Adv Comput Math (2020) 46:3
https://doi.org/10.1007/s10444-020-09752-8

Preserving the DAE structure in the Loewner model reduction and identification framework

Ion Victor Gosea[1] · Qiang Zhang[2] · Athanasios C. Antoulas[1,2,3]

Received: 1 October 2018 / Accepted: 21 November 2019 / [redacted]

Abstract

We propose an extension of the Loewner framework to descriptor linear systems that preserves the DAE (differential algebraic equation) structure of the underlying system. More precisely, by means of post-processing the data, the behavior at infinity is matched. As it turns out, the conventional procedure constructs a reduced model by directly compressing the data and hence losing information at infinity. By transforming the matrix pencil composed of the E and A matrices into a generalized block diagonal form, we can separate the descriptor system into two subsystems; one corresponding to the polynomial part and the other to the strictly proper part of the transfer function. Different algorithms are implemented to transform the matrix pencil into block diagonal form. Furthermore, a data-driven splitting of the descriptor system can be achieved in the Loewner framework. Hence, the coefficients of the polynomial part can be estimated directly from data. Several numerical examples are presented to illustrate the theoretical discussion.

Keywords Dynamical systems · Algorithms · Matrices · Differential equations · Complexity reduction

Mathematics subject classification (2010) 78M34 · 93A15 · 93C05 · 93A30 · 37N35 · 65L80

1 Introduction

Model order reduction (MOR) is a tool, commonly used to approximate complex, large-scale dynamical systems by means of much simpler systems that can still

Communicated by: Anthony Nouy

This article belongs to the Topical Collection: *Model reduction of parametrized Systems*
Guest Editors: Anthony Nouy, Peter Benner, Mario Ohlberger, Gianluigi Rozza, Karsten Urban and Karen Willcox

✉ Ion Victor Gosea
gosea@mpi-magdeburg.mpg.de

Extended author information available on the last page of the article.

faithfully reproduce the behavior or response of the original system. Such high-dimensional models usually appear from spatial discretization of partial differential equations (PDEs). The denser the grid used, the higher the number of equations that characterize the behavior. Reduced-order systems are described by significantly lower number of equations. Such systems could then be efficiently used as surrogates, replacing the large-scale systems in such tasks as simulation, analysis, or control. For details on different MOR techniques, we refer the reader to the books [1, 4, 12] and to the surveys [3, 7, 10].

In the current work, we will turn our attention to interpolatory model reduction methods. In this setup, ROMs (reduced-order models) are constructed by means of interpolating the transfer function of the original system at selected points.

The Loewner framework, as introduced in [34], will be the main focus of this study. It is a data-driven method that directly constructs a ROM using only measured data (samples of the transfer function). Additionally, by compressing the (usually large) data set, it extracts the dominant features. For a recent survey on the Loewner framework for linear systems, see [6]. For the nonlinear case, this method requires appropriate definition of transfer functions, as described in some of its extensions, i.e., to bilinear [5], quadratic-bilinear [23], and switched systems [24].

We propose an extension of the Loewner framework to a specific class of linear systems, i.e., descriptor linear systems. The new method can preserve the underlying DAE structure by means of post-processing of the data (estimating the polynomial part of the transfer function).

Whenever the dynamics of the driving system is described by not only differential but also algebraic equations, descriptor systems come into play. Such systems are described by DAEs and are used to model many different real-world applications, such as the ones appearing in electrical networks/circuits, biomedical and chemical processes, or computational fluid dynamics. We refer the reader to the books [22, 33] for more details. Additionally, the following surveys address particular topics related to systems described by linear DAEs: partial stabilization, in [9], description of the solution in [39], controllability in [16], and numerical linear algebra methods in [11]. The concept of partial realization is covered in [13], while topics on simulation and control in [36].

Modeling of complex physical and technical processes such as control of laminar/turbulent fluid flow, very large integrated circuit simulation for chip design or components coupling in mechanical systems, typically leads to descriptor systems of considerably large dimension. This generates bottlenecks such as storage requirements and expensive computations when dealing with this class of systems. This is where reduced models play an important role. Consequently, in recent years, many such methods for reducing conventional linear systems (with no algebraic constraints) have been extended for reduction of systems described by DAEs. In particular, balanced truncation has been extended to descriptor systems in [26, 35, 38] while interpolation-based methods were considered in [25, 37]. For more details on MOR methods applied to DAEs, we refer the readers to the extensive survey in [14].

