TECHNISCHE UNIVERSITÄT CHEMNITZ

Ulrike Baur Peter Benner

Gramian-Based Model Reduction for Data-Sparse Systems

CSC/07-01

Chemnitz Scientific Computing
Preprints

Impressum:

Chemnitz Scientific Computing Preprints — ISSN 1864-0087
(1995–2005: Preprintreihe des Chemnitzer SFB393)

Herausgeber:
Professuren für
Numerische und Angewandte Mathematik
an der Fakultät für Mathematik
der Technischen Universität Chemnitz

Postanschrift:
TU Chemnitz, Fakultät für Mathematik
09107 Chemnitz
Sitz:
Reichenhainer Str. 41, 09126 Chemnitz

http://www.tu-chemnitz.de/mathematik/csc/

TECHNISCHE UNIVERSITÄT CHEMNITZ

Chemnitz Scientific Computing Preprints

Ulrike Baur Peter Benner

Gramian-Based Model Reduction for Data-Sparse Systems

CSC/07-01

Abstract

Model reduction is a common theme within the simulation, control and optimization of complex dynamical systems. For instance, in control problems for partial differential equations, the associated large-scale systems have to be solved very often. To attack these problems in reasonable time it is absolutely necessary to reduce the dimension of the underlying system. We focus on model reduction by balanced truncation where a system theoretical background provides some desirable properties of the reduced-order system. The major computational task in balanced truncation is the solution of large-scale Lyapunov equations, thus the method is of limited use for really large-scale applications. We develop an effective implementation of balancing-related model reduction methods in exploiting the structure of the underlying problem. This is done by a data-sparse approximation of the large-scale state matrix A using the hierarchical matrix format. Furthermore, we integrate the corresponding formatted arithmetic in the sign function method for computing approximate solution factors of the Lyapunov equations. This approach is well-suited for a class of practical relevant problems and allows the application of balanced truncation and related methods to systems coming from 2D and 3D FEM and BEM discretizations.

CSC/07-01 ISSN 1864-0087 February 2007

Contents

Author's addresses:

Ulrike Baur, Peter Benner

TU Chemnitz
Fakultät für Mathematik
D-09107 Chemnitz

[ubaur,benner]@mathematik.tu-chemnitz.de

1 Introduction

The dynamical systems considered here are described for the continuous-time case by a differential equation, the *input-to-state equation*, and an algebraic equation, the *output equation*,

$$\begin{array}{rcll} \dot{x}(t) & = & Ax(t) + Bu(t), & t > 0, \qquad x(0) = x^0, \\ y(t) & = & Cx(t) + Du(t), & t \geq 0, \end{array} \tag{1}$$

with constant matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$, and $D \in \mathbb{R}^{p \times m}$. That is, we consider a linear, time-invariant (LTI) system. The vector $u(t) \in \mathbb{R}^m$ contains the input variables, $y(t) \in \mathbb{R}^p$ the output variables, and $x(t) \in \mathbb{R}^n$ denotes the vector of state variables. Applying the Laplace transformation to (1) under the assumption $x^0 = 0$ yields the connection between input and output variables in the frequency domain as $y(s) = G(s)u(s)$, where

$$G(s) = C(sI - A)^{-1}B + D$$

is the transfer function matrix (TFM) associated to the system (1), see, e.g. [48]. The complexity (*order*) of such a system is measured by the dimension n of the state-space. Often, in practice, e.g., in the control of partial differential equations (PDEs), the system matrix A comes from the spatial discretization of some partial differential operator. In this case, n is typically large (often $n \geq \mathcal{O}(10^4)$) and the system matrices are sparse. On the other hand, boundary element discretizations of integral equations lead to large-scale *dense* matrices that often have a data-sparse representation [41, 30, 40]. Hence, in general, we will not assume sparsity of A, but we will assume that a data-sparse representation of A exists. Usually, the number of inputs and outputs in practical applications is small compared to the number of states, so that it is reasonable to assume $m, p \ll n$ for the rest of this paper.

Alternatively, we consider LTI systems which are discretized in time

$$\begin{array}{rcll} x_{j+1} & = & Ax_j + Bu_j, & x_0 = x^0, \\ y_j & = & Cx_j + Du_j, & \end{array} \tag{2}$$

for $j = 0, 1, 2, \ldots$. The dimensions of the matrices are equal to those in the continuous-time setting. The TFM in discrete-time is obtained by applying the $\mathcal{Z}$-transformation (see, e.g., [34, Section 11]) to (2), yielding

$$G(z) = C(zI - A)^{-1}B + D.$$

Large-scale discrete-time LTI systems arise for instance when applying a full discretization scheme to a control problem for a time-dependent linear PDE [19].

In this paper, we will restrict our attention to stable systems, that is, all eigenvalues of the coefficient matrix A, denoted by $\Lambda(A)$, are assumed to be in the open left half plane $\mathbb{C}^-$ for continuous-time systems or in the interior of the unit disk for discrete-time systems. These properties are also refered to as A being *Hurwitz* in the continuous-time setting or A being *Schur stable* or *convergent* in the discrete-time case. This is typically the case for systems arising from the discretization of parabolic partial differential equations like the heat equation or linear convection-diffusion equations.

Model reduction aims at approximating a given large-scale system (1) or (2) by a system of reduced order r, $r \ll n$. In system theory and control of ordinary differential equations (ODEs), balanced truncation [36] and related methods are the methods of choice since they have some desirable properties: they preserve the stability of the system and provide a global computable error bound which allows an adaptive choice of the reduced order r. The basic approach relies on balancing the Gramians of the systems. For continuous-time systems, they are given by the solutions of the *Lyapunov equations*

$$A\mathcal{P} + \mathcal{P}A^T + BB^T = 0, \quad A^T\mathcal{Q} + \mathcal{Q}A + C^TC = 0, \tag{3}$$

while in the discrete-time case, they solve the *Stein* (or discrete-time Lyapunov) *equations*

$$\mathcal{P} = BB^T + A\mathcal{P}A^T, \quad \mathcal{Q} = C^TC + A^T\mathcal{Q}A. \tag{4}$$

Thus, the major part of the computational complexity of these methods stems from the solution of these two large-scale matrix equations. In general, numerical methods for linear matrix equations have a complexity of $\mathcal{O}(n^3)$ (see, e.g., [20, 42]) and therefore, all these approaches are restricted to problems of moderate size. To overcome this limitation for a special class of practically relevant large-scale systems, we consider modifications of a class of algorithms that allow the use of data-sparse matrix formats. In particular, we will describe iterative solvers for matrix equations based on the *sign function method* for continuous-time systems and on the *squared Smith method* in discrete time, incorporating data-sparse matrix approximations and a corresponding formatted arithmetic in the iteration scheme. The main properties of balanced truncation and of model reduction by singular perturbation approximation are described at the beginning of Section 2. The iterative solvers for the matrix equations are briefly illustrated in Section 2.3. In Section 3.1, we give a short introduction of the data-sparse matrix format employed here, the so called hierarchical matrix format ($\mathcal{H}$-matrix format), and describe the modified iterations in Section 3.2. By integrating the new solvers in the model reduction routines as done in Section 3.3 we obtain efficient methods of linear-polylogarithmic complexity which combine the desirable features of balanced truncation methods with structural information of the underlying PDE. Some accuracy results are presented in Section 4 and several numerical experiments demonstrate the performance of the new methods in Section 5.

2 Theoretical Background

2.1 Model Reduction by Balanced Truncation

Model reduction aims at eliminating some of the state variables of the original large-scale system. We will first focus on the continuous-time case, the discrete-time model reduction will be explained at the end of this section. Considering again the LTI system (1), then the task in model reduction is to find another LTI system

$$\begin{array}{rcll} \dot{\hat{x}}(t) & = & \hat{A}\hat{x}(t) + \hat{B}u(t), \qquad t > 0, \qquad \hat{x}(0) = \hat{x}^0, \\ \hat{y}(t) & = & \hat{C}\hat{x}(t) + \hat{D}u(t), \qquad t \geq 0, \end{array} \tag{5}$$

with reduced state-space dimension $r \ll n$ and $\hat{A} \in \mathbb{R}^{r \times r}$, $\hat{B} \in \mathbb{R}^{r \times m}$, $\hat{C} \in \mathbb{R}^{p \times r}$, $\hat{D} \in \mathbb{R}^{p \times m}$. The associated TFM $\hat{G}(s) = \hat{C}(sI - \hat{A})^{-1}\hat{B} + \hat{D}$ should approximate $G(s)$ in some sense. We are interested in a small error norm $\|G - \hat{G}\|_\infty$ where $\|\cdot\|_\infty$ denotes the $\mathcal{H}_\infty$-norm of a rational transfer function. In the scalar case, this 2-induced operator norm equals the peak magnitude of the transfer function on the imaginary axis, i.e., $\sup_{\omega \in \mathbb{R}} |G(\jmath\omega)|$ with $\jmath = \sqrt{-1}$, whereas in the multivariable case the following definition holds:

$$\|G\|_\infty := \sup_{\omega \in \mathbb{R}} \sigma_{\max}(G(\jmath\omega)),$$

where $\sigma_{\max}$ denotes the maximum singular value of a matrix. By driving both systems with the same input u, the worst output error $\|y - \hat{y}\|_2$ can be minimized by minimizing $\|G - \hat{G}\|_\infty$ because

$$\|y - \hat{y}\|_2 \leq \|G - \hat{G}\|_\infty \|u\|_2$$

due to the submultiplicativity property of the $\mathcal{H}_\infty$-norm [48].

One of the classical approaches to model reduction is balanced truncation, see, e.g., [2, 37, 48] and the references therein. The main principle of balanced truncation and of balancing-related model reduction is finding a particular state-space basis in which we can easily determine the states, which will be truncated. These states should have small impact on the system behavior concerning both, reachability and observability. Such a system representation, where states which are difficult to observe are also difficult to reach and vice-versa, is obtained by a balancing transformation. The required state-space transformation, $x \to Tx$, $T \in \mathbb{R}^{n \times n}$ non-singular, leads to a balanced realization of the original system

$$(A, B, C, D) \to (TAT^{-1}, TB, CT^{-1}, D),$$

where the *reachability Gramian* $\mathcal{P}$ and the *observability Gramian* $\mathcal{Q}$ are equal and diagonal:

$$\mathcal{P} = \mathcal{Q} = \Sigma := \operatorname{diag}\{\sigma_1, \ldots, \sigma_n\}.$$

For minimal systems (that is, the system order is minimal and thus equals the *McMillan degree* of the system), there always exist balancing transformations and we have

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0.$$

The numbers σ_i are called the *Hankel singular values* (HSVs) of the LTI system (1). They are given as the square roots of the eigenvalues of the product of the Gramians: $\sigma_i = \sqrt{\lambda_i(\mathcal{P}\mathcal{Q})}$, where $\mathcal{P}$ and $\mathcal{Q}$ are the unique positive definite solutions of the two dual Lyapunov equations in (3) corresponding to (1). The HSVs are system invariants as

$$\Lambda\left((T\mathcal{P}T^T)(T^{-T}\mathcal{Q}T^{-1})\right) = \Lambda\left(T\mathcal{P}\mathcal{Q}T^{-1}\right) = \{\sigma_1^2, \ldots, \sigma_n^2\}.$$

They provide a systematic way to identify the states which are least involved in the energy transfer from inputs to outputs. For a system in balanced coordinates an energy interpretation, see e.g. [46], determines these states as those which correspond to small HSVs. If we truncate the states corresponding to the $n-r$ smallest HSVs from a balanced realization we obtain a reduced-order model of size r where the worst output error is bounded [21]:

$$\|y - \hat{y}\|_2 \leq 2\left(\sum_{j=r+1}^{n} \sigma_j\right) \|u\|_2. \tag{6}$$

This error bound provides a nice way to adapt the selection of the reduced order. In addition, the reduced-order system remains stable and balanced with HSVs σ_1 to σ_r of the original system.

