Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 51-2 (2018) 547–552

Model Reduction of Descriptor Systems with the MORLAB Toolbox ⋆

Peter Benner * **Steffen W. R. Werner** **

* *Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany (e-mail: benner@mpi-magdeburg.mpg.de)*
** *Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany (e-mail: werner@mpi-magdeburg.mpg.de)*

Abstract: The MORLAB (Model Order Reduction Laboratory) toolbox is a software solution in MATLAB for the model reduction of linear time-invariant continuous-time systems. It contains several methods to deal with unstructured medium-scale descriptor systems. In order to illustrate the use of MORLAB in this context, in this paper, a damped mass-spring system with a holonomic constraint is reduced by the generalized balanced truncation subroutine from the MORLAB toolbox. The basic ideas of the MORLAB implementation are shown by introducing the used spectral projection methods based on the matrix sign and matrix disk function. To show the numerical behavior, the MORLAB routine and all the corresponding subroutines are used for computations applied to the introduced example. An overview about further model reduction methods for descriptor systems in MORLAB is given, on which similar implementational ideas as for the generalized balanced truncation method are used.

Keywords: model reduction, linear systems, descriptor systems, spectral projection methods, matrix equations, mechanical systems, numerical methods

1. INTRODUCTION

Many different real-world applications, like chemical processes, electrical circuits, or computational fluid dynamics, can be modeled by systems of differential equations. Since experiments can be very expensive and time-consuming, these models are used for simulations and the design of controllers. The modeling processes often result in linear time-invariant continuous-time descriptor systems

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

with $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$. Here, $u(t) \in \mathbb{R}^m$ are the inputs of the system which influence the internal states $x(t) \in \mathbb{R}^n$ to get the desired outputs $y(t) \in \mathbb{R}^p$. Throughout this paper it is assumed that the matrix pencil $\lambda E - A$ is regular, i.e., there exists at least one $\lambda \in \mathbb{C}$ such that $\det(\lambda E - A) \neq 0$. Then with the initial condition $Ex(0) = 0$, the input-output behavior of (1) in the frequency domain can be described by the system's transfer function

$$G(s) = C(sE - A)^{-1}B + D, \tag{2}$$

$s \in \mathbb{C}$, where the matrices E, A, B, C, D from (1) define a realization of G. Usually, the number of inputs and outputs is small compared to the number of internal states of (1). For this reason, using the full-order model (1) quickly reaches the limits of computational resources. The aim of model reduction is now to compute a surrogate model for (1) of order $r \ll n$ such that

$$\begin{aligned} E_r\dot{x}_r(t) &= A_r x_r(t) + B_r u(t),\\ y_r(t) &= C_r x_r(t) + D_r u(t) \end{aligned} \tag{3}$$

approximates the input-output behavior of (1) in a given system norm. Now, the reduced-order model (3) can be used instead of the original model for simulations and the design of controllers.

For the construction of (3) there are many different techniques, for some overviews see, e.g., Antoulas (2005), Benner et al. (2005) and Baur et al. (2014). These techniques are often implemented in several different versions. An overview about used model reduction software was set up by the MORwiki Community (2017). A MATLAB toolbox for model reduction of medium-scale unstructured dynamical systems of the form (1) is the MORLAB, Model Order Reduction Laboratory, toolbox; see Benner and Werner (2017); Benner (2006). The toolbox comes with appropriate solvers for the occurring matrix equations and methods for the decomposition of transfer functions. These algorithms are based on spectral projection methods like the matrix sign and matrix disk function; see, for example, (Benner et al., 2005, Chapter 1) and Benner (2011).

In this paper, the functionality of MORLAB subroutines for descriptor systems is illustrated reducing a benchmark example, introduced in Section 2. For the reduction of the example system, the generalized balanced truncation (GBT) method will be used as shown in Section 3, reformulated in the version from the MORLAB toolbox. Also, the used spectral projection methods are presented in this section. The computational results can be found in Section 4. In Section 5, a short overview about further model reduction routines for descriptor systems in MORLAB is

⋆ This work was supported by the DFG priority program 1897: "Calm, Smooth and Smart – Novel Approaches for Influencing Vibrations by Means of Deliberately Introduced Dissipation" and the DFG - 314838170, GRK 2297 MathCoRe.

10.1016/j.ifacol.2018.03.092

given. Finally, the conclusions of this paper can be found in Section 6.

2. THE BENCHMARK EXAMPLE

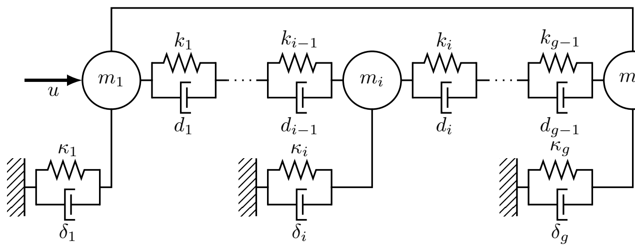

Fig. 1. A damped mass-spring system with a holonomic constraint.