The paper is organized in the following way: in Section 2, we give a short overview on descriptor systems, introduce the Weierstrass and Kronecker canonical forms together with some relaxed versions thereof. Additionally, we discuss how to explicitly compute the polynomial coefficients of the transfer function. In Section 3, we present several methods for splitting the strictly proper and polynomial part of the transfer function, for both systems involving regular and singular pencils. Section 4 introduces the classical Loewner framework and afterwards, the main proposed procedure that is based on it. The difference is that, by means of post-processing of the data, we are able to match the behavior at infinity of the underlying system. In Section 5, several numerical experiments are presented (three of them chosen from benchmark examples commonly used for MOR applications), while a summary of the findings and the conclusion are stated in Section 6.

2 Descriptor systems

2.1 Setup

We study linear time-invariant continuous-time descriptor systems described by the following equations:

$$\boldsymbol{\Sigma} : \begin{cases} \mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t), \end{cases} \tag{1}$$

where $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n \times n}$, $\mathbf{C} \in \mathbb{R}^{p \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, and $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\mathbf{y}(t) \in \mathbb{R}^p$ are the internal variable, input and output vectors, respectively. The order of system $\boldsymbol{\Sigma}$ in (1) is given by the number of internal variables n. If the $\mathbf{E}$ matrix is the identity matrix, then the system in (1) is a standard state space system. Otherwise, it is a descriptor system or generalized state space system.

The goal is to approximate the original descriptor system $\boldsymbol{\Sigma}$ in (1) by a reduced-order linear time-invariant continuous-time descriptor system $\hat{\boldsymbol{\Sigma}}$

$$\hat{\boldsymbol{\Sigma}} : \begin{cases} \hat{\mathbf{E}}\dot{\hat{\mathbf{x}}}(t) &= \hat{\mathbf{A}}(t) + \hat{\mathbf{B}}\mathbf{u}(t), \\ \hat{\mathbf{y}}(t) &= \hat{\mathbf{C}}\hat{\mathbf{x}}(t), \end{cases} \tag{2}$$

where $\hat{\mathbf{A}}, \hat{\mathbf{E}} \in \mathbb{R}^{r \times r}$, $\hat{\mathbf{C}} \in \mathbb{R}^{p \times r}$, $\hat{\mathbf{B}} \in \mathbb{R}^{r \times m}$, and $\hat{\mathbf{x}}(t) \in \mathbb{R}^r$ is the reduced internal variable, with $r \ll n$. The control input $\mathbf{u}(t)$ is the same for both systems in (1) and (2). In general, it is required that the approximation error be small.

The input-output behavior of the descriptor system $\boldsymbol{\Sigma}$ is usually described by its transfer function $\mathbf{H}$. Assuming that the pencil $\lambda\mathbf{E} - \mathbf{A}$ is regular, i.e., $\det(\lambda\mathbf{E} - \mathbf{A}) \neq 0$ for $\lambda \in \mathbb{C}$, then one can write the transfer function $\mathbf{H}$ as follows:

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}. \tag{3}$$

If the pencil $\lambda\mathbf{E} - \mathbf{A}$ is singular (not regular), i.e., $\det(\lambda\mathbf{E} - \mathbf{A}) = 0$ for all $\lambda \in \mathbb{C}$, one needs to replace the conventional inverse of the matrix $s\mathbf{E} - \mathbf{A}$ in (3) with a pseudo-inverse. In this case, the transfer function $\mathbf{H}$ can be written

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{\#}\mathbf{B}, \tag{4}$$

where $\mathbf{X}^{\#} \in \mathbb{C}^{u \times v}$ is the Moore–Penrose inverse of the matrix $\mathbf{X} \in \mathbb{C}^{v \times u}$ with $u, v \in \mathbb{N}$ (for more details, see [2]).

For simplicity of exposition, in what follows, we will treat solely the single-input, single-output (SISO) case with $m = p = 1$. The multi-input, multi-output case is technically more involved but it is based on the same ideas.