The *square root method* (SR method) of balanced truncation is based on Cholesky factors of the Gramians $\mathcal{P} = SS^T$ and $\mathcal{Q} = RR^T$. The approach computes projection matrices which balance a given minimal system and simultaneously truncate states corresponding to small HSVs. The SR method can also be applied to non-minimal systems where we have $\operatorname{rank}(S) < n$ and/or $\operatorname{rank}(R) < n$, see [32, 45]. In these papers it is also observed, that we need not compute the whole transformation matrix T. An efficient implementation of this method was proposed in [13] where the solution factors are computed as full-rank factors $S \in \mathbb{R}^{n \times r_\mathcal{P}}$, $R \in \mathbb{R}^{n \times r_\mathcal{Q}}$, where $r_\mathcal{P}$ and $r_\mathcal{Q}$ denote the ranks of the Gramians $\mathcal{P}$ respectively $\mathcal{Q}$. This is of particular interest for large-scale computation if the Gramians have low rank at least numerically, so we have reduced memory requirements for the solution factors. An additional benefit of this ansatz is that all computational costs are of order $\mathcal{O}(r_\mathcal{P} r_\mathcal{Q} n)$ during the computation of the reduced-order system as soon as the matrix equations (3) are solved.

A similarity relation between the product of the full-rank factors and the square root of the Gramian product, $(\mathcal{P}\mathcal{Q})^{1/2} \sim S^T R$, suggests to compute an SVD of $S^T R$ for obtaining a balancing transformation. The method requires only the computation of the parts U_1, V_1 and Σ_1 of the SVD

$$S^T R = \left[\begin{array}{cc} U_1 & U_2 \end{array}\right] \left[\begin{array}{cc} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{array}\right] \left[\begin{array}{c} V_1^T \\ V_2^T \end{array}\right], \tag{7}$$

where $\Sigma_1 = \operatorname{diag}\{\sigma_1, \ldots, \sigma_r\}$. If we assume that $r_{\mathcal{P}} \leq r_{\mathcal{Q}}$, we have $\Sigma_2 = (\bar{\Sigma}\ 0)$ and $\bar{\Sigma} = \operatorname{diag}\{\sigma_{r+1}, \ldots, \sigma_{r_{\mathcal{P}}}\}$. The case $r_{\mathcal{P}} > r_{\mathcal{Q}}$ can be treated analogously. If there is a significant gap between σ_r and σ_{r+1}, $\sigma_r \gg \sigma_{r+1}$, the splitting in (7) seems natural. We compute parts $T_l \in \mathbb{R}^{r \times n}$ and $T_r \in \mathbb{R}^{n \times r}$ of the balancing transformation matrices T and T^{-1}, respectively, where $T_l T_r = I_r$,

$$T_l = \Sigma_1^{-1/2} V_1^T R^T, \qquad T_r = S U_1 \Sigma_1^{-1/2},$$

apply them to (1),

$$(\hat{A}, \hat{B}, \hat{C}, \hat{D}) = (T_l A T_r, T_l B, C T_r, D),$$

and end up with a balanced and reduced-order stable system of order r.

In the discrete-time setting we are looking for a reduced-order system

$$\begin{array}{rcl} \dot{\hat{x}}_{j+1} & = & \hat{A}\hat{x}_j + \hat{B}u_j, \qquad \hat{x}_0 = \hat{x}^0, \\ \hat{y}_j & = & \hat{C}\hat{x}_j + \hat{D}u_j, \end{array} \tag{8}$$

for $j = 0, 1, 2, \ldots$, and $\hat{A} \in \mathbb{R}^{r \times r}$, $\hat{B} \in \mathbb{R}^{r \times m}$, $\hat{C} \in \mathbb{R}^{p \times r}$, $\hat{D} \in \mathbb{R}^{p \times m}$. Again, the goal is to preserve stability and to approximate $G(z)$ by $\hat{G}(z) = \hat{C}(zI - \hat{A})^{-1}\hat{B} + \hat{D}$. Balanced truncation methods for discrete LTI systems (2) are performed analogously to the continuous-time case. The only difference is the computation of the two Gramians, which are in the discrete-time setting the unique, symmetric and positive semidefinite solutions of two Stein equations (4). Note that in the discrete-time case, the reduced-order model will in general not be balanced [37, Section 1.9].

2.2 Model Reduction with Singular Perturbation Approximation

Model reduction by balanced truncation performs well at high frequencies as

$$\lim_{\omega \to \infty} (G(\jmath\omega) - \hat{G}(\jmath\omega)) = D - \hat{D} = 0.$$

In some situations we are more interested in a reduced-order model with perfect matching of the transfer function G at $\omega = 0$. In state-space this corresponds to a zero steady-state error. Zero steady-state errors can be obtained by a *singular perturbation approximation* (SPA) to the original system [35, 47], also called *balanced residualization*. Assume the realization of the system (1) is minimal (otherwise use balanced truncation to reduce the order to the McMillan degree of the system) and balanced. Then, in the continuous-time setting, consider the following partitioned representation

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u, \\ y &= \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + D\, u, \end{aligned}$$

where $A_{11} \in \mathbb{R}^{r \times r}$, $B_1 \in \mathbb{R}^{r \times m}$, $C_1 \in \mathbb{R}^{p \times r}$ and r is the desired reduced order. Neglecting the dynamics of the faster state variables x_2 by setting $\dot{x}_2(t) = 0$ and assuming A_{22} to be nonsingular, we obtain a reduced-order model as in (5) with

$$\begin{aligned} \hat{A} &:= A_{11} - A_{12}A_{22}^{-1}A_{21}, & \hat{B} &:= B_1 - A_{12}A_{22}^{-1}B_2, \\ \hat{C} &:= C_1 - C_2A_{22}^{-1}A_{21}, & \hat{D} &:= D - C_2A_{22}^{-1}B_2. \end{aligned} \tag{9}$$

The balanced truncation error bound (6) holds as well and the SPA reduced-order model additionally satisifies $\hat{G}(0) = G(0)$ and provides a good approximation at low frequencies.

For discrete-time systems, the formulae

$$\begin{aligned} \hat{A} &:= A_{11} + A_{12}(I - A_{22})^{-1}A_{21}, & \hat{B} &:= B_1 + A_{12}(I - A_{22})^{-1}B_2, \\ \hat{C} &:= C_1 + C_2(I - A_{22})^{-1}A_{21}, & \hat{D} &:= D + C_2(I - A_{22})^{-1}B_2, \end{aligned} \tag{10}$$

yield an SPA, where the resulting reduced-order system is stable and balanced and its TFM fulfills $\hat{G}(e^{\jmath \cdot 0}) = \hat{G}(1) = G(1) = G(e^{\jmath \cdot 0})$ [37, Section 1.9].

2.3 Solution of Linear Matrix Equations

It has already been noted in the introduction that solving the Lyapunov equations (3) associated to continuous-time systems and the discrete analogon called Stein equations (4) is the main computational task in balanced truncation and related methods. Therefore we will describe solvers for these matrix equations which are particularly adapted for the purpose of model reduction.

A well-suited iterative scheme for solving *stable Lyapunov equations* (that is, Lyapunov equations with stable A) is based on the sign function method. Roberts [39] introduced the sign function method for the solution of Lyapunov equations (or of the more general Riccati equations). Consider the two dual Lyapunov equations (3)

$$A\mathcal{P} + \mathcal{P}A^T + BB^T = 0, \quad A^T\mathcal{Q} + \mathcal{Q}A + C^TC = 0$$

and an initialization given by $A_0 = A$, $B_0 = B$ and $C_0 = C$. We compute the two Gramians simultaneously by the following iteration:

$$\begin{aligned} A_{j+1} &\leftarrow \frac{1}{2}(A_j + A_j^{-1}), \\ B_{j+1}B_{j+1}^T &\leftarrow \frac{1}{2}(B_jB_j^T + A_j^{-1}B_jB_j^TA_j^{-T}), \\ C_{j+1}^TC_{j+1} &\leftarrow \frac{1}{2}(C_j^TC_j + A_j^{-T}C_j^TC_jA_j^{-1}), \quad j = 0, 1, 2, \ldots, \end{aligned}$$

with quadratic convergence rate and

$$\mathcal{P} = \frac{1}{2}\lim_{j\to\infty} B_jB_j^T, \quad \mathcal{Q} = \frac{1}{2}\lim_{j\to\infty} C_j^TC_j.$$

In [8, 11], this iteration scheme is modified for the direct computation of the Cholesky (or full-rank) factors which are needed in the SR method. To obtain the Gramians in factorized form, we partition the iteration as follows:

$$\begin{aligned} A_{j+1} &\leftarrow \frac{1}{2}(A_j + A_j^{-1}), \\ B_{j+1} &\leftarrow \frac{1}{\sqrt{2}}\left[\ B_j, \quad A_j^{-1}B_j\ \right], \\ C_{j+1} &\leftarrow \frac{1}{\sqrt{2}}\left[\begin{array}{c} C_j \\ C_jA_j^{-1} \end{array}\right], \quad j = 0, 1, 2, \ldots, \end{aligned} \tag{11}$$

see [11] for details. The matrices $S = \frac{1}{\sqrt{2}}\lim_{j\to\infty} B_j$ and $R^T = \frac{1}{\sqrt{2}}\lim_{j\to\infty} C_j$ are solution factors as

$$\mathcal{P} = SS^T = \frac{1}{2}\lim_{j\to\infty} B_jB_j^T, \quad \mathcal{Q} = RR^T = \frac{1}{2}\lim_{j\to\infty} C_j^TC_j.$$

In many large-scale applications it can be observed that the eigenvalues of the Gramians decay rapidly, in particular when $n \gg m, p$, see e.g., [3, 24, 38]. Then the memory requirements for storing the solution factors as well as the computational costs of the over-all balanced truncation algorithm can be considerably reduced by computing low-rank approximations to the factors directly. Since the sizes of the matrices B_j and C_j in (11) are doubled in each iteration step, it is proposed in [11] to apply a rank-revealing QR factorization (RRQR) [22] in order to reveal the expected low numerical rank and to limit the exponentially growing number of rows and columns. The modified iteration scheme for solving Lyapunov equations is explained in detail in [11, 6].

For the numerical solution of the two dual Stein equations (4),

$$\mathcal{P} = BB^T + A\mathcal{P}A^T, \quad \mathcal{Q} = C^TC + A^T\mathcal{Q}A,$$

we consider a fixed point iteration scheme called *squared Smith iteration* [43] with initializations $A_0 = A$, $B_0 = B$, and $C_0 = C$:

$$\begin{aligned} B_{j+1}B_{j+1}^T &\leftarrow A_j B_j B_j^T A_j^T + B_j B_j^T, \\ C_{j+1}^T C_{j+1} &\leftarrow A_j^T C_j^T C_j A_j + C_j^T C_j, \\ A_{j+1} &\leftarrow A_j^2, \quad j = 0, 1, 2, \ldots. \end{aligned}$$

The iteration converges quadratically to the Gramians as

$$\mathcal{P} = \lim_{j\to\infty} B_j B_j^T, \quad \mathcal{Q} = \lim_{j\to\infty} C_j^T C_j,$$

if the matrix A is Schur stable. Some remarks concerning convergence theory and overflow are presented in [12]. A problem adapted variant can be found in [14] for the direct computation of low-rank approximations to the full-rank or Cholesky factors of the solutions. With the modified iteration scheme

$$\begin{aligned} B_{j+1} &\leftarrow \begin{bmatrix} B_j, & A_j B_j \end{bmatrix}, \\ C_{j+1} &\leftarrow \begin{bmatrix} C_j \\ C_j A_j \end{bmatrix}, \\ A_{j+1} &\leftarrow A_j^2, \quad j = 0, 1, 2, \ldots, \end{aligned} \tag{12}$$

we obtain convergence to the solution factors $S = \lim_{j\to\infty} B_j$ and $R^T = \lim_{j\to\infty} C_j$.