As benchmark example for a descriptor system, a damped mass-spring system with a holonomic constraint is considered. Therefore, the i-th mass is connected to the $(i+1)$-st mass by a spring and a damper with the coefficients k_i and d_i. Also, the i-th mass is connected to an additional spring and damper with coefficients κ_i and δ_i. Moreover, the first mass is connected to the last one by a rigid bar. This system, as shown in Figure 1, can be found in (Benner et al., 2005, Chapter 3).

The vibrations of the resulting system are described by the following second-order equations

$$\begin{aligned} \mathcal{M}\ddot{p}(t) &= \mathcal{K}p(t) + \mathcal{D}\dot{p}(t) - \mathcal{L}^T\lambda(t) + \mathcal{B}_u u(t), \\ 0 &= \mathcal{L}p(t), \\ y(t) &= \mathcal{C}_p p(t), \end{aligned} \tag{4}$$

where $p(t)$ is the position vector, $\lambda(t) \in \mathbb{R}$ is the Lagrange multiplier, $\mathcal{M} = \operatorname{diag}(m_1, \ldots, m_g)$ is the mass matrix, $\mathcal{K}, \mathcal{D} \in \mathbb{R}^{g\times g}$ are the tridiagonal stiffness and damping matrices and $\mathcal{L} = [1, 0, \ldots, 0, -1]$ is the constraint matrix. According to Figure 1, $\mathcal{B}_u = e_1$ is the input matrix and three positions of masses are measured by $\mathcal{C}_p = [e_1,\ e_2,\ e_{g-1}]^T$, where e_i is the i-th column of I_g.

The system (4) is given in second-order form. For applying the routines of the MORLAB toolbox, the system has to be rewritten in first-order form. Therefore, the velocity vector $v(t) = \dot{p}(t)$ is introduced and all states are collected in $x(t) = \left[p(t)^T, v(t)^T, \lambda(t)\right]^T$, such that the system (4) can be rewritten in the form

$$\begin{aligned} \underbrace{\begin{bmatrix} I_g & 0 & 0 \\ 0 & \mathcal{M} & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{E} \dot{x}(t) &= \underbrace{\begin{bmatrix} 0 & I_g & 0 \\ \mathcal{K} & \mathcal{D} & -\mathcal{L}^T \\ \mathcal{L} & 0 & 0 \end{bmatrix}}_{A} x(t) + \underbrace{\begin{bmatrix} 0 \\ \mathcal{B}_u \\ 0 \end{bmatrix}}_{B} u(t), \\ y(t) &= \underbrace{\begin{bmatrix} \mathcal{C}_p & 0 & 0 \end{bmatrix}}_{C} x(t). \end{aligned} \tag{5}$$

This first-order formulation is an index-3 descriptor system. Furthermore, the system parameters are set like in (Benner et al., 2005, Chapter 3), which means the masses are all taken as $m_1 = \ldots = m_g = 100$ and the spring and damper coefficients are set to

$$\begin{aligned} k_1 = \ldots = k_{g-1} = \kappa_2 = \ldots = \kappa_{g-1} = 2, \quad \kappa_1 = \kappa_g = 4, \\ d_1 = \ldots = d_{g-1} = \delta_2 = \ldots = \delta_{g-1} = 5, \quad \delta_1 = \delta_g = 10. \end{aligned}$$

By construction, the system (5) is c-stable, i.e., the matrix pencil $\lambda E - A$ is regular and Hurwitz. Most of the published software like the model reduction routines in the MATLAB Control System Toolbox® or the SLICOT Model and Controller Reduction Toolbox, see Benner et al. (2010), is not applicable for such systems due to the problem of the singular E matrix. MORLAB provides appropriate algorithms to reduce such descriptor systems. Also, the DSTOOLS provide a generalized balanced truncation method, that is currently only applicable for non-singular E matrices; see Varga (2016).

3. MODEL REDUCTION IN MORLAB

3.1 The Generalized Balanced Truncation

For the reduction of the system (5), the implementation of the generalized balanced truncation method will be used. This method is the generalization of the well-known balanced truncation method to the case of descriptor systems (1) with singular E matrices. For a detailed derivation of the theory, the corresponding theorems and the method itself, see Stykel (2004) or (Benner et al., 2005, Chapter 3).

