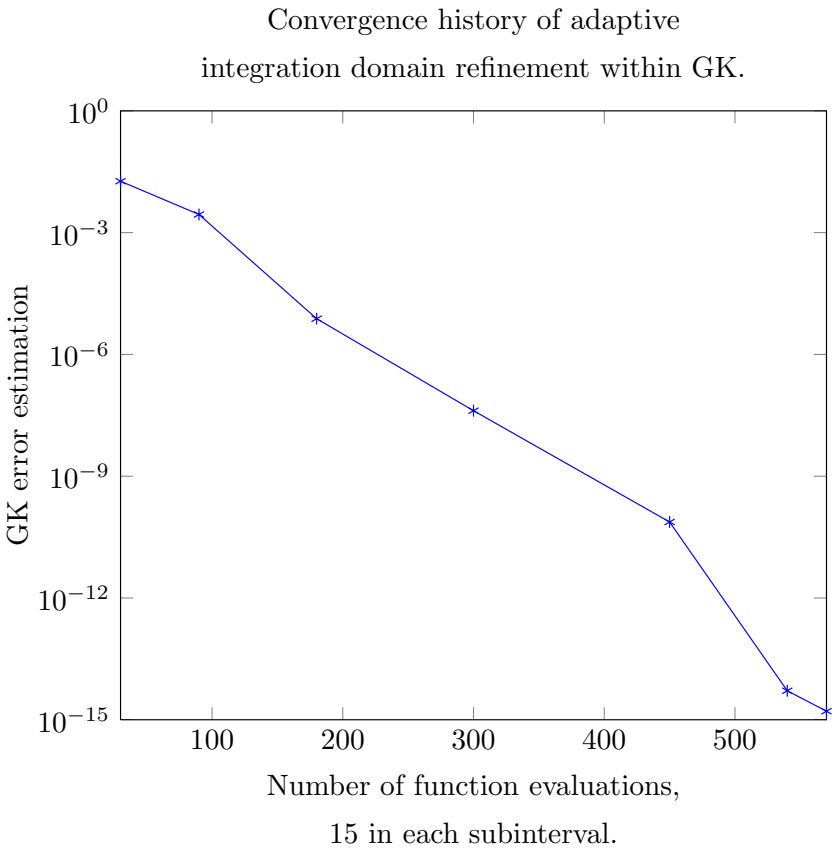
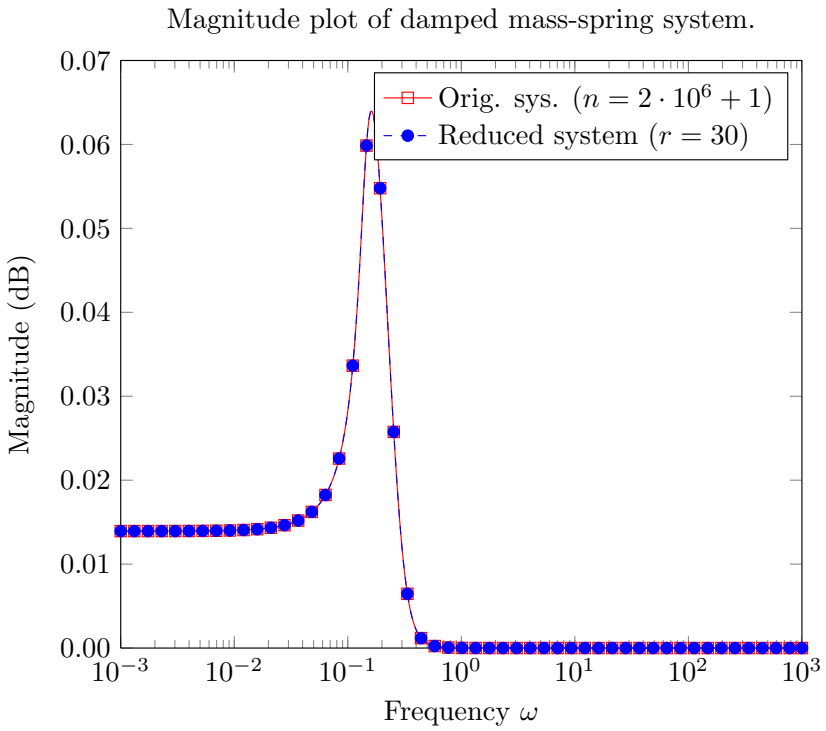
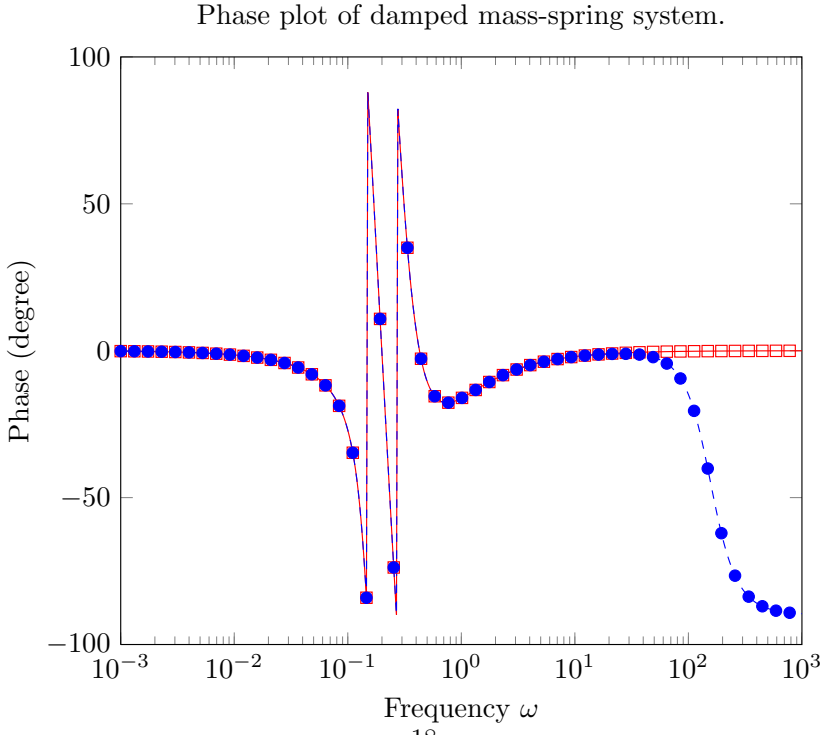