This iteration is less expensive during the first iteration steps, if we assume $n \gg m, p$. But clearly this advantage gets lost caused by the doubling of workspace in the first two lines of the iteration (12). As mentioned already for the continuous-time case, we expect that the Gramians have a low numerical rank so that the iterates also remain of low numerical rank. To exploit this property and to avoid the exponential growth of the matrices, we apply a RRQR to B_{j+1}^T and C_{j+1} in each iteration step as proposed in [14]. We review this row compression for the computation of a low-rank approximation to S:

$$B_{j+1}^T = Q_{j+1}\hat{B}_{j+1}\Pi_{j+1} = Q_{j+1}\begin{bmatrix} \hat{B}_{j+1}^{11} & \hat{B}_{j+1}^{12} \\ 0 & \hat{B}_{j+1}^{22} \end{bmatrix}\Pi_{j+1}. \tag{13}$$

Here Π_{j+1} is a permutation matrix, Q_{j+1} is orthogonal and $\hat{B}_{j+1}^{11}$ is a $\mathbb{R}^{m_{j+1}\times m_{j+1}}$ matrix. The order m_{j+1} of $\hat{B}_{j+1}^{11}$ denotes the numerical rank of B_{j+1} determined by a threshold τ. Given a threshold τ, the numerical rank of a matrix with singular values $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n \geq 0$ is the largest r such that $\mu_r > \mu_1\tau$. In the RRQR, the 2-norm condition number is estimated by $\mathrm{cond}_2\left(\hat{B}_{j+1}^{11}\right) \leq 1/\tau$. Thus, only the entries in the upper triangular part of $\hat{B}_{j+1}$, that is the well-conditioned part of the matrix, have to be stored for obtaining an approximate solution $\tilde{S} = \lim_{j\to\infty} \tilde{B}_j$, with

$$\tilde{B}_{j+1} := \begin{bmatrix} \hat{B}_{j+1}^{11} & \hat{B}_{j+1}^{12} \end{bmatrix}\Pi_{j+1}.$$

We thus have reduced storage requirements of order $\mathcal{O}(r_{\mathcal{P}}n)$ since the numerical rank of each iterate is bounded by the numerical rank of the Gramian.

Despite the low memory requirements for the approximate solution factors we still have storage requirements of order $\mathcal{O}(n^2)$ and $\mathcal{O}(n^3)$ operations during both iterations (11), (12) for the iterates A_j. Therefore we will integrate a data-sparse matrix format and the corresponding approximate arithmetic in the iteration schemes. This format and the modified algorithms will be described in the next section.

3 Solvers Based on Data-Sparse Approximation

3.1 $\mathcal{H}$-Matrix Arithmetic Introduction

In [26], the sign function method for solving algebraic Riccati equations is combined with a data-sparse matrix representation and a corresponding approximate arithmetic. This initiated the idea to use the same approach for solving Lyapunov equations as these are special cases of algebraic Riccati equations. As our approach also makes use of this $\mathcal{H}$-matrix format, we will introduce some of its basic facts in the following.

The $\mathcal{H}$-matrix format is a data-sparse representation for a special class of matrices, which often arise in applications. Matrices that belong to this class result, for instance, from the discretization of partial differential or integral equations. Exploiting the special structure of these matrices in computational methods yields reduced computing time and memory requirements. A detailed description of the $\mathcal{H}$-matrix format can be found, e.g. in [23, 25, 30, 31].

The basic idea of the $\mathcal{H}$-matrix format is to partition a given matrix recursively into submatrices that admit low-rank approximations. To determine such a partitioning, we consider a product index set $I \times I$, where $I = \{1, \dots, n\}$ corresponds to a finite element or boundary element basis $(\varphi_i)_{i \in I}$. The product index set is hierarchically partitioned into blocks $r \times s$, where we stop the block splitting as soon as the corresponding submatrix $M_{|r\times s}$ admits a low-rank approximation

$$\operatorname{rank}(M_{|r\times s}) \leq k.$$

An hierarchically partitioned product index is called block $\mathcal{H}$-tree and is denoted by $T_{I\times I}$. The suitable blocks in $T_{I\times I}$ are determined by a problem dependent admissibility condition. The submatrices corresponding to admissible leaves are stored in factorized form as *Rk-matrices* (matrices of rank at most k)

$$M_{|r\times s} = AB^T, \quad A \in \mathbb{R}^{r\times k},\, B \in \mathbb{R}^{s\times k}.$$

The remaining inadmissible (but small) submatrices corresponding to leaves are stored as usual full matrices. The set of $\mathcal{H}$-matrices of block-wise rank k based on $T_{I\times I}$ is denoted by $\mathcal{M}_{\mathcal{H},k}(T_{I\times I})$. The storage requirements for a matrix $M \in \mathcal{M}_{\mathcal{H},k}(T_{I\times I})$ are

$$\mathcal{N}_{\mathcal{M}_{\mathcal{H},k}St} = \mathcal{O}(n\log(n)k)$$

instead of $\mathcal{O}(n^2)$ for the original (full) matrix. We denote by $M_{\mathcal{H}}$ the hierarchical approximation of a matrix M.

The formatted arithmetic $\oplus$, $\ominus$, $\odot$ on the set of $\mathcal{H}$-matrices is defined by using standard arithmetic for the full matrices in the inadmissible blocks. In the Rk-matrix blocks we apply standard arithmetic followed by a truncation, that maps the submatrices (which, e.g. in case of addition generically have rank $2k$) back to the Rk-format. The truncation operator, denoted by $\mathcal{T}_k$, can be achieved by a truncated singular value decomposition and results in a best Frobenius and spectral norm approximation, see, e.g., [25] for more details. For $\mathcal{H}$-matrices the truncation operator $\mathcal{T}_{\mathcal{H},k} : \mathbb{R}^{n\times m} \to \mathcal{M}_{\mathcal{H},k}(T_{I\times I})$, $M \mapsto \tilde{M}$, is defined blockwise for all leaves of $T_{I\times I}$ by

$$\tilde{M}_{|_{r\times s}} := \begin{cases} \mathcal{T}_k(M_{|_{r\times s}}) & \text{if } r\times s \text{ admissible,} \\ M_{|_{r\times s}} & \text{otherwise.} \end{cases}$$

For two matrices $A, B \in \mathcal{M}_{\mathcal{H},k}(T_{I\times I})$ and a vector $v \in \mathbb{R}^n$ we consider the formatted arithmetic operations, which all have linear-polylogarithmic complexity:

$$\begin{array}{llll} v \mapsto Av : & & & \mathcal{O}(n\log(n)k), \\ A\oplus B & = & \mathcal{T}_{\mathcal{H},k}(A+B) : & \mathcal{O}(n\log(n)k^2), \\ A\odot B & = & \mathcal{T}_{\mathcal{H},k}(AB) : & \mathcal{O}(n\log^2(n)k^2), \\ \mathrm{Inv}_{\mathcal{H}}(A) & = & \mathcal{T}_{\mathcal{H},k}(\tilde{A}^{-1}) : & \mathcal{O}(n\log^2(n)k^2). \end{array} \tag{14}$$

Here, $\tilde{A}^{-1}$ denotes the approximate inverse of A which is computed by using the Frobenius formula (obtained by block-Gaussian elimination on A under the assumption that all principal submatrices of A are non-singular) with formatted addition and multiplication. In some situations it is recommended to compute the inverse V of a matrix A using an approximate $\mathcal{H}$-LU factorization $A \approx L_{\mathcal{H}}U_{\mathcal{H}}$ followed by an $\mathcal{H}$-forward ($L_{\mathcal{H}}W = (I)_{\mathcal{H}}$) and $\mathcal{H}$- backward substitution ($U_{\mathcal{H}}V = W$).

Note that it is also possible to choose the rank adaptively for each matrix block instead of using a fixed rank k. Depending on a given approximation error ϵ, the approximate matrix operations are exact up to ϵ in each block. The truncation operator for the Rk-matrices is then changed in the following way:

$$\mathcal{T}_\epsilon(A) = \operatorname{argmin}\left\{\operatorname{rank}(R) \,\middle|\, \frac{\|R-A\|_2}{\|A\|_2} \le \epsilon\right\},$$

where the parameter ϵ determines the desired accuracy in each matrix block. Using the corresponding truncation operator $\mathcal{T}_{\mathcal{H},\epsilon}$ of hierarchical matrices changes the formatted arithmetic in (14) to a so-called adaptive arithmetic.

We will use the $\mathcal{H}$-matrix structure to compute solution factors of Lyapunov and of Stein equations, which reduces the complexity and the storage requirements of the underlying iteration scheme.

3.2 Sign Function and Smith Iterations with Formatted Arithmetic

We consider the modified iteration schemes (11) and (12) for the direct computation of full-rank solution factors S and R of the Gramians $\mathcal{P}$ and $\mathcal{Q}$. If we consider the amount of memory which is needed throughout the iterations, we remark reduced requirements for the solution factors if we apply a RRQR factorization (13) in each iteration step. But in the other part of the iteration schemes, the part for the iterates A_j, we still have memory requirements of order $\mathcal{O}(n^2)$. In this part, we also have computational cost of order $\mathcal{O}(n^3)$ caused by inversion or multiplication of $n \times n$ matrices. Therefore, we approximate A and its iterates in $\mathcal{H}$-matrix format and replace the standard operations by the hierarchical matrix arithmetic (compare with Section 3.1). The matrices B_j and C_j, which yield the solution factors at the end of the iteration, are stored in the usual "full" format. In these parts of the iteration, arithmetic operations from standard linear algebra packages such as LAPACK [1] and BLAS [33] can be used.

For the sign function iteration (11) we replace the inversion of A_j by computing an approximate $\mathcal{H}$-LU factorization as described in the previous section (the inverse is denoted by V):

$$\begin{aligned}
A_{j+1} &\leftarrow \frac{1}{2}(A_j \oplus V), \\
B_{j+1} &\leftarrow \frac{1}{\sqrt{2}}\left[\begin{array}{cc} B_j, & VB_j \end{array}\right], \\
C_{j+1} &\leftarrow \frac{1}{\sqrt{2}}\left[\begin{array}{c} C_j \\ C_jV \end{array}\right], \quad j = 0, 1, 2, \ldots.
\end{aligned}$$

Since $\lim_{j\to\infty} A_j = -I_n$, as it was seen in Section 2.3, we choose

$$\|A_j + I_n\| \leq \text{tol}$$

as stopping criterion for the iteration. With two additional iteration steps and an appropriate choice of norm and relaxed tolerance, we usually get a sufficient accuracy due to the quadratic convergence, see [11] for details. Note that the stopping citerion is meaningful even using formatted arithmetic since the identity

Algorithm 1 Calculate approximate low rank factors $\tilde{S}$ and $\tilde{R}$ of (3)

```
INPUT: $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$; tolerances tol for convergence of (11),
    $\epsilon$ for the $\mathcal{H}$-matrix approximation error and $\tau$ for the rank detection.
OUTPUT: Approximations to full-rank factors $S$ and $R$, such that $\mathcal{P} \approx \tilde{S}\tilde{S}^T$,
    $\mathcal{Q} \approx \tilde{R}\tilde{R}^T$.
$A_0 \leftarrow (A)_{\mathcal{H}}$
$B_0 \leftarrow B$, $C_0 \leftarrow C$
$j = 0$
while $\|A_j + I_n\|_2 >$ tol do
    $[L, U] \leftarrow LU_{\mathcal{H}}(A_j)$
    Solve $LW = (I_n)_{\mathcal{H}}$ by $\mathcal{H}$-forward substitution.
    Solve $UV = W$ by $\mathcal{H}$-back substitution.
    $A_{j+1} \leftarrow \frac{1}{2}(A_j \oplus V)$
    $B_{j+1} \leftarrow \frac{1}{\sqrt{2}} \begin{bmatrix} B_j, & VB_j \end{bmatrix}$
    $C_{j+1} \leftarrow \frac{1}{\sqrt{2}} \begin{bmatrix} C_j \\ C_j V \end{bmatrix}$
    Compress columns of $B_{j+1}$, rows of $C_{j+1}$ using a RRQR with threshold $\tau$
        (see (13)).
    $j = j + 1$
end while
$\tilde{S} \leftarrow \frac{1}{\sqrt{2}} B_{j+1}$, $\tilde{R}^T \leftarrow \frac{1}{\sqrt{2}} C_{j+1}$.
```

is contained in the class of $\mathcal{H}$-matrices. A detailed description of the $\mathcal{H}$-matrix arithmetic based sign function iteration for solving Lyapunov equations (also in generalized form) can be found in [5, 6]. Based on this, we obtain Algorithm 1 which solves both equations in (3) simultaneously.