For asymptotically stable systems, the proper controllability and observability Gramians $\mathcal{G}_{pc}$ and $\mathcal{G}_{po}$ are defined as the unique solutions of the projected generalized continuous-time Lyapunov equations

$$\begin{aligned} E\mathcal{G}_{pc}A^T + A\mathcal{G}_{pc}E^T + P_\ell BB^T P_\ell^T &= 0, \\ P_r\mathcal{G}_{pc}P_r^T &= \mathcal{G}_{pc}, \end{aligned} \tag{6}$$

$$\begin{aligned} E^T\mathcal{G}_{po}A + A^T\mathcal{G}_{po}E + P_r^T C^T C P_r &= 0, \\ P_\ell^T\mathcal{G}_{po}P_\ell &= \mathcal{G}_{po}, \end{aligned} \tag{7}$$

with P_ℓ and P_r the left and right spectral projectors onto the deflating subspace corresponding to the finite eigenvalues of the matrix pencil $\lambda E - A$. The square roots of the eigenvalues of the matrix product $\mathcal{G}_{pc}E^T\mathcal{G}_{po}E$ are the proper Hankel singular values of the system, which are a measure for the controllability and observability of the corresponding states. Zero and small proper Hankel singular values are truncated in the generalized balanced truncation to get the reduced-order model. For descriptor systems, also improper controllability and observability Gramians $\mathcal{G}_{ic}$ and $\mathcal{G}_{io}$ are defined. These are the unique solutions of the projected generalized discrete-time Lyapunov equations

$$\begin{aligned} A\mathcal{G}_{ic}A^T - E\mathcal{G}_{ic}E^T - Q_\ell BB^T Q_\ell^T &= 0, \\ Q_r\mathcal{G}_{ic}Q_r^T &= \mathcal{G}_{ic}, \end{aligned} \tag{8}$$

$$\begin{aligned} A^T\mathcal{G}_{io}A - E^T\mathcal{G}_{io}E - Q_r^T C^T C Q_r &= 0, \\ Q_\ell^T\mathcal{G}_{io}Q_\ell &= \mathcal{G}_{io}, \end{aligned} \tag{9}$$

with Q_ℓ and Q_r the left and right spectral projectors onto the deflating subspace corresponding to the infinite eigenvalues of the matrix pencil $\lambda E - A$. The square roots of the eigenvalues of the matrix product $\mathcal{G}_{ic}A^T\mathcal{G}_{io}A$ are the improper Hankel singular values. Zero improper Hankel singular values correspond to unnecessary algebraic constraints, which can be truncated.

For asymptotically stable systems, all the Gramians are positive semi-definite. Therefore, they can be rewritten in factorized form $\mathcal{G}_{pc} = R_pR_p^T$, $\mathcal{G}_{po} = L_pL_p^T$, $\mathcal{G}_{ic} = R_iR_i^T$ and $\mathcal{G}_{io} = L_iL_i^T$. Consider the skinny singular value decompositions (SVDs)

$$L_p^T E R_p = [U_1\ U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix},$$
$$L_i^T A R_i = U_3 \Theta V_3^T,$$

where Σ_1, Σ_2 are diagonal matrices with the non-zero proper Hankel singular values and Θ is diagonal with the non-zero improper Hankel singular values. The partitioning in the first SVD results from the choice of the small Hankel singular values, which will be truncated. In the square root method, the following projection matrices

$$W = \left[L_p U_1 \Sigma_1^{-\frac{1}{2}},\ L_i U_3 \Theta^{-\frac{1}{2}}\right],$$
$$T = \left[R_p V_1 \Sigma_1^{-\frac{1}{2}},\ R_i V_3 \Theta^{-\frac{1}{2}}\right]$$

are used, such that the realization of the reduced-order model (3) is computed as

$$(E_r, A_r, B_r, C_r, D_r) = \left(W^T E T, W^T A T, W^T B, C T, D\right)$$
$$= \left(\begin{bmatrix} E_{r_f} & 0 \\ 0 & E_{r_\infty} \end{bmatrix}, \begin{bmatrix} A_{r_f} & 0 \\ 0 & A_{r_\infty} \end{bmatrix}, \begin{bmatrix} B_{r_f} \\ B_{r_\infty} \end{bmatrix}, \begin{bmatrix} C_{r_f} & C_{r_\infty} \end{bmatrix}, D \right).$$