Figure 5: Convergence history (proper part) of the damped mass-spring system.



(a) Magnitude plot



(b) Phase plot

Figure 6: Bode plot of the original and the reduced damped mass-spring system.

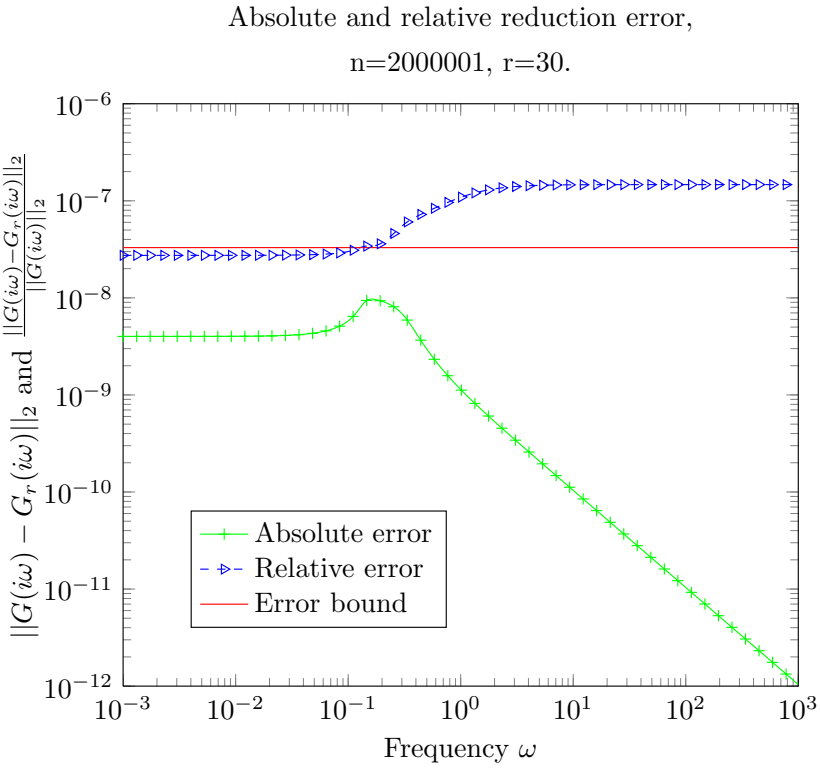


Figure 7: Error plot of the reduced damped mass-spring system.

constants as in [11]. Choosing $g = 10^6$, we get $n = 2 \cdot 10^6 + 1$, $m = 1$, and $p = 2 \cdot 10^6$. That huge number n makes the Bartels-Stewart algorithm and Hammarling's method unusable. Figure 5 shows the estimated error $err = |\mathbf{G}_7 - \mathbf{K}_{15}|$ depending on the level of integration domain refinement. In Figure 7 we see the relative and absolute reduction error as well as the BT error bound. In the used setting this bound holds for all frequencies. Figure 6 shows the magnitude and the phase plot of the input to third output (position of the third mass) component.