For the squared Smith iteration, we replace the multiplication of the large-scale iterates A_j by formatted arithmetic

$$\begin{aligned} B_{j+1} &\leftarrow \begin{bmatrix} B_j, & A_j B_j \end{bmatrix}, \\ C_{j+1} &\leftarrow \begin{bmatrix} C_j \\ C_j A_j \end{bmatrix}, \\ A_{j+1} &\leftarrow A_j \odot A_j, \quad j = 0, 1, 2, \ldots. \end{aligned}$$

This iteration scheme has reduced memory requirements in the expensive part of the iteration, that is for $A_j \in \mathcal{M}_{\mathcal{H},k}(T_{I \times I})$ we have a demand of order $\mathcal{O}(n \log(n) k)$ instead of $\mathcal{O}(n^2)$. The computational complexity reduces to $\mathcal{O}(n \log^2(n) k^2)$ in this part of the iteration scheme. Since the sizes of the two solution iterates $B_j \in \mathbb{R}^{n \times m_j}$ and $C_j \in \mathbb{R}^{p_j \times n}$ are bounded above by the numerical rank $r_{\mathcal{P}}$ and $r_{\mathcal{Q}}$ during the RRQR factorization, compare (13), the complexity of the iterations in lines 5.– 6. of Algorithm 2 is bounded by $\mathcal{O}(r_{\mathcal{P}} n \log(n) k)$ and $\mathcal{O}(r_{\mathcal{Q}} n \log(n) k)$,

Algorithm 2 Calculate approximate low rank factors $\tilde{S}$ and $\tilde{R}$ of (4)

INPUT: $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$; tolerances tol for convergence of (12), ϵ for the $\mathcal{H}$-matrix approximation error and τ for the rank detection.

OUTPUT: Approximations to full-rank factors S and R, such that $\mathcal{P} \approx \tilde{S}\tilde{S}^T$, $\mathcal{Q} \approx \tilde{R}\tilde{R}^T$.

```
$A_0 \leftarrow (A)_{\mathcal{H}}$
$B_0 \leftarrow B$, $C_0 \leftarrow C$
$j = 0$
while $\|A_j\|_2 > \text{tol}$ do
    $B_{j+1} \leftarrow \begin{bmatrix} B_j, & A_j B_j \end{bmatrix}$
    $C_{j+1} \leftarrow \begin{bmatrix} C_j \\ C_j A_j \end{bmatrix}$
    Compress columns of $B_{j+1}$, rows of $C_{j+1}$ using a RRQR with threshold $\tau$ (see (13)).
    $A_{j+1} \leftarrow A_j \odot A_j$
    $j = j + 1$
end while
$\tilde{S} \leftarrow B_{j+1}$, $\tilde{R}^T \leftarrow C_{j+1}$
```

respectively. Instead of a constant given rank k we will use an adaptive rank choice based on a prescribed approximation error ϵ in our numerical experiments in Section 5. In the investigated examples, i.e., discretized control problems for PDEs defined on $\Omega \subset \mathbb{R}^d$, it is observed that $k \sim \log^{d+1}(1/\epsilon)$ is sufficient to obtain a relative approximation error of $\mathcal{O}(\epsilon)$, [7].

For the squared Smith iteration we have $\lim_{j \to \infty} A_j = 0$; thus it is advised to choose

$$\|A_j\|_2 \leq \text{tol}$$

as stopping criterion for the iteration, which is easy to check. A parallel implementation of the method is described in [15]. The developed $\mathcal{H}$-matrix arithmetic based implementation of the Smith iteration is summarized in Algorithm 2, which again solves both equations in (4) simultaneously.

3.3 $\mathcal{H}$-Matrix Based Model Reduction

We integrate the $\mathcal{H}$-matrix based sign function iteration as summarized in Algorithm 1 in the SR method for balanced truncation (as introduced in Section 2.1) for computing a continuous-time system of reduced order. For discrete-time systems the sign function solver is replaced by the $\mathcal{H}$-matrix based Smith iteration as described in Algorithm 2. This is summarized in Algorithm 3. By using the

Algorithm 3 Approximate Balanced Truncation for LTI systems (1) and (2)

```
INPUT: LTI system $A_{\mathcal{H}} \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, $D \in \mathbb{R}^{p\times m}$; tolerance tol
    for the approximation error of the reduced-order model.
OUTPUT: Reduced-order model (of order $r$) $\hat{A}, \hat{B}, \hat{C}, \hat{D}$; error bound $\delta$.
1: Compute approximate full-rank factors $\tilde{S} \in \mathbb{R}^{n\times r_{\mathcal{P}}}$, $\tilde{R} \in \mathbb{R}^{n\times r_{\mathcal{Q}}}$ of the system
   Gramians using Algorithm 1 for continuous-time systems, Algorithm 2 in the
   discrete-time case.
2: Compute SVD of $\tilde{S}^T\tilde{R}$ ($\tilde{n} := \min\{r_{\mathcal{P}}, r_{\mathcal{Q}}\}$)
       $$\tilde{S}^T\tilde{R} = [\,U_1\; U_2\,] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix},$$
   with $\Sigma_1 = \mathrm{diag}(\sigma_1, \ldots, \sigma_r)$, $\Sigma_2 = \mathrm{diag}(\sigma_{r+1}, \ldots, \sigma_{\tilde{n}})$, HSVs in decreasing order
   with $\sigma_r > \sigma_{r+1}$. Adaptive choice of $r$ by given tolerance: $2\sum_{j=r+1}^{\tilde{n}} \sigma_j \leq$ tol.
3: Compute truncation matrices: $T_l = \Sigma_1^{-\frac{1}{2}} V_1^T \tilde{R}^T \in \mathbb{R}^{n\times r}$, $T_r = \tilde{S}\, U_1\, \Sigma_1^{-\frac{1}{2}} \in$
   $\mathbb{R}^{r\times n}$.
4: Compute BT reduced-order model:
       $$\hat{A} = T_l A T_r,\ \hat{B} = T_l B,\ \hat{C} = C T_r,\ \hat{D} = D$$
   and the estimate $\delta = 2\sum_{j=r+1}^{\tilde{n}} \sigma_j$ for the error bound (6).
```

formatted arithmetic for the solution of the large-scale matrix equations we reduce the computational complexity in the first stage of Algorithm 3 from $\mathcal{O}(n^3)$ to $\mathcal{O}(n \log^2(n) k^2)$. A detailed analysis of the complexity of the SR method can be found in [10]. It is shown that all subsequent steps do not contribute significantly to the cost of the algorithm as their complexity is reduced to $\mathcal{O}(r_{\mathcal{P}}\, r_{\mathcal{Q}}\, n)$. In Algorithm 4 the $\mathcal{H}$-matrix based SPA method is presented. For the computation of a balanced and minimal realization of (1) (respectively (2) in discrete-time) with McMillan degree $\hat{n}$ Algorithm 1 (Algorithm 2) is used as first stage in the model reduction process. The computed approximate low-rank factors $\tilde{S} \in \mathbb{R}^{n\times r_{\mathcal{P}}}$ and $\tilde{R} \in \mathbb{R}^{n\times r_{\mathcal{Q}}}$ of the two system Gramians are used for computing the truncation matrices T_l and T_r.

Note that if the $\mathcal{H}$-matrix based iteration schemes are used for approximating the solution of the corresponding matrix equations, e.g., $P \approx \tilde{S}\tilde{S}^T$, then it is sufficient to choose $\tau \leq \sqrt{\epsilon}$ (τ is the threshold for the numerical rank decision) to obtain $\|P - \tilde{S}\tilde{S}^T\|_2 \sim \epsilon$; see [10], although the accuracy of the solution factors is $\|S - \tilde{S}\|_2 \sim \sqrt{\epsilon}$. But for the purpose of balanced truncation, we need $\tau \sim \epsilon$ as the accuracy of the reduced-order model is affected by the accuracy of the solution factors themselves: we may assume that Algorithms 1,2 yield $\tilde{S}, \tilde{R}$ so that $S = \left[\begin{array}{cc} \tilde{S}, & E_S \end{array}\right]$ and $R = \left[\begin{array}{cc} \tilde{R}, & E_R \end{array}\right]$, where $\|E_S\|_2 \leq \tau\|S\|_2$, $\|E_R\|_2 \leq \tau\|R\|_2$. Then

$$S^T R = \left[\begin{array}{cc} \tilde{S}^T\tilde{R} & \tilde{S}^T E_R \\ E_S^T\tilde{R} & E_S^T E_R \end{array}\right].$$

Hence, the relative error introduced by using the "small" SVD, i.e., that of $\tilde{S}^T\tilde{R}$, rather than the full SVD, i.e., that of $S^T R$, is proportional to τ. Therefore, a choice of $\tau = \sqrt{\epsilon}$ would lead to an error of size $\sqrt{\epsilon}$ in the computed Hankel singular values as well as the projection matrices T_l, T_r and thus in the reduced-order model. This very rough error analysis motivates setting $\tau = \epsilon$.

Note that for both model reduction algorithms only the first $\tilde{n}$ HSVs are computed (with $\tilde{n} := \min\{r_{\mathcal{P}}, r_{\mathcal{Q}}\}$). Usually, $\tilde{n}$ equals the numerical rank of $\tilde{S}^T\tilde{R}$ with respect to τ and can thus be considered as a "numerical McMillan degree with respect to τ". Thus, the original balanced truncation error bound as given in (6) is under-estimated if $\tilde{n} < \hat{n}$ by using only the computable part

$$\delta = 2\sum_{j=r+1}^{\tilde{n}} \sigma_j$$

as approximation for the error in Algorithms 3 and 4. Moreover, a more detailed error analysis in [27, 28] suggests that the error in the computed bound δ, introduced by using approximate low-rank factors $\tilde{S}, \tilde{R}$, is also affected by $\text{cond}^2(T)$, where $A = T\Lambda T^{-1}$ is a spectral decomposition of A. Hence, for ill-conditioned T, the computed error bound may under-estimate the model reduction error significantly; see Example 5.3.

4 Accuracy of the Reduced-Order System

Besides the balanced truncation error bound (6), which measures the worst output error between the original and the reduced-order system, we introduce further errors using the $\mathcal{H}$-matrix format and the corresponding approximate arithmetic. Errors resulting from using the formatted arithmetic during the calculation can be controlled by choosing the parameter for the adaptive rank choice accordingly, see [6] for details. In this section we will specify the influence of the $\mathcal{H}$-matrix error introduced by the approximation of the original coefficient matrix A in $\mathcal{H}$-matrix format. Thus, balanced truncation is actually applied to

$$G_{\mathcal{H}}(s) := C(sI - A_{\mathcal{H}})^{-1}B + D.$$

We ignore the influence of rounding errors as they are expected to be negligible compared to the other error sources. Note that we assume B to be unaffected by the $\mathcal{H}$-matrix approximation, see also Remark 4.3.