The transfer function shown in (3) can be expressed as a rational function in the variable $s \in \mathbb{C}$

$$\mathbf{H}(s) = \frac{\sum_{i=0}^{z} b_i s^i}{\sum_{j=0}^{q} a_j s^j}\,, \tag{5}$$

where $a_j, b_i \in \mathbb{R}$, z is the number of zeros and q is the number of poles. When $z > q$, the transfer function $\mathbf{H}(s)$ is improper. When $z \leq q$, $\mathbf{H}(s)$ is proper and when $z < q$, $\mathbf{H}(s)$ is strictly proper. If $z \geq q$, the transfer function $\mathbf{H}(s)$ in (3) can be decomposed into a strictly proper part and a polynomial part, as

$$\mathbf{H}(s) = \mathbf{H}_{sp}(s) + \mathbf{H}_{\text{poly}}(s), \tag{6}$$

where

$$\mathbf{H}_{sp}(s) = \frac{\sum_{i=0}^{q-1} c_i s^i}{\sum_{j=0}^{q} a_j s^j}, \qquad \mathbf{H}_{\text{poly}}(s) = \sum_{k=0}^{z-q} p_k s^k, \tag{7}$$

are the strictly proper part and polynomial part, respectively. In (7), the polynomial coefficients of $\mathbf{H}_{\text{poly}}(s)$ are denoted with p_k, for $k \in \{0, 1, \ldots, z-q\}$. In what follows, we discuss how to determine the polynomial part $\mathbf{H}_{\text{poly}}(s)$ of the descriptor system $\boldsymbol{\Sigma}$ and how to explicitly compute the coefficients p_k.

2.2 Canonical forms for regular pencils

2.2.1 The Weierstrass canonical forms

Let $\boldsymbol{\Sigma}$ be a descriptor system as in (1) and assume that the pencil $\lambda\mathbf{E} - \mathbf{A}$ is regular. In this case, the pencil $\lambda\mathbf{E} - \mathbf{A}$ can be reduced to the Weierstrass canonical form, defined next.

Definition 1 The regular matrix pencil $\lambda\mathbf{E} - \mathbf{A}$ can be written in the Weierstrass canonical form [35] given by

$$\mathbf{S}^{-1}(\lambda\mathbf{E} - \mathbf{A})\mathbf{T} = \begin{bmatrix} \lambda\mathbf{I}_{n_f} - \mathbf{J}_f & \mathbf{0} \\ \mathbf{0} & \lambda\mathbf{N} - \mathbf{I}_{n_\infty} \end{bmatrix}, \tag{8}$$

where the matrices $\mathbf{S}, \mathbf{T} \in \mathbb{C}^{n \times n}$ are invertible, $\mathbf{J}_f \in \mathbb{C}^{n_f \times n_f}$ and $\mathbf{N} \in \mathbb{C}^{n_\infty \times n_\infty}$ are matrices in Jordan canonical form and $\mathbf{N}$ is a nilpotent matrix. The positive integers n_f and n_∞ are the dimensions of the deflating subspaces of the pencil corresponding to the finite and infinite eigenvalues, respectively. Furthermore, $\mathbf{I}_{n_f} \in \mathbb{R}^{n_f \times n_f}$ and $\mathbf{I}_{n_\infty} \in \mathbb{R}^{n_\infty \times n_\infty}$ are identity matrices.

Let ν be the nilpotency index of $\mathbf{N}$, i.e., the smallest positive integer ν such that $\mathbf{N}^\nu = \mathbf{0}$. The positive integer ν is sometimes referred to as the index of the pencil $\lambda\mathbf{E} - \mathbf{A}$ and also the Hessenberg index of the descriptor system $\boldsymbol{\Sigma}$.

In [15], a so-called quasi-Weierstrass canonical form is introduced. It is weaker than the classical form although it also contains relevant information.

Next, introduce a generalized Weierstrass canonical form of the matrix pencil $\lambda\mathbf{E}-\mathbf{A}$. Here, the transformed $\mathbf{E}$ and $\mathbf{A}$ are indeed block diagonal, but the composing blocks need not have any special structure.