The idea in the MORLAB implementation is to exploit the resulting block structure of this system to avoid the explicit computation of the spectral projectors for solving the projected Lyapunov equations (6)–(9). This is done by decoupling the system (1) into its slow subsystem

$$\begin{aligned} E_f \dot{x}_f(t) &= A_f x_f(t) + B_f u(t), \\ y_f(t) &= C_f(t), \end{aligned} \tag{10}$$

where $\lambda E_f - A_f$ contains all the finite eigenvalues of the original pencil $\lambda E - A$, and its fast subsystem

$$\begin{aligned} E_\infty \dot{x}_\infty(t) &= A_\infty x_\infty(t) + B_\infty u(t), \\ y_\infty(t) &= C_\infty(t) + D u(t), \end{aligned} \tag{11}$$

where $\lambda E_\infty - A_\infty$ has only infinite eigenvalues. In this case, the projected Lyapunov equations (6)–(9) simplify on the subsystems. For the proper part containing the potentially non-zero proper Hankel singular values, only the generalized continuous-time Lyapunov equations

$$E_f X_{pc} A_f^T + A_f X_{pc} E_f^T + B_f B_f^T = 0, \tag{12}$$
$$E_f^T X_{po} A_f + A_f^T X_{po} E_f + C_f^T C_f = 0, \tag{13}$$

with a non-singular E_f matrix, have to be solved. The solutions X_{pc} and X_{po} correspond to the proper controllability and observability Gramians, respectively. The same simplification also holds for the discrete-time Lyapunov equations

$$A_\infty X_{ic} A_\infty^T - E_\infty X_{ic} E_\infty^T - B_\infty B_\infty^T = 0, \tag{14}$$
$$A_\infty^T X_{io} A_\infty - E_\infty^T X_{io} E_\infty - C_\infty^T C_\infty = 0, \tag{15}$$

with a nilpotent E_∞, where X_{pc} and X_{po} correspond to the improper controllability and observability Gramians, respectively. The reconstruction of the complete system Gramians from the solutions of the unprojected equations is shown in (Stykel, 2002, Chapter 5).

3.2 Decoupling of Subsystems

The decoupling into the slow and fast subsystems is equivalent to an additive decomposition of the transfer function (2) as

$$G(s) = G_{sp}(s) + P(s),$$

with G_{sp} the strictly proper part, which corresponds to the slow subsystem (10), and P the polynomial part, corresponding to the fast subsystem (11). This decomposition is done by a block diagonalization of the matrix pencil $\lambda E - A$ and the partitioning of the other system matrices accordingly, such that

$$\begin{aligned} G(s) &= C(sE - A)^{-1}B + D \\ &= [C_f,\ C_\infty] \left(s \begin{bmatrix} E_f & 0 \\ 0 & E_\infty \end{bmatrix} - \begin{bmatrix} A_f & 0 \\ 0 & A_\infty \end{bmatrix} \right)^{-1} \begin{bmatrix} B_f \\ B_\infty \end{bmatrix} + D \\ &= \underbrace{C_f(sE_f - A_f)^{-1}B_f}_{G_{sp}(s)} + \underbrace{C_\infty(sE_\infty - A_\infty)^{-1}B_\infty + D}_{P(s)}. \end{aligned}$$

In MORLAB, block orthogonal transformations are used for the block diagonalization computed by the matrix disk function.

Let $\lambda X - Y$ be a regular matrix pencil with no eigenvalues on the unit circle. Then the right matrix pencil disk function is given as

$$\operatorname{disk}(Y, X) = \lambda \mathcal{P}^0 - \mathcal{P}^\infty,$$

where $\mathcal{P}^0$ and $\mathcal{P}^\infty$ are oblique projections onto the deflating subspaces corresponding to the eigenvalues of $\lambda X - Y$ inside and outside the unit circle, respectively. For the computation of these oblique projections, the inverse free iteration method, see Bai et al. (1997) and Benner (1997), and a subspace extraction method are used; see Sun and Quintana-Ortí (2004) and Benner (2011). Using both of the oblique projections, block orthogonal transformation matrices are derived to do the block diagonalization of the matrix pencil. For the decomposition above, the disk function method has to be applied on the modified matrix pencil $\lambda(\alpha A) - E$, where the parameter α is chosen as

$$\frac{1}{\alpha} > \max\{|\lambda| : \lambda \in \Lambda(A, E) \setminus \{\infty\}\},$$

such that the finite eigenvalues of $\lambda E - A$ correspond to the eigenvalues of the scaled matrix pencil outside of the unit circle and the infinite eigenvalues of $\lambda E - A$ become the zero eigenvalues of the scaled matrix pencil. A more detailed derivation with the formulated algorithms can be found in (Werner, 2016, Chapter 5).

Note that the disk function method is also used for handling unstable systems. After the splitting into finite and infinite eigenvalues, the finite part has to be separated into parts with stable and anti-stable eigenvalues. Therefore, the disk function method is applied to the Cayley-transformed matrix pencil $\lambda(A-E)-(A+E)$. This step is unnecessary for the asymptotically stable system (5). The complete additive decomposition of transfer functions of linear descriptor systems is implemented in the MORLAB routine `ml_adtf_dss`. It should be noted that the disk function method is only suited for medium-scale systems since the block diagonalization of the system matrices destroys in general any sparsity structure. The advantage at this point is, that the additive decomposition is independent of the structure of the original system. Due to the modularity and functionality of this approach, MORLAB is applicable to unstructured, unstable dense systems and it was used to transfer many different model reduction methods from standard to descriptor systems.