Algorithm 4 Approximate SPA for LTI systems (1) and (2)

```
INPUT: LTI system $A_{\mathcal{H}} \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$, $D \in \mathbb{R}^{p \times m}$; tolerance tol
    for the approximation error of the reduced-order model.
OUTPUT: Reduced-order model (of order $r$) $\hat{A}, \hat{B}, \hat{C}, \hat{D}$; error bound $\delta$.
1: Compute approximate full-rank factors $\tilde{S} \in \mathbb{R}^{n \times r_{\mathcal{P}}}$, $\tilde{R} \in \mathbb{R}^{n \times r_{\mathcal{Q}}}$ of the system
   Gramians using Algorithm 1 for continuous-time systems, Algorithm 2 in the
   discrete-time case.
2: Compute SVD of $\tilde{S}^T \tilde{R}$ ($\tilde{n} := \min\{r_{\mathcal{P}}, r_{\mathcal{Q}}\}$)
       $\tilde{S}^T \tilde{R} = U \Sigma V^T,$
   with $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_{\tilde{n}})$ and HSVs in decreasing order.
3: Compute truncation matrices: $T_l = \Sigma^{-\frac{1}{2}} V^T \tilde{R}^T \in \mathbb{R}^{\hat{n} \times n}$, $T_r = \tilde{S} U \Sigma^{-\frac{1}{2}} \in$
   $\mathbb{R}^{n \times \hat{n}}$.
4: Compute balanced and minimal realization:
       $\hat{A} = T_l A T_r,\ \hat{B} = T_l B,\ \hat{C} = C T_r,\ \hat{D} = D.$
5: Partition matrices according to reduced order $r$ ($A_{11} \in \mathbb{R}^{r \times r}$), $r$ is determined
   by given tolerance: $2 \sum_{j=r+1}^{\tilde{n}} \sigma_j \leq \mathrm{tol}$,
       $\hat{A} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},\ \hat{B} = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix},\ \hat{C} = \begin{bmatrix} C_1 & C_2 \end{bmatrix}.$
6: Compute SPA reduced-order model $\hat{A}$, $\hat{B}$, $\hat{C}$, $\hat{D}$ with formulas (9) for
   continuous-time and with (10) for discrete-time systems and the estimate
   $\delta = 2 \sum_{j=r+1}^{\tilde{n}} \sigma_j$ for the error bound (6).
```

We can split the approximation error into two parts using the triangle inequality:

$$\|G - \hat{G}\|_\infty \leq \|G - G_{\mathcal{H}}\|_\infty + \|G_{\mathcal{H}} - \hat{G}\|_\infty, \tag{15}$$

where the first term accounts for the $\mathcal{H}$-matrix approximation error and the second part is taken care of by the balanced truncation error bound as well as other sources of error like those introduced by using approximate Gramians. Here, we will analyze the first term only, a complete analysis is beyond the scope of this paper and will be given elsewhere, see also [27].

We will derive some expressions and results that may also be of use if A is approximated by some other matrix $\tilde{A}$. (In our case, we will have $\tilde{A} = A_{\mathcal{H}}$.) We note that the following results are related to the perturbation theory for transfer functions derived in [44] and can partially be obtained as special cases of error bounds given there.

First, we note the identity

$$C(\jmath\omega I - A)^{-1}B - C(\jmath\omega I - \tilde{A})^{-1}B = C[(\jmath\omega I - A)^{-1}(A - \tilde{A})(\jmath\omega I - \tilde{A})^{-1}]B.$$

If we denote the TFM of the perturbed system by

$$\tilde{G}(s) := C(sI - \tilde{A})^{-1}B + D,$$

then the error can be expressed as

$$\|G - \tilde{G}\|_\infty = \sup_{\omega\in\mathbb{R}} \|C[(\jmath\omega I - A)^{-1}(A - \tilde{A})(\jmath\omega I - \tilde{A})^{-1}]B\|_2.$$

Thus

$$\|G - \tilde{G}\|_\infty \leq \|C\|_2\|B\|_2\|A - \tilde{A}\|_2 \sup_{\omega\in\mathbb{R}} \|(\jmath\omega I - A)^{-1}\|_2 \sup_{\omega\in\mathbb{R}} \|(\jmath\omega I - \tilde{A})^{-1}\|_2. \quad (16)$$

As in our application $\tilde{A}$ comes from the $\mathcal{H}$-matrix approximation of some elliptic operator, we will provide some specific bounds for matrices with the "nice" spectral properties often obtained in these situations. In the following, let $\tilde{A} = A + \Delta A$ so that $\|\Delta A\|_2$ accounts for the approximation error in A.

Theorem 4.1 *Let A and $A + \Delta A$ be stable and assume that both matrices are diagonalizable so that*

$$T^{-1}AT = \mathrm{diag}\{\lambda_1, \dots, \lambda_n\}, \quad \tilde{T}^{-1}(A + \Delta A)\tilde{T} = \mathrm{diag}\{\tilde{\lambda}_1, \dots, \tilde{\lambda}_n\}.$$

Furthermore, assume that

$$\mathrm{cond}_2\,(T)\,\|\Delta A\|_2 \leq \min_{i=1,\dots,n} |Re(\lambda_i(A))|. \quad (17)$$

Then the $\mathcal{H}_\infty$-norm of the corresponding error system $G - \tilde{G}$ is bounded by

$$\|G - \tilde{G}\|_\infty \leq \|C\|_2\|B\|_2\,\mathrm{cond}_2\,(T)\,\mathrm{cond}_2\left(\tilde{T}\right) \frac{1}{\min\limits_{i=1,\dots,n} |Re(\lambda_i(A))|^2} \mathcal{O}(\|\Delta A\|_2). \quad (18)$$

Proof: Using the notation

$$\begin{aligned} \Lambda &:= \mathrm{diag}\{\lambda_1, \dots, \lambda_n\}, \quad \tilde{\Lambda} := \mathrm{diag}\{\tilde{\lambda}_1, \dots, \tilde{\lambda}_n\}, \\ \alpha &:= \|\Delta A\|_2, \end{aligned}$$

setting

$$\mu = \min_{i=1,\dots,n} |\mathrm{Re}(\lambda_i(A))|, \quad \tilde{\mu} = \min_{i=1,\dots,n} |\mathrm{Re}(\lambda_i(A + \Delta A))|,$$

and invoking (16) yields, by simple calculations, the following bounds:

$$\begin{aligned}
\|G-\tilde{G}\|_\infty &\leq \alpha\,\|C\|_2\|B\|_2\,\mathrm{cond}_2\,(T)\,\mathrm{cond}_2\left(\tilde{T}\right)\sup_{\omega\in\mathbb{R}}\|(\jmath\omega I-\Lambda)^{-1}\|_2\sup_{\omega\in\mathbb{R}}\|(\jmath\omega I-\tilde{\Lambda})^{-1}\|_2\\
&= \alpha\,\|C\|_2\|B\|_2\,\mathrm{cond}_2\,(T)\,\mathrm{cond}_2\left(\tilde{T}\right)\max_{\lambda\in\Lambda(A)}\frac{1}{|\mathrm{Re}(\lambda)|}\max_{\tilde{\lambda}\in\Lambda(A+\Delta A)}\frac{1}{|\mathrm{Re}(\tilde{\lambda})|}\\
&\overset{(*)}{=} \alpha\,\|C\|_2\|B\|_2\,\mathrm{cond}_2\,(T)\,\mathrm{cond}_2\left(\tilde{T}\right)\frac{1}{\mu\,\tilde{\mu}}\\
&\overset{(**)}{\leq} \alpha\,\|C\|_2\|B\|_2\,\mathrm{cond}_2\,(T)\,\mathrm{cond}_2\left(\tilde{T}\right)\frac{1}{\mu(\mu-\mathrm{cond}_2\,(T)\,\alpha)}\\
&\overset{(***)}{\leq} \alpha\,\|C\|_2\|B\|_2\,\mathrm{cond}_2\,(T)\,\mathrm{cond}_2\left(\tilde{T}\right)\left(\frac{1}{\mu^2}+\frac{1}{\mu^3}\mathcal{O}(\alpha)\right)
\end{aligned}$$

The identity $(*)$ follows from the observation that the maximum of $1/\min_{\lambda\in\Lambda(A)}|\jmath\omega-\lambda|$ over the imaginary axis is taken for the eigenvalue closest to the imaginary axis, that is the eigenvalue with minimal absolute value of the real part. The estimate in $(**)$ is a consequence of the Bauer-Fike theorem, see, e.g., in [22, Theorem 7.2.2]. Due to (17) we can apply the geometric series to obtain $(***)$. □

For unitarily diagonalizable A as obtained, e.g., from a finite-differences discretization of a self-adjoint elliptic operator, the error bound (18) becomes much nicer.

Corollary 4.2 *With the same assumptions as in Theorem 4.1, and assuming additionally that A and $A+\Delta A$ are unitarily diagonalizable by*

$$U^HAU=\mathrm{diag}\{\lambda_1,\ldots,\lambda_n\},\quad \tilde{U}^H(A+\Delta A)\tilde{U}=\mathrm{diag}\{\tilde{\lambda}_1,\ldots,\tilde{\lambda}_n\},$$

we obtain the error bound

$$\|G-\tilde{G}\|_\infty\leq\|C\|_2\|B\|_2\frac{1}{\min\limits_{i=1,\ldots,n}|Re(\lambda_i(A))|^2}\mathcal{O}(\|\Delta A\|_2).$$

□

Thus, for a symmetric negative-definite A with spectrum

$$-\lambda_n\leq\ldots\leq-\lambda_1<0 \tag{19}$$

and symmetric negative-definite approximation $\tilde{A}$ we get the error bound

$$\|G-\tilde{G}\|_\infty\leq\frac{1}{\lambda_1^2}\|C\|_2\|B\|_2\mathcal{O}(\|\Delta A\|_2). \tag{20}$$

Remark 4.3 If a finite element method is used for the spatial semi-discretization of a parabolic PDE, the corresponding differential equation looks as follows:

$$M\dot{x}(t) = -Sx(t) + \hat{B}u(t),$$

where M is the mass matrix and S the stiffness matrix. For a self-adjoint spatial differential operator, both are symmetric and positive (semi-)definite. With a Cholesky decomposition of $M = M_c M_c^T$, we obtain a symmetric, stable system matrix $A = -M_c^{-1} S M_c^{-T}$ if we multiply the state equation by M_c^{-1} from the left and define $\hat{x} := M_c^T x$, and a transformed state equation

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t),$$

with $B := M_c^{-1}\hat{B}$. For these systems with symmetric state matrix A and with corresponding $\mathcal{H}$-matrix approximation $A_{\mathcal{H}}$ the assumptions of Corollary 4.2 with $\tilde{A} = A_{\mathcal{H}}$ are fulfilled. □

Example 4.4 As an example assume that M, S are the mass and stiffness matrices associated to a finite-element approximation of a second-order elliptic operator with corresponding coercive, symmetric bilinear form and coercivity constant ρ on a bounded domain $\Omega \subset \mathbb{R}^2$, using a family of meshes with a certain regularity (see [4, Section 5.5]). We can thus order the eigenvalues of M and S as

$$0 < \lambda_1^S \leq \lambda_2^S \leq \ldots \leq \lambda_n^S \quad \text{and} \quad 0 < \lambda_1^M \leq \lambda_2^M \leq \ldots \leq \lambda_n^M, \quad \text{respectively.}$$

Then $A = -M_c^{-1} S M_c^{-T}$ is negative definite with eigenvalues λ_j as in (19). As $S - \lambda M$ is a symmetric-definite pencil (see, e.g., [22, Section 8.7] for properties of those), we have

$$-\lambda_1 = \min_{\|x\|_2=1} \frac{x^T S x}{x^T M x} \geq \frac{\min_{\|x\|_2=1} x^T S x}{\max_{\|x\|_2=1} x^T M x} = \frac{\lambda_1^S}{\lambda_n^M}.$$

Using the bound $\lambda_1^S \geq \rho \lambda_1^M$ for the minimal eigenvalue of S given in [4, Section 5.5], we get

$$-\lambda_1 \geq \frac{\rho \lambda_1^M}{\lambda_n^M} = \frac{\rho}{\operatorname{cond}_2(M)}.$$

Thus, for such problems, we obtain from (20)

$$\|G - \tilde{G}\|_\infty \leq \frac{\operatorname{cond}_2(M)^2}{\rho^2} \|C\|_2 \|B\|_2 \mathcal{O}(\|\Delta A\|_2).$$

According to [4, Section 5.5], the spectral condition number of M is uniformly bounded, i.e., there exists a constant c_M, independent of the mesh (here, represented by the dimension n of the finite-element ansatz space), so that $\operatorname{cond}_2(M) \leq c_M \cdot \frac{1}{n^2}$. Hence

$$\|G - \tilde{G}\|_\infty \leq \frac{c_M}{(\rho n)^2} \|C\|_2 \|B\|_2 \mathcal{O}(\|\Delta A\|_2)$$

for all meshes in the considered family. □

We can now state our main result of this section which combines the errors due to the $\mathcal{H}$-matrix approximation and balanced truncation.