5 Conclusions

We introduced a new balanced truncation approach for descriptor systems with either many inputs or many outputs, called BTMT. The approach mainly enables the reduction of these systems which can be used to test different model settings like the influence of the external temperature in Example 4.1. Remember, that we can not expect, e.g., the observability Gramian to be representable in low rank form. Due to the introduced approach of using the GK quadrature, this problem for usual large-scale Lyapunov solvers is circumvented. Nevertheless, the solution of the linear systems in steps 2 and 4 of Algorithm 3.1 remains the numerical bottleneck and leads to a time consuming reduction process. The computational complexity of evaluating, e.g., $f_p(\omega)$ in (16) explained in Algorithm 3.1, is, assuming that all matrices are dense, $\mathcal{O}(N_p(\frac{4}{3}n^3 + 12n^2 n_{pc} + 3n^2))$. The n^3 and n^2 terms are caused by the dense LU factorization. Theses costs reduce for sparse computations. We see that $n_{pc} \in [1, n_f] \subset \mathbb{Z}$ makes a difference between quadratic or nearly cubic complexity. Consequently, we want n_{pc} to be as small as possible, which motivates the usage of the algorithm introduced in [32]. Additionally, the question rises how accurate the known low rank factor approximation of the (in this case) controllability Gramian needs to be. Another way of saving costs might be the replacement of step 4 of Algorithm 3.1 by the once-only computation of a pseudoinverse of S_p and multiplying this matrix from the left to $f_p(\omega)$ at the end. These problems are worth future investigations.

6 ACKNOWLEDGMENTS

The authors want to thank Tobias Breiten, Norman Lang, and Martin Stoll for their support. Special thanks go to Tatjana Stykel for providing her experience, the ADI codes and the damped mass-spring example.

References

[1] L. Dai, Singular Control Systems, Vol. 118 of Lecture Notes in Control and Information Sciences, Springer-Verlag, Berlin, 1989.

[2] A. C. Antoulas, D. C. Sorensen, S. Gugercin, A survey of model reduction methods for large-scale systems, Contemporary Mathematics 280 (2001) 193–219.

[3] J. Saak, Efficient numerical solution of large scale algebraic matrix equations in PDE control and model order reduction, Ph.D. thesis, TU Chemnitz, available from http://www.tu-chemnitz.de/~saak/Data/Diss-web.pdf (2009).

[4] H. Egger, Y. Heng, W. Marquardt, A. Mhamdi, Efficient solution of a three-dimensional inverse heat conduction problem in pool boiling, IOPscience – Inverse Problems 25 (9) (2009) 95006 (19pp).

[5] N. Lang, Lösen der inversen Wärmeleitungsgleichung als linear quadratisches Regelungsproblem, Master's thesis, TU Chemnitz, www.mpi-magdeburg.mpg.de/mpcsc/lang/Data/diplarb.pdf (2011).

[6] H. Antil, M. Heinkenschloss, R. W. Hoppe, D. C. Sorensen, Domain decomposition and model reduction for the numerical solution of PDE constrained optimization problems with localized optimization variables, Computing and Visualization in Science 13 (6) (2010) 249–264. doi:10.1007/s00791-010-0142-4.

[7] H. Antil, M. Heinkenschloss, R. W. Hoppe, Domain decomposition and balanced truncation model reduction for shape optimization of the Stokes system, Optimization Methods and Software 26 (4-5) (2011) 643–669. doi:10.1080/10556781003767904.

[8] M. D. Gunzburger, S. Manservisi, Analysis and Approximation of the Velocity Tracking Problem for Navier-Stokes Flows with Distributed Control, SIAM Journal on Numerical Analysis 37 (5) (2000) pp. 1481–1512.

[9] D. Schmidthausler, S. Schöps, M. Clemens, Reduction of Linear Subdomains for Non-Linear Electro-Quasistatic Field Simulations, IEEE Transactions on Magnetics 49 (5) (2013) 1669–1672. doi:10.1109/TMAG.2013.2238905.