3.3 Solving Lyapunov Equations

First, the generalized continuous-time Lyapunov equations (12) and (13) have to be solved. In MORLAB, methods based on the matrix sign function are used for this type of equations.

Let $Y \in \mathbb{R}^{n\times n}$ be a matrix with no purely imaginary eigenvalues. The Jordan canonical form of Y can be written as

$$Y = S \begin{bmatrix} J_- & 0 \\ 0 & J_+ \end{bmatrix} S^{-1}, \qquad (16)$$

where S is an invertible transformation matrix, J_- contains the k eigenvalues with negative real parts and J_+ all the $n-k$ eigenvalues with positive real parts. Then the matrix sign function is defined as

$$\operatorname{sign}(Y) = S \begin{bmatrix} -I_k & 0 \\ 0 & I_{n-k} \end{bmatrix} S^{-1}, \qquad (17)$$

with S the transformation matrix from (16); see, e.g., Roberts (1980). For matrix pencils $\lambda X - Y$ with no purely imaginary eigenvalues and X and Y non-singular, the matrix sign function (17) can be generalized. The computation is done using a Newton iteration of the form

$$Y_{k+1} = \frac{1}{2}\left(\frac{1}{c_k}Y_k + c_k X Y_k^{-1} X\right), \qquad Y_0 = Y, \qquad (18)$$

where c_k are scalars, chosen to accelerate the convergence of the method. In the MORLAB implementation, the Frobenius norm scaling

$$c_k = \sqrt{\frac{\|Y_k\|_F}{\|XY_k^{-1}X\|_F}}$$

is used in the sign function based methods.

To use the sign function for solving the matrix equation (12), the iteration scheme (18) has to be applied on the block matrix pencil

$$\lambda X - Y = \lambda \begin{bmatrix} E_f^T & 0 \\ 0 & E_f \end{bmatrix} - \begin{bmatrix} A_f^T & 0 \\ B_f B_f^T & -A_f \end{bmatrix}.$$

With $Y_{k\to\infty} = \lim_{k\to\infty} Y_k$, the iteration converges to

$$Y_{k\to\infty} = \begin{bmatrix} -E_f^T & 0 \\ 2E_f X_{pc} E_f^T & E_f \end{bmatrix},$$

where X_{pc} is the solution of (12). The resulting block scheme is used to develop an iteration on the system matrices. Additionally, the factorization of the solution $X_{pc} = R_p R_p^T$ is exploited to compute only full-rank factors of B_f. The resulting iteration looks like

$$A_{k+1} = \frac{1}{2}\left(\frac{1}{c_k}A_k + c_k E_f A_k^{-1} E_f\right), \quad A_0 = A_f,$$
$$B_k = \frac{1}{\sqrt{2c_k}}\left[B_k, c_k E_f A_k^{-1} B_k\right], \qquad B_0 = B_f.$$

The solution factor is then obtained by

$$R_p = \frac{1}{\sqrt{2}} E_f^{-1} B_{k\to\infty}.$$

Since the computation of the solution factor in (13) works the same way, both solutions can be computed at the same time using only one iteration on the A matrix. The derivation of the dual Lyapunov solver is shown in Benner et al. (1998) and implemented in the MORLAB routine `ml_lyapdl_sgn_fac`. Like the disk function, also the sign function method is only suited for medium-scale system and will destroy the sparsity structure of the original system.

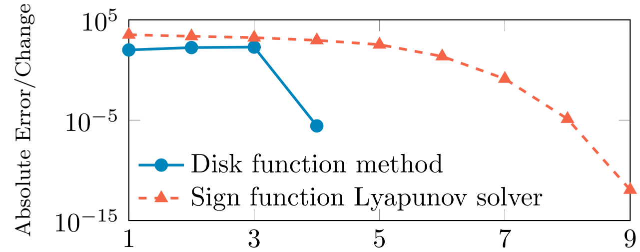

Fig. 2. Convergence of spectral projection methods.

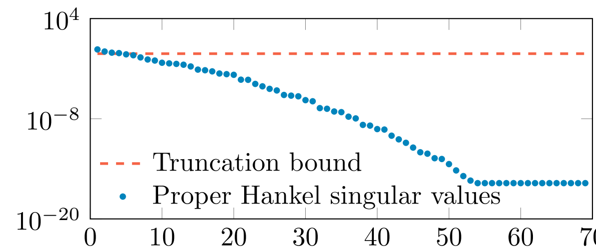

Fig. 3. Computed proper Hankel singular values.