Theorem 4.5 *With $\hat{G}$ as TFM associated to the reduced-order system (5) obtained by applying balanced truncation to $G_{\mathcal{H}}$ and the assumptions of Theorem 4.1 using $\tilde{A} = A_{\mathcal{H}}$, with $\mathcal{H}$-matrix approximation error*

$$\|A - A_{\mathcal{H}}\|_2 \leq c_{\mathcal{H}}\epsilon,$$

we obtain for the whole approximation error (15)

$$\|G-\hat{G}\|_\infty \leq \|C\|_2\|B\|_2 \operatorname{cond}_2(T) \operatorname{cond}_2(T_{\mathcal{H}}) \frac{1}{\min\limits_{i=1,\ldots,n} |Re(\lambda_i(A))|^2} \mathcal{O}(\epsilon) + 2\left(\sum_{j=r+1}^{n} \sigma_j\right).$$

The bound simplifies for the practical relevant case of symmetric, negative definite matrices A and $A + \Delta A$ with ordered real eigenvalues $\lambda_i \in \Lambda(A)$ as in (19) to

$$\begin{aligned} \|G - \hat{G}\|_\infty &\leq c_{\mathcal{H}}\epsilon\|C\|_2\|B\|_2 \max_{\lambda\in\Lambda(A)} \frac{1}{|\lambda|} \max_{\tilde{\lambda}\in\Lambda(A+\Delta A)} \frac{1}{|\tilde{\lambda}|} + 2\left(\sum_{j=r+1}^{n} \sigma_j\right) \\ &\leq \frac{1}{\lambda_1^2}\|C\|_2\|B\|_2\mathcal{O}(\epsilon) + 2\left(\sum_{j=r+1}^{n} \sigma_j\right). \end{aligned}$$

All error bounds derived in this section are of merely qualitative nature and suggest to choose the tolerance for the $\mathcal{H}$-matrix approximation small enough to compensate for possible error amplification due to eigenvalues close to the imaginary axis. In the next section, we will show the approximation errors $\|G_{\mathcal{H}} - \hat{G}\|_\infty$ which were not analyzed in this section. Thus, for reduced-order models to exhibit the accuracy displayed there, the $\mathcal{H}$-matrix approximation error discussed in this section needs to be of the same order as the errors $\|G_{\mathcal{H}} - \hat{G}\|_\infty$.

5 Numerical Experiments

Before we describe the exemplary systems on which we have tested the developed model reduction methods, we consider how to measure the accuracy of the resulting reduced-order system in practice. Note that we can only compute the second part in (15), i.e., $\|G_{\mathcal{H}} - \hat{G}\|_\infty$, of the approximation error between original and reduced-order system $G - \hat{G}$. This part were bounded by the usual error bound (6) if the reduced-order system were computed by exact balanced truncation. Using the $\mathcal{H}$-matrix format and the approximate arithmetic we compute approximations to the low-rank factors of the Gramians. Therefore, we introduce further errors, also in further computational steps based on these Gramians. Thus, the reduced-order model cannot be expected to fulfil the error bound for all examples, in particular if the underlying system involves an ill-conditioned matrix A. A complete analysis including all these error terms is beyond the scope of this paper and will be reported elsewhere.

5.1 Computing Frequency Response Errors in $\mathcal{H}$-Matrix Arithmetic

For computing a bound for the latter part $\|G_{\mathcal{H}} - \hat{G}\|_\infty$ of the error estimate (15) we have to note that the $\mathcal{H}$-matrix format is defined only for real-valued matrices. So we have to compute the frequency response of the complex transfer function $G_{\mathcal{H}}$ separately for the real and for the imaginary part.

We discuss the results of the model reduction methods by help of a Bode diagram which is often used in systems theory and signal processing to show the transfer function or frequency response of an LTI model. This model can be continuous or discrete, and single-input/single-output (SISO) or multi-input/multi-output (MIMO). We consider only one part of the diagram, the Bode magnitude plot, where the magnitude of the frequency response is plotted against the frequency. Typically, logarithmic scales are used for both axis to display a large range of values.

In the continuous-time setting, the frequency response of the error system $G(\jmath\omega) - \hat{G}(\jmath\omega)$ is evaluated at some fixed frequencies ω_k and used to quantify the error employing the spectral norm:

$$\begin{array}{rcll} \|G_{\mathcal{H}}(\jmath\omega_k) - \hat{G}(\jmath\omega_k)\|_2 & = & |G_{\mathcal{H}}(\jmath\omega_k) - \hat{G}(\jmath\omega_k)|, & \text{for SISO systems,} \\ \|G_{\mathcal{H}}(\jmath\omega_k) - \hat{G}(\jmath\omega_k)\|_2 & = & \sigma_{\max}(G_{\mathcal{H}}(\jmath\omega_k) - \hat{G}(\jmath\omega_k)), & \text{for MIMO systems.} \end{array}$$

For discrete-time systems the absolute error is computed as the maximum singular value of the error system $G_{\mathcal{H}}(z) - \hat{G}(z)$ for $z = e^{\jmath\omega_k T_s}$ and sampling time T_s at some fixed frequencies $\omega_k \in [0, 2\omega_N]$, where $\omega_N = \pi/T_s$ is the so-called *Nyquist frequency*,

$$\begin{array}{rcll} \|G_{\mathcal{H}}(e^{\jmath\omega_k T_s}) - \hat{G}(e^{\jmath\omega_k T_s})\|_2 & = & |G_{\mathcal{H}}(e^{\jmath\omega_k T_s}) - \hat{G}(e^{\jmath\omega_k T_s})|, & \text{for SISO systems,} \\ \|G_{\mathcal{H}}(e^{\jmath\omega_k T_s}) - \hat{G}(e^{\jmath\omega_k T_s})\|_2 & = & \sigma_{\max}(G_{\mathcal{H}}(e^{\jmath\omega_k T_s}) - \hat{G}(e^{\jmath\omega_k T_s})), & \text{for MIMO systems.} \end{array}$$

We will treat the continuous-time case in more detail. For the complex-valued matrix in the definition of the transfer function we consider a splitting into real part X_{Re} and imaginary part X_{Im}:

$$G_{\mathcal{H}}(\jmath\omega_k) = C(\jmath\omega_k I - A_{\mathcal{H}})^{-1}B + D = C(X_{\text{Re}} + \jmath\, X_{\text{Im}}) + D.$$

Since B is real-valued we obtain a system of equations for the unknowns X_{Re} and X_{Im}

$$\begin{aligned} -A_{\mathcal{H}}X_{\mathrm{Re}} - \omega_k X_{\mathrm{Im}} &= B, \\ \omega_k X_{\mathrm{Re}} - A_{\mathcal{H}}X_{\mathrm{Im}} &= 0, \end{aligned}$$

and by some simple calculations the following solution formulas:

$$\begin{aligned} X_{\mathrm{Re}} &= -(A_{\mathcal{H}}^2 + \omega_k^2 I)^{-1} A_{\mathcal{H}} B, \\ X_{\mathrm{Im}} &= -\omega_k (A_{\mathcal{H}}^2 + \omega_k^2 I)^{-1} B. \end{aligned}$$

The norm of the error system with formatted arithmetics can now be approximated as follows:

$$\begin{aligned} \|G_{\mathcal{H}}(\jmath\omega_k) - \hat{G}(\jmath\omega_k)\|_2 &= \sigma_{\max}(G_{\mathcal{H}}(\jmath\omega_k) - \hat{G}(\jmath\omega_k)) \\ &= \sigma_{\max}(C(X_{\mathrm{Re}} + \jmath\, X_{\mathrm{Im}}) - \hat{C}(\jmath\omega I - \hat{A})^{-1}\hat{B}) \\ &= \sigma_{\max}(C[-\operatorname{Inv}_{\mathcal{H}}(A_{\mathcal{H}} \odot A_{\mathcal{H}} \oplus \omega_k^2 I)A_{\mathcal{H}}B - \jmath\omega_k \operatorname{Inv}_{\mathcal{H}}(A_{\mathcal{H}} \odot A_{\mathcal{H}} \oplus \omega_k^2 I)B] \\ &\qquad -\hat{C}(\jmath\omega I - \hat{A})^{-1}\hat{B}). \end{aligned}$$

5.2 Results by Balanced Truncation and Singular Perturbation Approximation

All numerical experiments were performed on an SGI Altix 3700 (32 Itanium II processors, 1300 MHz, 64 GBytes shared memory, only one processor is used). We made use of the LAPACK and BLAS libraries for performing the standard dense matrix operations and include the routine DGEQPX of the RRQR library [16] for computing the RRQR factorization. For the $\mathcal{H}$-matrix approximation we employ HLib 1.3 [18] with adaptive rank choice (see [23]) instead of a given constant rank. The parameter ϵ which determines the desired accuracy in each matrix block is chosen in dependency on the RRQR parameter τ. As we choose $\tau = 10^{-8}$, the discussion at the end of Section 3.3 implies setting $\epsilon = 10^{-8}$, too. Accordingly, we choose $10^{-4} = \sqrt{\epsilon}$ as stopping criterion for the matrix equation solvers and perform two additional iteration steps, thereby exploiting the quadratic convergence rate of the sign or Smith iteration. For reducing the order n of the systems we apply the $\mathcal{H}$-matrix based model reduction methods where the reduced order is determined by the threshold $\mathrm{tol} = 10^{-4}$ for the approximation quality. We denote by δ the computable main part of the estimate of the global error bound (6),

$$\delta = 2 \sum_{j=r+1}^{\tilde{n}} \sigma_j.$$

Then, the reduced order r is chosen as minimal integer such that

$$2 \sum_{j=r+1}^{\tilde{n}} \sigma_j \leq \mathrm{tol}.$$

For determining the numerical McMillan degree of the LTI system in the approximate SPA method we use a threshold of size 10^{-14}.