Definition 2 The regular matrix pencil $\lambda\mathbf{E}-\mathbf{A}$ can be written in the generalized quasi-Weierstrass canonical form (or generalized block diagonal form) given by

$$\mathbf{U}^*(\lambda\mathbf{E}-\mathbf{A})\mathbf{V}=\begin{bmatrix}\lambda\mathbf{E}_f-\mathbf{A}_f & \mathbf{0}\\ \mathbf{0} & \lambda\mathbf{E}_\infty-\mathbf{A}_\infty\end{bmatrix}, \tag{9}$$

where $\mathbf{U},\mathbf{V}\in\mathbb{R}^{n\times n}$ are invertible, $\mathbf{E}_f\in\mathbb{R}^{n_f\times n_f}$ is invertible, $\mathbf{A}_f\in\mathbb{R}^{n_f\times n_f}$ and $\mathbf{E}_\infty\in\mathbb{R}^{n_\infty\times n_\infty}$, $\mathbf{A}_\infty\in\mathbb{R}^{n_\infty\times n_\infty}$ are invertible.

The transformation matrices $\mathbf{U}$ and $\mathbf{V}$ can be split as $\mathbf{U}=\begin{bmatrix}\mathbf{U}_f & \mathbf{U}_\infty\end{bmatrix}$ and $\mathbf{V}=\begin{bmatrix}\mathbf{V}_f & \mathbf{V}_\infty\end{bmatrix}$. According to the block diagonal form in (9), identify the diagonal blocks for both $\mathbf{U}^*\mathbf{E}\mathbf{V}$ and $\mathbf{U}^*\mathbf{A}\mathbf{V}$ matrices as

$$\mathbf{E}_f=\mathbf{U}_f^*\mathbf{E}\mathbf{V}_f,\quad \mathbf{E}_\infty=\mathbf{U}_\infty^*\mathbf{E}\mathbf{V}_\infty,\quad \mathbf{A}_f=\mathbf{U}_f^*\mathbf{A}\mathbf{V}_f,\quad \mathbf{A}_\infty=\mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_\infty, \tag{10}$$

and, respectively, the off-diagonal blocks as

$$\mathbf{U}_f^*\mathbf{E}\mathbf{V}_\infty=\mathbf{0},\quad \mathbf{U}_\infty^*\mathbf{E}\mathbf{V}_f=\mathbf{0},\quad \mathbf{U}_f^*\mathbf{A}\mathbf{V}_\infty=\mathbf{0},\quad \mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_f=\mathbf{0}. \tag{11}$$

The matrix pencil $\lambda\mathbf{E}-\mathbf{A}$ is transformed into a block diagonal form, and hence, the descriptor system $\boldsymbol{\Sigma}$ can be decomposed into two subsystems, one with realization $\boldsymbol{\Sigma}_f:(\mathbf{C}_f,\mathbf{E}_f,\mathbf{A}_f,\mathbf{B}_f)=(\mathbf{C}\mathbf{V}_f,\mathbf{E}_f,\mathbf{A}_f,\mathbf{U}_f^*\mathbf{B})$ and another with realization $\boldsymbol{\Sigma}_\infty:(\mathbf{C}_\infty,\mathbf{E}_\infty,\mathbf{A}_\infty,\mathbf{B}_\infty)=(\mathbf{C}\mathbf{V}_\infty,\mathbf{E}_\infty,\mathbf{A}_\infty,\mathbf{U}_\infty^*\mathbf{B})$. The transfer functions of the two subsystems $\boldsymbol{\Sigma}_f$ and $\boldsymbol{\Sigma}_\infty$ are computed as

$$\begin{aligned}\mathbf{H}_f(s)&=\mathbf{C}\mathbf{V}_f(s\mathbf{E}_f-\mathbf{A}_f)^{-1}\mathbf{U}_f^*\mathbf{B},\\ \mathbf{H}_\infty(s)&=\mathbf{C}\mathbf{V}_\infty(s\mathbf{E}_\infty-\mathbf{A}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{B}.\end{aligned} \tag{12}$$