The solution factor R_i of the generalized discrete-time Lyapunov equation (14) is obtained by the exact construction formula

$$R_i = \left[A_\infty^{-1}B_\infty,\ A_\infty^{-1}E_\infty A_\infty^{-1}B_\infty,\ \ldots,\ (A_\infty^{-1}E_\infty)^{\nu-1}A_\infty^{-1}B_\infty\right],$$

with ν the index of the matrix E_∞, which is the index of the system. An analogous formula can be used for the factorized solution of (15). Note that in our example $\nu = 3$, so that $R_i = \left[A_\infty^{-1}B_\infty,\ A_\infty^{-1}E_\infty A_\infty^{-1}B_\infty,\ (A_\infty^{-1}E_\infty)^2 A_\infty^{-1}B_\infty\right]$. The implementation in MORLAB is provided in the subroutine `ml_gdlyap_smith_fac` using a Smith iteration.

4. NUMERICAL RESULTS

In this section, the single steps from the previous sections are applied to the introduced benchmark example (5). The number of masses was chosen as $g = 1500$, which leads to $n = 3001$ states in the first-order system (5). All the computations were done on a machine with one Intel(R) Core(TM) i3-2100 CPU processor running at 3.10GHz and equipped with 4 GB total main memory. The computer is running on Ubuntu 12.04.5 LTS and uses MATLAB 8.0.0.783 (R2012b). The generalized balanced truncation method is implemented in the `ml_bt_dss` function from the version 3.0 of the MORLAB toolbox; see Benner and Werner (2017). By construction, the example is c-stable. That is why unnecessary computations for the decoupling of stable and anti-stable system parts were turned off by setting the optional argument for the dimension of the deflating anti-stable subspace to 0. The MORLAB function call was made by

```
[rom, info] = ml_bt_dss(sys, opts),
```

where `sys` and `rom` are structs containing the full and reduced-order models in first-order form, `info` contains information of all used algorithms and the optional parameters are set by

```
opts = struct( ...
   'stabdecopts', struct('Dimension', 0), ...
   'Tolerance'  , 1.0e-03).
```

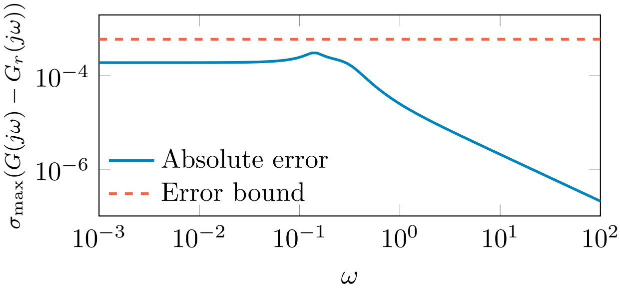

Fig. 4. Sigma error of the full and reduced-order model.

First, the decoupling of the subsystems, using the matrix disk function, was performed to generate the slow subsystem (10) of dimension $n_f = 2998$ and the fast subsystem (11) of dimension $n_\infty = 3$. During the iteration steps of the disk function method, the absolute change of the iteration variable was measured and the four performed steps can be seen in Figure 2. Afterwards, the matrix sign function based dual Lyapunov equation solver was applied to the matrices of the slow subsystem. There, the convergence of A_f to $-E_f$ was measured and is also shown in Figure 2.

The reduced-order model was then computed using the square root method. The desired order of the reduced-order model was determined using the error formula of generalized balanced truncation

$$\|G - G_r\|_{\mathcal{H}_\infty} \leq 2 \sum_{k=r_f+1}^{n_f} \varsigma_k(G),$$

with $\varsigma_k(G)$ the k-th proper Hankel singular value of the system G, to find a reduced-order, such that the $\mathcal{H}_\infty$ approximation error is smaller than 10^{-3}. The computed proper Hankel singular values are shown in Figure 3. One can see the fast decrease in the magnitudes of the singular values such that for the required accuracy only a reduced-order model of order $r = 6$, with $r_f = 6$ and $r_\infty = 0$, is needed. The bound for truncating the proper Hankel singular values is also shown in Figure 3. It has to be noted that, due to numerical errors, the MORLAB routine recognized one improper Hankel singular value with a magnitude of 10^{-18}. That one was automatically truncated by the routine itself. Otherwise, it could be reduced as index-1 part of the reduced-order system to an additional feed-through term of the form $-C_{r_\infty} A_{r_\infty}^{-1} B_{r_\infty} = [0,\ 0,\ 0]^T$. Consequently, this improper Hankel singular value is zero and does not influence the reduced-order system. In general, the number of non-zero improper Hankel singular values is bounded from above by

$$n_{r_\infty} \leq \min(\nu m, \nu p, n_\infty),$$

where ν is the index of the system, m the number of inputs, p the number of outputs and n_∞ the number of infinite eigenvalues of $\lambda E - A$. For the example (5), this upper bound is 3. Since the system was strictly proper by construction, it was expected that all improper Hankel singular values are zero. In Figure 4 the absolute sigma error of the reduced-order ($r = 6$) and the full-order model ($n = 3001$) can be seen. The red dashed line is the computed error bound of the generalized balanced truncation $2 \sum_{k=r_f+1}^{n_f} \varsigma_k(G) \approx 6.0382 \cdot 10^{-4}$.