[10] A. C. Antoulas, Approximation of Large-Scale Dynamical Systems, SIAM Publications, Philadelphia, PA, 2005.

[11] V. Mehrmann, T. Stykel, Balanced truncation model reduction for large-scale systems in descripter form, in: P. Benner, V. Mehrmann, D. Sorensen (Eds.), Dimension Reduction of Large-Scale Systems, Vol. 45 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, Berlin/Heidelberg, Germany, 2005, pp. 83–115.

[12] B. C. Moore, Principal component analysis in linear systems: controllability, observability, and model reduction, IEEE Trans. Automat. Control AC-26 (1) (1981) 17–32.

[13] M. Safonov, R. Chiang, A Schur method for balanced-truncation model reduction, IEEE Trans. Automat. Control 34 (7) (1989) 729–733. doi:10.1109/9.29399.

[14] G. W. Stewart, J.-G. Sun, Matrix perturbation theory, Academic Press, INC., Boston, Massachusetts, UK, 1990.

[15] R. E. Kalman, P. L. Falb, M. A. Arbib, Topics in Mathematical System Theory, McGraw-Hill, New York, 1969.

[16] E. D. Sontag, Mathematical Control Theory, 2nd Edition, Springer-Verlag, New York, NY, 1998.

[17] T. Stykel, Analysis and numerical solution of generalized Lyapunov equations, Ph.D. thesis, Technische Universität Berlin (2002).

[18] F. D. Freitas, J. Rommes, N. Martins, Gramian-Based Reduction Method Applied to Large Sparse Power System Descriptor Models, IEEE Trans. Power Systems 23 (3) (2008) 1258–1270. doi:10.1109/TPWRS.2008.926693.

[19] T. Penzl, Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case, Sys. Control Lett. 40 (2000) 139–144.

[20] J.-R. Li, J. White, Low rank solution of Lyapunov equations, SIAM J. Matrix Anal. Appl. 24 (1) (2002) 260–280.

[21] S. Gugercin, D. C. Sorensen, A. C. Antoulas, A modified low-rank Smith method for large-scale Lyapunov equations, Numerical Algorithms 32 (2003) 27–55.

[22] P. Benner, J.-R. Li, T. Penzl, Numerical solution of large Lyapunov equations, Riccati equations, and linear-quadratic control problems, Numer. Lin. Alg. Appl. 15 (9) (2008) 755–777.

[23] T. Stykel, Numerical solution and perturbation theory for generalized Lyapunov equations, Linear Algebra Appl. 349 (1-3) (2002) 155–185. doi:10.1016/S0024-3795(02)00255-0.

[24] T. Stykel, V. Simoncini, Krylov subspace methods for projected Lyapunov equations, Applied Numerical Mathematics 62 (1) (2012) 35–50. doi:10.1016/j.apnum.2011.09.007.

[25] T. Stykel, Solving projected generalized Lyapunov equations using SLICOT, in: Proceedings of the IEEE International Symposium on Computer-Aided Control Systems Design, 2006, pp. 14 –18. doi:10.1109/CACSD-CCA-ISIC.2006.4776617.

[26] T. Stykel, Low-rank iterative methods for projected generalized Lyapunov equations, Electronic Transactions on Numerical Analysis 30 (2008) 187–202.

[27] A. Varga, Minimal realization procedures based on balancing and related techniques, in: F. Pichler, R. Moreno-Díaz (Eds.), Computer Aided Systems Theory EUROCAST '91, Vol. 585 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 1992, pp. 733–761. doi:10.1007/BFb0021056.

[28] D. Kahaner, C. B. Moler, S. Nash, G. E. Forsythe, Numerical methods and software, Prentice-Hall Series in Computational Mathematics, Prentice Hall, 1989.

[29] G. Monegato, An overview of the computational aspects of Kronrod quadrature rules, Numer. Algorithms 26 (2001) 173–196.

[30] D. Calvetti, G. H. Golub, W. B. Gragg, L. Reichel, Computation of Gauss-Kronrod quadrature rules, Math. Comp 69 (2000) 1035–1052.

[31] U. Baur, P. Benner, Cross-Gramian based model reduction for data-sparse systems, Electronic Trans. Numerical Analysis 31 (2008) 256–270.

[32] P. Benner, T. Breiten, On optimality of interpolation-based low rank approximations of large-scale matrix equations, Tech. Rep. MPIMD/11-10, Max Planck Institute Magdeburg Preprints (December 2011).

Max Planck Institute Magdeburg Preprints