Note that the transfer functions introduced in (12) coincide with those in (7):

$$\mathbf{H}_{sp}(s)=\mathbf{H}_f(s),\qquad \mathbf{H}_{poly}(s)=\mathbf{H}_\infty(s). \tag{13}$$

2.2.2 Computing the polynomial coefficients of the transfer function

After splitting the polynomial part $\mathbf{H}_\infty$ from the transfer function $\mathbf{H}(s)$ of system $\boldsymbol{\Sigma}$, the polynomial coefficients p_k in (7) are explicitly computed. As presented in the following result, this is done by calculating the derivatives of $\mathbf{H}_\infty$ at $s=0$ or, equivalently, expressing them in terms of the system matrices of the subsystem $\boldsymbol{\Sigma}_\infty$.

Lemma 1 *The coefficients p_k corresponding to the polynomial part of transfer function $\mathbf{H}$ can be expressed as*

$$p_k=\frac{1}{k!}\left[\frac{\mathrm{d}^k}{\mathrm{d}s^k}\mathbf{H}_\infty(s)\right]_{s=0}=-\mathbf{C}\mathbf{V}_\infty(\mathbf{A}_\infty^{-1}\mathbf{E}_\infty)^k\mathbf{A}_\infty^{-1}\mathbf{U}_\infty^*\mathbf{B}, \tag{14}$$

so that the constant term is written $p_0=-\mathbf{C}\mathbf{V}_\infty\mathbf{A}_\infty^{-1}\mathbf{U}_\infty^*\mathbf{B}=\mathbf{H}_\infty(0)$ and then the linear term $p_1=-\mathbf{C}\mathbf{V}_\infty(\mathbf{A}_\infty^{-1}\mathbf{E}_\infty)\mathbf{A}_\infty^{-1}\mathbf{U}_\infty^*\mathbf{B}$ and so on.

Proof In (7), since $\mathbf{H}_\infty(s) = \mathbf{H}_{\text{poly}}(s) = \sum_{k=0}^{z-q} p_k s^k$, it directly follows that the kth derivative of $\mathbf{H}_\infty(s)$ can be written as

$$\frac{\mathrm{d}^k}{\mathrm{d}s^k}\mathbf{H}_\infty(s) = k!p_k + \frac{(k+1)!}{1!}p_{k+1}s + \cdots + \frac{(z-q)!}{(z-q-k)!}p_{z-q}s^{z-q-k}. \tag{15}$$

By substituting $s = 0$ in (15), it automatically follows $p_k = \frac{1}{k!}\left[\frac{\mathrm{d}^k}{\mathrm{d}s^k}\mathbf{H}_\infty(s)\right]_{s=0}$. Using the definition of $\mathbf{H}_\infty$ in equation (12), the kth derivative of $\mathbf{H}_\infty$ is

$$\begin{aligned}\tfrac{\mathrm{d}^k}{\mathrm{d}s^k}\mathbf{H}_\infty(s) &= \tfrac{\mathrm{d}^k}{\mathrm{d}s^k}\mathbf{C}\mathbf{V}_\infty(s\mathbf{E}_\infty - \mathbf{A}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{B} \\ &= (-1)^k k!\mathbf{C}\mathbf{V}_\infty((s\mathbf{E}_\infty - \mathbf{A}_\infty)^{-1}\mathbf{E}_\infty)^k(s\mathbf{E}_\infty - \mathbf{A}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{B}.\end{aligned}$$

So if $k > 0$, then

$$p_k = \frac{1}{k!}\left[\frac{\mathrm{d}^k}{\mathrm{d}s^k}\mathbf{H}_\infty(s)\right]_{s=0} = -\mathbf{C}\mathbf{V}_\infty(\mathbf{A}_\infty^{-1}\mathbf{E}_\infty)^k\mathbf{A}_\infty^{-1}\mathbf{U}_\infty^*\mathbf{B} \tag{16}$$

and otherwise $p_0 = -\mathbf{C}\mathbf{V}_\infty\mathbf{A}_\infty^{-1}\mathbf{U}_\infty^*\mathbf{B}$. □