The sigma plots of the full-order and reduced-order system are both shown in Figure 5. There is no visible difference between the two curves.

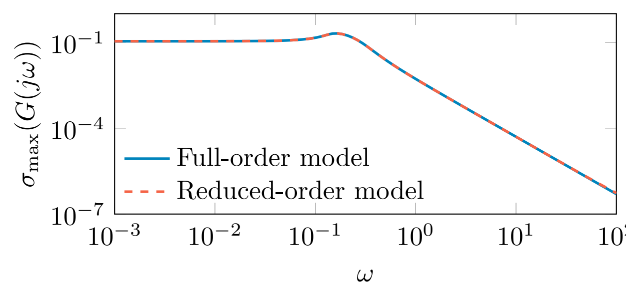

Fig. 5. Sigma plots of the full and reduced-order model.

5. FURTHER MODEL REDUCTION METHODS

Beside the generalized balanced truncation (GBT) method, there are several other model reduction methods for the work with linear descriptor systems (1) implemented in MORLAB. The following listing shows these routines in the version 3.0 of the MORLAB toolbox:

`ml_brbt_dss`	- bounded-real balanced truncation
`ml_bst_dss`	- balanced stochastic truncation
`ml_bt_dss`	- balanced truncation
`ml_hinfbt_dss`	- $\mathcal{H}_\infty$ balanced truncation
`ml_hna_dss`	- Hankel-norm approximation
`ml_lqgbt_dss`	- LQG balanced truncation
`ml_mt_dss`	- modal truncation
`ml_prbt_dss`	- positive-real balanced truncation

In the following, the important properties of these methods are shortly mentioned.

For the model reduction of closed-loop systems, the linear-quadratic Gaussian balanced truncation (LQGBT) method is usually used. Here, the proper and improper Gramians are replaced by the solutions of generalized algebraic Riccati equations (AREs); see Möckel et al. (2011) for more details on this method. A more general approach of the LQGBT is provided by the $\mathcal{H}_\infty$ balanced truncation (HinfBT) using a scaled version of AREs; see Mustafa and Glover (1991). Another method is the balanced stochastic truncation (BST). In case of $m \geq p$, the proper controllability Gramian from the Lyapunov equation (6) is used as in the GBT method. The proper observability Gramian is replaced by the solution of an ARE. BST preserves the minimum phase of the transfer function and for invertible transfer functions, a relative error bound is provided for $\|G^{-1}(G - G_r)\|_{\mathcal{H}_\infty}$. For more details about this method see Benner and Stykel (2017). Important methods in the context of mechanical systems and electrical circuits are positive-real balanced truncation (PRBT) and bounded-real balanced truncation (BRBT) which preserve the positive- and bounded-realness of the descriptor system, respectively. For both methods, AREs have to be solved; see Reis and Stykel (2010) for more details. The different AREs are solved in MORLAB using Newton-type methods, which compute Lyapunov equations in each step using the matrix sign function method.

Besides the balancing-related methods, MORLAB also implements a modal truncation method. The idea here is that the peak behavior of the transfer function in the frequency-response plot strongly depends on the finite eigenvalues of $\lambda E - A$ with negative real parts close to the imaginary axis. The states corresponding to the eigenvalues with larger negative real parts are then truncated; see (Benner et al., 2005, Chapter 1). In the current implementation of MORLAB, the disk function method from Section 3.2 is used for this purpose. Finally, MORLAB also includes a method to solve a best approximation problem, i.e., finding a reduced-order model of a determined order, which minimizes the approximation error in a certain system norm. The resulting method is the Hankel-norm approximation, which solves the best approximation problem in the Hankel semi-norm. Also, the $\mathcal{H}_\infty$ approximation error is often better than for the balanced truncation method. A detailed derivation of the method and its MORLAB implementation can be found in Werner (2016).

For all of these model reduction techniques there is also an adjusted implementation for the standard system case $E = I_n$ in MORLAB using similar techniques as for the descriptor case.