Example 5.1 As a first example we consider the two-dimensional heat equation in the unit square $\Omega = (0,1)^2$ with constant heat source in some subdomain Ω_u as described in [26]:

$$\begin{aligned} \frac{\partial \mathbf{x}}{\partial t}(t,\xi) &= \nabla(k(\xi)\cdot\nabla\mathbf{x}(t,\xi)) + b(\xi)u(t), \qquad \xi\in\Omega,\, t\in(0,\infty), \qquad (21)\\ b(\xi) &= \begin{cases} 1, & \xi\in\Omega_u,\\ 0, & \text{otherwise.}\end{cases} \end{aligned}$$

The diffusion coefficient k is a material-specific quantity depending on the heat conductivity, the density and the heat capacity. In this example we choose the diffusion constant as $k(\xi)\equiv 1.0$. We impose homogeneous Dirichlet boundary conditions

$$\mathbf{x}(t,\xi) = 0, \quad \xi\in\partial\Omega,$$

and allow the measurement of the temperature in a small subdomain Ω_o

$$y(t,\xi) = \mathbf{x}(t,\xi)_{|\Omega_o}.$$

We discretize the heat equation (21) with linear finite elements and n inner grid points ξ_i. In the weak form of the partial differential equation we use a classical Galerkin approach with bilinear finite element ansatz functions φ_i: $\mathbf{x}(t,\xi)\approx\sum_{i=1}^n x_i(t)\varphi_i(\xi)$. For the n unknowns x_i we obtain a system of linear differential equations

$$E\dot{x}(t) = -\tilde{A}x(t) + \tilde{B}u(t) \qquad (22)$$

with matrices E, $\tilde{A}$, $\tilde{B}$ defined by the entries

$$\begin{aligned} E_{ij} &= \int_\Omega \varphi_i(\xi)\varphi_j(\xi)\,d\xi,\\ \tilde{A}_{ij} &= \int_\Omega k(\xi)\,\langle\nabla\varphi_i(\xi),\nabla\varphi_j(\xi)\rangle\,d\xi, \qquad (23)\\ \tilde{B}_{i1} &= \int_\Omega b(\xi)\varphi_i(\xi)\,d\xi, \quad \text{for } i,j=1,\ldots,n. \end{aligned}$$

The additional output equation is given as

$$y(t) = Cx(t),$$

where

$$C_{1j} = \begin{cases} 1, & \xi_j\in\Omega_o,\\ 0, & \text{otherwise,}\end{cases} \qquad \text{for } j=1,\ldots,n.$$

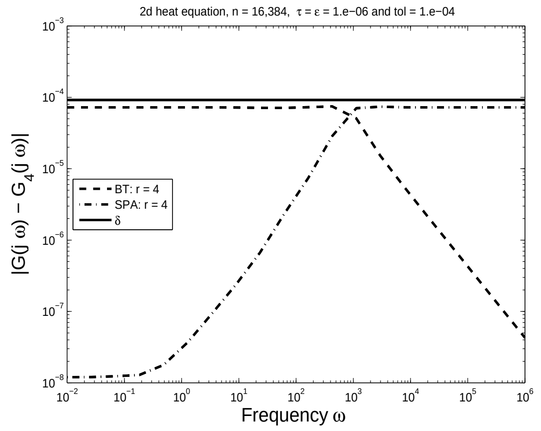

Figure 1: Absolute errors in Example 5.1.

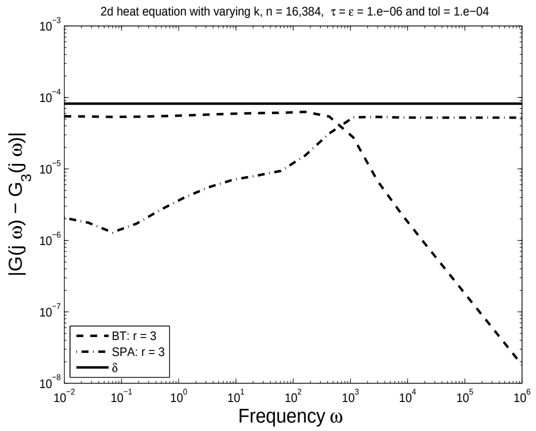

Figure 2: Absolute errors in Example 5.2.

The order of the finite element ansatz space is chosen as $n = 16,384$. We approximate the $n \times n$ mass matrix E in $\mathcal{H}$-matrix format and invert it using a formatted LU decomposition. The resulting state matrix $A = -E^{-1}\tilde{A}$ is also stored as $\mathcal{H}$-matrix. Thus, we have a large-scale stable LTI system as introduced in (1) with $B = E^{-1}\tilde{B}$, $C^T \in \mathbb{R}^{n\times 1}$ (SISO). With the given approximation error threshold of tol $= 10^{-4}$, the reduced order is determined as $r = 4$ and the approximate error bound is computed to be $\delta = 9.18 \times 10^{-5}$. The frequency response errors for the $\mathcal{H}$-matrix based BT and SPA method are shown in Figure 1. The errors are computed as described in Section 5.1 as the pointwise absolute values of the error system at 20 fixed frequencies ω_k from $10^{-2}, \ldots, 10^{6}$ in logarithmic scale by use of the formatted $\mathcal{H}$-matrix arithmetic. We observe good matching at high frequencies for the BT model while the SPA model has good approximation at low frequencies as expected. □

Example 5.2 In this example we use the same FEM discretization of the heat equation (21) as in Example 5.1. Instead of a constant choice of the diffusion coefficient $k(\xi)$ we vary $k(\xi)$ over the domain similar to [26]:

$$k(\xi) = \begin{cases} 10, & \xi \in [-1,1] \times [-\frac{1}{3}, \frac{1}{3}], \\ 10^{-4}, & \xi \in [-\frac{1}{3}, \frac{1}{3}] \times \left([-1, -\frac{1}{3}) \cup (\frac{1}{3}, 1]\right), \\ 1.0, & \text{otherwise.} \end{cases}$$

Here, the reduced order is determined as $r = 3$. The frequency response errors for BT and SPA reduced-order models are compared in Figure 2. As in the previous example we observe a typical good approximation of the BT method for larger frequencies and of the SPA reduced systems for frequencies close to zero. The results fulfil the approximate BT error bound $\delta = 8.2 \times 10^{-5}$. □

Example 5.3 Next we include a constant convective term in (21). Thus, we consider systems with nonsymmetric stiffness matrix $\tilde{A}$ resulting from the convection-diffusion equation

$$\frac{\partial \mathbf{x}}{\partial t}(t,\xi) = k\,\Delta \mathbf{x}(t,\xi) + c \cdot \nabla \mathbf{x}(t,\xi) + b(\xi)u(t), \qquad \xi \in \Omega,\ t \in (0,\infty),$$

with a constant diffusion coefficient $k(\xi) \equiv 10^{-4}$ and a fixed choice of the convection vector $c = (0,1)^T$. The left plot of Figure 3 reports the absolute errors of the transfer functions for the original system and the reduced-order models computed by BT and SPA methods. By a choice of $\text{tol} = 10^{-4}$ a reduced order of $r = 11$ is determined and the error bound estimate is computed as $\delta = 7.74 \times 10^{-5}$. It is seen that the reduced systems satisfy the error estimate. To examine the influence of the parameter setting, BT results for different choices of τ and ϵ are depicted in the right plot. As analyzed in Section 3.3, choosing $\tau \gg \epsilon$ influences the accuracy of the reduced-order model: combining $\tau = 10^{-4}$ with $\epsilon = 10^{-6}$ or $\epsilon = 10^{-8}$, the error is clearly dominated by the value of τ. In this example, reduced-order models which satisfy the given error bound can only be obtained by choosing $\epsilon \leq \tau \ll \text{tol}$. For $\tau = \epsilon = \text{tol}$ (in the presented example, all values are 10^{-4}), the accumulated errors obtained from using $\mathcal{H}$-matrix arithmetic and the resulting approximate Gramians are obviously larger than the required tolerance. Also note that in this example, the condition number of T is much larger than 1 so that a significant error amplification can be expected, see the discussion at the end of Section 3.3.

This example confirms that for model reduction purposes, the relation of τ to tol is the main critical issue. As ϵ should be chosen as large as possible to minimize workspace requirements and computing time, this confirms the sensible choice $\tau = \epsilon$. □

Example 5.4 We consider a time discretization of the instationary heat equation (21) with time step size $T_s = 10^{-4}$. Using the FEM space discretization as introduced in Example 5.1 (setting $k(\xi) \equiv 1.0$) and an backward Euler scheme we obtain a discrete time-invariant system

$$\begin{aligned} x_{j+1} &= \underbrace{(E - T_sA)^{-1}E}_{A_d}\, x_j + \underbrace{T_s(E - T_sA)^{-1}B}_{B_d}\, u_j, \\ y_j &= C_d x_j, \qquad \text{for } j = 0,1,2,\ldots, \end{aligned}$$

with stable state matrix $A_d \in \mathbb{R}^{n \times n}$ and B_d, $C_d^T \in \mathbb{R}^{n \times 1}$. The order of the system is chosen as $n = 16,384$. We compute the absolute error at 30 frequencies ω_k in logarithmic scale with $\omega_k \in [0, 2\omega_N]$. The absolute errors of the transfer functions for the original system and the reduced-order models computed by BT and SPA methods are shown in Figure 4. We again observe a good matching at high

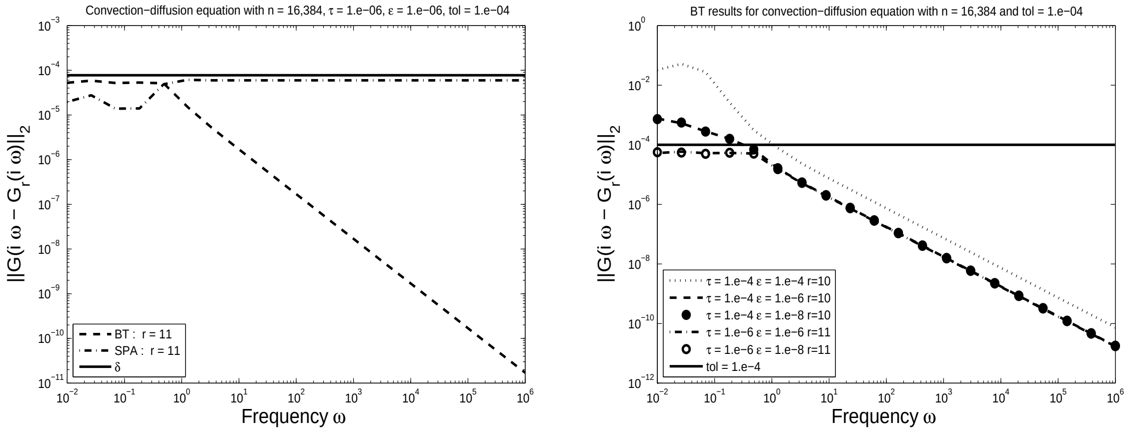

Figure 3: Frequency response errors for BT and SPA reduced-order models in Example 5.3.

frequencies for the BT model of size $r = 4$ with an estimate $\delta = 8.39 \times 10^{-5}$ for the error. The reduced-order models computed by SPA have good approximation at low frequencies. □

Example 5.5 In this example we consider a finite element discretization of a boundary integral equation for solving the Laplace equation in $\Omega \subset \mathbb{R}^3$. Using the Ritz-Galerkin method with n piecewise constant ansatz functions $\{\varphi_1, \dots, \varphi_n\}$ we obtain the following entries of the stiffness matrix

$$A_{ij} = \int_\Gamma \varphi_i(y) \int_\Gamma \frac{1}{4\pi} \frac{1}{|x-y|} \varphi_j(x) \, d\Gamma_x d\Gamma_y$$

for $i, j = 1, \dots, n$, where $|\,.\,|$ denotes the Euclidean norm, see [29] for details. To construct a dynamical system we introduce an artifical time dependence. By use of the stiffness matrix A, taking $B, C^T \in \mathbb{R}^{n \times 1}$ as introduced in Example 5.1, we obtain a stable LTI system

$$\begin{aligned} \dot{x}(t) &= -Ax(t) + Bu(t), \\ y(t) &= Cx(t). \end{aligned}$$

We choose Ω as a three-dimensional sphere and compute the entries in the low-rank blocks of the $\mathcal{H}$-matrix using adaptive cross approximation [17] with a block-wise accuracy of $\epsilon = 10^{-8}$. By a problem size of $n = 32,768$ the frequency response errors for BT and SPA reduced models are depicted in Figure 5. For tol $= 10^{-4}$ we determine the order $r = 10$ and the approximate error bound $\delta = 4.82 \times 10^{-5}$. We observe a good approximation quality of the BT and SPA

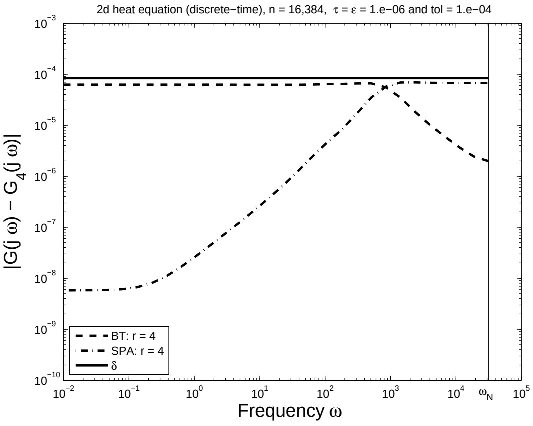

Figure 4: Absolute errors in Example 5.4.