2.2.3 The right matrix pencil disk function

In this section, we assume that the original system is a discrete-time descriptor system computed, for example, by means of applying finite difference approximation of the derivative of the continuous internal variable in (1). Let $\lambda\mathbf{E} - \mathbf{A}$ be a regular matrix pencil with no eigenvalues on the unit circle. The Weierstrass canonical form of $\lambda\mathbf{E} - \mathbf{A}$ is given by

$$\overline{\mathbf{S}}^{-1}(\lambda\mathbf{E} - \mathbf{A})\overline{\mathbf{T}} = \begin{bmatrix}\lambda\mathbf{I}_{n_k} - \overline{\mathbf{J}}_0 & \mathbf{0} \\ \mathbf{0} & \lambda\overline{\mathbf{N}} - \overline{\mathbf{J}}_\infty\end{bmatrix}, \tag{17}$$

where $\overline{\mathbf{J}}_0 \in \mathbb{C}^{n_k \times n_k}$ and $\overline{\mathbf{J}}_\infty \in \mathbb{C}^{(n-n_k)\times(n-n_k)}$.

Definition 3 The *right matrix pencil disk function* is defined

$$\operatorname{disk}(\mathbf{A}, \mathbf{E}) = \overline{\mathbf{T}}\left(\lambda\begin{bmatrix}\mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n-n_k}\end{bmatrix} - \begin{bmatrix}\mathbf{I}_{n_k} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\right)\overline{\mathbf{T}}^{-1} = \lambda\mathcal{P}_\infty - \mathcal{P}_0. \tag{18}$$

The matrices $\mathcal{P}_0$ and $\mathcal{P}_\infty$ define skew projections onto the right deflating subspace corresponding to the eigenvalues of $\lambda\mathbf{E} - \mathbf{A}$ inside the unit circle and respectively outside the unit circle. The disk function provides a tool for spectral decomposition along the unit circle. Splittings for other curves can be computed by applying a suitable conformal mapping to the pencil $\lambda\mathbf{E} - \mathbf{A}$.

The method for the computation of the right matrix disk function can be found in [9, 41]. The steps of this method are summarized in Algorithm 3 from [9] and in Algorithm 8 from [41].

2.3 Canonical forms for singular pencils

2.3.1 The Kronecker canonical forms

Let $\boldsymbol{\Sigma}$ be a continuous-time descriptor system, as in (1), and assume that the pencil $\lambda\mathbf{E}-\mathbf{A}$ is singular, i.e., $\det(\lambda\mathbf{E}-\mathbf{A}) = 0$ for all $\lambda \in \mathbb{C}$. In this case, the pencil $\lambda\mathbf{E}-\mathbf{A}$ can be reduced to the Kronecker canonical form (KCF).

Definition 4 The matrix pencil $\lambda\mathbf{E}-\mathbf{A}$ can be written in the Kronecker canonical form [40]

$$\mathbf{P}(\lambda\mathbf{E}-\mathbf{A})\mathbf{Q} = \mathrm{diag}(\mathbf{L}_{\epsilon_1}, \ldots, \mathbf{L}_{\epsilon_p}, \mathbf{L}^P_{\eta_1}, \ldots, \mathbf{L}^P_{\eta_q}, \lambda\mathbf{N}-\mathbf{I}, \lambda\mathbf{I}-\mathbf{J}), \tag{19}$$

where $\mathbf{P}, \mathbf{Q} \in \mathbb{C}^{n\times n}$ are invertible matrices. The building blocks present in (19) (by following the description in [40]) are as follows:

- $\mathbf{L}_\mu \in \mathbb{C}^{\mu\times(\mu+1)}$ is the bidiagonal pencil.
- $\mathbf{L}^P_\mu \in \mathbb{C}^{(\mu+1)\times\mu}$ is the pertransposed bidiagonal pencil.
- $\mathbf{N}$ is a nilpotent Jordan matrix.
- $\mathbf{J}$ is a matrix in Jordan canonical form.