6. CONCLUSIONS

As an example for a linear continuous-time time-invariant descriptor system with singular E matrix, a damped mass-spring system with a holonomic constraint was considered. The generalized balanced truncation method, reformulated as the version implemented in MORLAB, was then introduced for the reduction of the resulting system. The basic methods for the decoupling of descriptor systems using the matrix disk function and for solving the occurring Lyapunov equations were presented, which showed the advantages and disadvantages of MORLAB as a model reduction software for unstructured medium-scale dense systems. The numerical results from the `ml_bt_dss` routine were shown for the introduced example system. Also, an overview about further routines of implemented model reduction methods for descriptor systems in version 3.0 of MORLAB was given.

The presented example of the generalized balanced truncation shows the workflow of model reduction routines for descriptor systems implemented in MORLAB. The toolbox offers a broad spectrum of model reduction methods for the work with general medium-scale descriptor systems.

REFERENCES

Antoulas, A.C. (2005). *Approximation of Large-Scale Dynamical Systems*, volume 6 of *Adv. Des. Control*. SIAM Publications, Philadelphia, PA.

Bai, Z., Demmel, J., and Gu, M. (1997). An inverse free parallel spectral divide and conquer algorithm for nonsymmetric eigenproblems. *Numer. Math.*, 76(3), 279–308.

Baur, U., Benner, P., and Feng, L. (2014). Model order reduction for linear and nonlinear systems: A system-theoretic perspective. *Arch. Comput. Methods Eng.*, 21(4), 331–358.

Benner, P. (1997). *Contributions to the Numerical Solution of Algebraic Riccati Equations and Related Eigenvalue Problems*. Dissertation, Fakultät für Mathematik, TU Chemnitz–Zwickau, 1997.

Benner, P. (2006). A MATLAB repository for model reduction based on spectral projection. In *2006 IEEE Conference on Computer Aided Control System Design*, 19–24.

Benner, P. (2011). Partial stabilization of descriptor systems using spectral projectors. In P. Van Dooren, S.P. Bhattacharyya, R.H. Chan, V. Olshevsky, and A. Routray (eds.), *Numerical Linear Algebra in Signals, Systems and Control*, volume 80 of *Lect. Notes Electr. Eng.*, 55–76. Springer Netherlands.

Benner, P., Claver, J.M., and Quintana-Ortí, E.S. (1998). Efficient solution of coupled Lyapunov equations via matrix sign function iteration. In A. Dourado et al. (eds.), *Proc. 3rd Portuguese Conf. on Automatic Control CONTROLO'98,* Coimbra, 205–210.

Benner, P., Kressner, D., Sima, V., and Varga, A. (2010). Die SLICOT-Toolboxen für MATLAB. *at-Automatisierungstechnik*, 58(1), 15–25.

Benner, P., Mehrmann, V., and Sorensen, D.C. (2005). *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lect. Notes Comput. Sci. Eng.* Springer-Verlag, Berlin/Heidelberg, Germany.

Benner, P. and Stykel, T. (2017). Model order reduction for differential-algebraic equations: A survey. In A. Ilchmann and T. Reis (eds.), *Surveys in Differential-Algebraic Equations IV*, Differential-Algebraic Equations Forum, 107–160. Springer Int. Pub., Cham.

Benner, P. and Werner, S.W.R. (2017). MORLAB-3.0 – model order reduction laboratory. doi:10.5281/zenodo.842659. See also: `http://www.mpi-magdeburg.mpg.de/projects/morlab`.

Möckel, J., Reis, T., and Stykel, T. (2011). Linear-quadratic Gaussian balancing for model reduction of differential-algebraic systems. *Internat. J. Control*, 84(10), 1627–1643.

MORwiki Community (2017). MORwiki - Model Order Reduction Wiki. `http://modelreduction.org`. Date of last access: August 14, 2017.

Mustafa, D. and Glover, K. (1991). Controller design by $\mathcal{H}_\infty$-balanced truncation. *IEEE Trans. Autom. Control*, 36(6), 668–682.

Reis, T. and Stykel, T. (2010). Positive real and bounded real balancing for model reduction of descriptor systems. *Internat. J. Control*, 83(1), 74–88.

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

Stykel, T. (2002). *Analysis and Numerical Solution of Generalized Lyapunov Equations*. Dissertation, TU Berlin.

Stykel, T. (2004). Gramian-based model reduction for descriptor systems. *Math. Control Signals Systems*, 16(4), 297–319.

Sun, X. and Quintana-Ortí, E.S. (2004). Spectral division methods for block generalized Schur decompositions. *Mathematics of Computation*, 73, 1827–1847.

Varga, A. (2016). DSTOOLS - the Descriptor System Tools for MATLAB. `https://sites.google.com/site/andreasvargacontact/home/software/dstools`. Version 0.6: August 31, 2017.

Werner, S. (2016). *Hankel-Norm Approximation of Descriptor Systems*. Master's thesis, Otto-von-Guericke-Universität, Magdeburg, Germany. URL `http://nbn-resolving.de/urn:nbn:de:gbv:ma9:1-8845`.