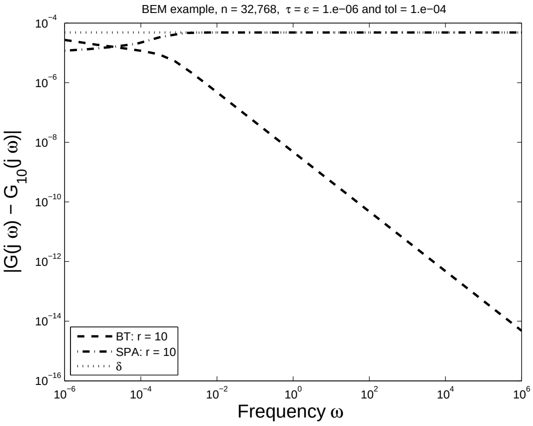

Figure 5: Absolute errors in Example 5.5.

reduced-order systems. Again, we observe the usual small error of BT at large and for SPA at low frequencies.

We would like to emphasize that in this example, we have computed very small reduced-order models ($r = 10$) for a fairly large LTI system ($n = 32,768$). In particular, here A is a *dense* $32,768 \times 32,768$ matrix. This becomes only possible through the combination of $\mathcal{H}$-matrix approximation and model reduction. □

6 Conclusions

We have shown that balanced truncation can be used for model reduction of large-scale linear systems resulting from (semi-)discretizations of parabolic control systems when the state matrix may be dense, but has a data-sparse representation. Employing formatted arithmetic in sign function-based Lyapunov solvers, the resulting implementations of balanced truncation and singular perturbation approximation have linear-polylogarithmic complexity. The approximation quality is critical with respect to the several parameters that have to be chosen in the computations. The usual error bound obtained in balanced truncation can here only serve as an estimate. If used for determining the size of the reduced-order model based on a given tolerance threshold, the parameters determining the accuracy in the formatted arithmetic and the approximation quality of the low-rank factors of the system Gramians need to be chosen with care. A rough error analysis confirmed by the numerical experiments indicates that the quality of the reduced-order model is essentially determined by the accuracy of the low-rank factors of the Gramians as long as the $\mathcal{H}$-matrix approximation error is of the same order. The numerical examples discussed in this paper should give a good indication for reasonable parameter combinations.

Acknowledgements

This work was supported by the DFG Research Center "Mathematics for key technologies" and DFG grant BE 2174/7-1, *Automatic, Parameter-Preserving Model Reduction for Applications in Microsystems Technology*.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide*, SIAM, Philadelphia, PA, third ed., 1999.

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

[3] A. Antoulas, D. Sorensen, and Y. Zhou, *On the decay rate of Hankel singular values and related issues*, Sys. Control Lett., 46 (2002), pp. 323–342.

[4] O. Axelsson and V. Barker, *Finite Element Solution of Boundary Value Problems*, SIAM Publications, Philadelphia, PA, 2001. Originally published by Academic Press, Orlando, Fl, 1984.

[5] U. Baur, *Low Rank Solution of Data-Sparse Sylvester Equations*, Preprint #266, MATHEON, DFG Research Center "Mathematics for Key Technologies", Berlin, FRG, `http://www.math.tu-berlin.de/DFG-Forschungszentrum`, Oct. 2005. To appear in Numer. Lin. Alg. Appl.

[6] U. Baur and P. Benner, *Factorized solution of Lyapunov equations based on hierarchical matrix arithmetic*, Computing, 78 (2006), pp. 211–234.

[7] M. Bebendorf and W. Hackbusch, *Existence of $\mathcal{H}$-matrix approximants to the inverse FE-matrix of elliptic operators with L^∞-coefficients*, Numer. Math., 95 (2003), pp. 1–28.

[8] P. Benner, J. Claver, and E. Quintana-Ortí, *Efficient solution of coupled Lyapunov equations via matrix sign function iteration*, in Proc. 3rd Portuguese Conf. on Automatic Control CONTROLO'98, Coimbra, A. D. et al., ed., 1998, pp. 205–210.

[9] P. Benner, V. Mehrmann, and 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.

[10] P. Benner and E. Quintana-Ortí, *Model reduction based on spectral projection methods.* Chapter 1 (pages 5–48) of [9].

[11] ——, *Solving stable generalized Lyapunov equations with the matrix sign function*, Numer. Algorithms, 20 (1999), pp. 75–100.

[12] P. Benner, E. Quintana-Ortí, and G. Quintana-Ortí, *Solving stable Stein equations on distributed memory computers*, in EuroPar'99 Parallel Processing, P. Amestoy, P. Berger, M. Daydé, I. Duff, V. Frayssé, L. Giraud, and D. Ruiz, eds., no. 1685 in Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, New York, 1999, pp. 1120–1123.

[13] ——, *Balanced truncation model reduction of large-scale dense systems on parallel computers*, Math. Comput. Model. Dyn. Syst., 6 (2000), pp. 383–405.

[14] ——, *Numerical solution of discrete stable linear matrix equations on multicomputers*, Parallel Algorithms and Appl., 17 (2002), pp. 127–146.

[15] ——, *Parallel algorithms for model reduction of discrete-time systems*, Int. J. Syst. Sci., 34 (2003), pp. 319–333.

[16] C. Bischof and G. Quintana-Ortí, *Algorithm 782: codes for rank-revealing QR factorizations of dense matrices*, ACM Trans. Math. Software, 24 (1998), pp. 254–257.

[17] S. Börm and L. Grasedyck, *Hybrid cross approximation of integral operators*, Numer. Math., 101 (2005), pp. 221–249.

[18] S. Börm, L. Grasedyck, and W. Hackbusch, *HLib 1.3*, 2004. Available from `http://www.hlib.org`.

[19] Y. Chahlaoui and P. Van Dooren, *A collection of benchmark examples for model reduction of linear time invariant dynamical systems*, SLICOT Working Note 2002–2, Feb. 2002. Available from `www.slicot.org`.

[20] B. Datta, *Numerical Methods for Linear Control Systems*, Elsevier Academic Press, 2004.

[21] K. Glover, *All optimal Hankel-norm approximations of linear multivariable systems and their L^∞ norms*, Internat. J. Control, 39 (1984), pp. 1115–1193.

[22] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, third ed., 1996.

[23] L. Grasedyck, *Theorie und Anwendungen Hierarchischer Matrizen*, Dissertation, University of Kiel, Kiel, Germany, 2001. In German, available at `http://e-diss.uni-kiel.de/diss_454`.

[24] ——, *Existence of a low rank or $\mathcal{H}$-matrix approximant to the solution of a Sylvester equation*, Numer. Lin. Alg. Appl., 11 (2004), pp. 371–389.

[25] L. Grasedyck and W. Hackbusch, *Construction and arithmetics of $\mathcal{H}$-matrices*, Computing, 70 (2003), pp. 295–334.

[26] L. Grasedyck, W. Hackbusch, and B. N. Khoromskij, *Solution of large scale algebraic matrix Riccati equations by use of hierarchical matrices*, Computing, 70 (2003), pp. 121–165.

[27] S. Gugercin and J.-R. Li, *Smith-type methods for balanced truncation of large systems.* Chapter 2 (pages 49–82) of [9].

[28] S. Gugercin, D. Sorensen, and A. Antoulas, *A modified low-rank Smith method for large-scale Lyapunov equations*, Numer. Algorithms, 32 (2003), pp. 27–55.

[29] W. Hackbusch, *Integral equations. Theory and numerical treatment.*, ISNM. International Series of Numerical Mathematics. 120. Basel: Birkhäuser. xiv, 359 p., 1995.

[30] ——, *A sparse matrix arithmetic based on $\mathcal{H}$-matrices. I. Introduction to $\mathcal{H}$-matrices*, Computing, 62 (1999), pp. 89–108.

[31] W. Hackbusch and B. N. Khoromskij, *A sparse $\mathcal{H}$-matrix arithmetic. II. Application to multi-dimensional problems*, Computing, 64 (2000), pp. 21–47.

[32] A. Laub, M. Heath, C. Paige, and R. Ward, *Computation of system balancing transformations and other application of simultaneous diagonalization algorithms*, IEEE Trans. Automat. Control, 34 (1987), pp. 115–122.

[33] C. Lawson, R. Hanson, D. Kincaid, and F. Krogh, *Basic linear algebra subprograms for FORTRAN usage*, ACM Trans. Math. Software, 5 (1979), pp. 303–323.

[34] W. Levine, ed., *The Control Handbook*, CRC Press, 1996.

[35] Y. Liu and B. Anderson, *Controller reduction via stable factorization and balancing*, Internat. J. Control, 44 (1986), pp. 507–531.

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

[37] G. Obinata and B. Anderson, *Model Reduction for Control System Design*, Communications and Control Engineering Series, Springer-Verlag, London, UK, 2001.

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

[39] J. D. Roberts, *Linear model reduction and solution of the algebraic Riccati equation by use of the sign function*, Internat. J. Control, 32 (1980), pp. 677–687. (Reprint of Technical Report No. TR-13, CUED/B-Control, Cambridge University, Engineering Department, 1971).

[40] S. A. Sauter and C. Schwab, *Randelementmethoden*, B. G. Teubner, Stuttgart, Leipzig, Wiesbaden, 2004.

[41] R. Schneider, *Multiskalen- und Wavelet-Matrixkompression: Analysisbasierte Methoden zur effizienten Lösung großer vollbesetzter Gleichungssysteme*, B. G. Teubner, Stuttgart, 1998.

[42] V. Sima, *Algorithms for Linear-Quadratic Optimization*, vol. 200 of Pure and Applied Mathematics, Marcel Dekker, Inc., New York, NY, 1996.

[43] R. Smith, *Matrix equation $XA+BX = C$*, SIAM J. Appl. Math., 16 (1968), pp. 198–201.

[44] V. Sokolov, *Contributions to the Minimal Realization Problem for Descriptor Systems*, Dissertation, Fakultät für Mathematik, TU Chemnitz, 09107 Chemnitz (Germany), Jan. 2006.

[45] M. Tombs and I. Postlethwaite, *Truncated balanced realization of a stable non-minimal state-space system*, Internat. J. Control, 46 (1987), pp. 1319–1330.

[46] P. Van Dooren, *Gramian based model reduction of large-scale dynamical systems*, in Numerical Analysis 1999. Proc. 18th Dundee Biennial Conference on Numerical Analysis, D. Griffiths and G. Watson, eds., London, UK, 2000, Chapman & Hall/CRC, pp. 231–247.

[47] A. Varga, *Efficient minimal realization procedure based on balancing*, in Prepr. of the IMACS Symp. on Modelling and Control of Technological Systems, vol. 2, 1991, pp. 42–47.

[48] K. Zhou, J. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice-Hall, Upper Saddle River, NJ, 1996.

Chemnitz Scientific Computing Preprints – ISSN 1864-0087