Note that the pencil $\lambda\mathbf{I}-\mathbf{J}$ contains the finite elementary divisors (finite eigenvalues of $\lambda\mathbf{E}-\mathbf{A}$), while $\lambda\mathbf{N}-\mathbf{I}$ contains the infinite elementary divisors (infinite eigenvalues of $\lambda\mathbf{E}-\mathbf{A}$). Moreover, the blocks $\mathbf{L}_{\epsilon_i}$ and $\mathbf{L}^P_{\eta_j}$ contain the singularities of the pencil. The values ϵ_i and η_j are known as the right and left Kronecker indices. The pencil $\lambda\mathbf{N}-\mathbf{I}$ is determined by the degrees δ_i of the infinite elementary divisors, while $\lambda\mathbf{I}-\mathbf{J}$ is determined by the finite elementary divisors $(\lambda-\lambda_i)^j$ (here, the finite eigenvalues of the pencil $\lambda\mathbf{E}-\mathbf{A}$ are denoted with λ_i).

As mentioned in [8, 40], to compute the Kronecker canonical form, it is recommended to first compute a quasi-triangular form (also called the generalized Schur form) in order to avoid numerical issues:

$$\overline{\mathbf{P}}(\lambda\mathbf{E}-\mathbf{A})\overline{\mathbf{Q}} = \begin{bmatrix} \lambda\mathbf{E}_\epsilon-\mathbf{A}_\epsilon & \mathbf{X}_{12} & \mathbf{X}_{13} & \mathbf{X}_{14} \\ \mathbf{0} & \lambda\mathbf{E}_\eta-\mathbf{A}_\eta & \mathbf{X}_{23} & \mathbf{X}_{24} \\ \mathbf{0} & \mathbf{0} & \lambda\mathbf{E}_\infty-\mathbf{A}_\infty & \mathbf{X}_{34} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \lambda\mathbf{E}_f-\mathbf{A}_f \end{bmatrix}. \tag{20}$$

This form can be obtained using the unitary transformation matrices $\overline{\mathbf{P}}, \overline{\mathbf{Q}} \in \mathbb{C}^{n\times n}$. Moreover, the block diagonal structure is as follows:

1. The square regular pencil $\lambda\mathbf{E}_f-\mathbf{A}_f$ contains the finite eigenvalues of $\lambda\mathbf{E}-\mathbf{A}$.
2. The square regular pencil $\lambda\mathbf{E}_\infty-\mathbf{A}_\infty$ contains the infinite eigenvalues of $\lambda\mathbf{E}-\mathbf{A}$.
3. The singular pencils $\lambda\mathbf{E}_\eta-\mathbf{A}_\eta$ and $\lambda\mathbf{E}_\epsilon-\mathbf{A}_\epsilon$ contain the Kronecker row and column structure, respectively.

3 Separating the strictly proper and polynomial part of the transfer function

In Section 2, we presented the theoretical aspects of decomposing the matrix pencil $\lambda\mathbf{E}-\mathbf{A}$ into two subsystems, corresponding to the strictly proper part and polynomial part of the transfer function $\mathbf{H}(s)$. Next, we present different numerical algorithms for computing the matrices $\mathbf{U}$ and $\mathbf{V}$. These are used in order to construct a quasi-Weierstrass form. Four such algorithms are studied in this section. First, the Jordan chain (JC) method finds the transformation matrices by calculating the generalized eigenvectors corresponding to the infinite eigenvalues. Then, the ADTF1 (additive decomposition of a transfer function no. 1) and ADTF2 (additive decomposition of a transfer function no. 2) methods from [30] use the QZ decomposition to separate the descriptor system. Finally, a recent procedure based on the matrix disk function in [41] is mentioned.

3.1 Regular pencils

3.1.1 Jordan chain method

This method requires computing matrices $\mathbf{U}_\infty \in \mathbb{C}^{n\times n_\infty}$ and $\mathbf{V}_\infty \in \mathbb{C}^{n\times n_\infty}$ so that their columns span the same subspace as the left and, respectively, right generalized eigenvectors corresponding to the eigenvalue at infinity of the pencil $\lambda\mathbf{E}-\mathbf{A}$ (or, equivalently, of eigenvalues at 0 of the pencil $\lambda\mathbf{A}-\mathbf{E}$)

$$\mathbf{E}\mathbf{V}_\infty = \mathbf{A}\mathbf{V}_\infty\mathbf{J}, \quad \mathbf{U}_\infty^*\mathbf{E} = \mathbf{J}^*\mathbf{U}_\infty^*\mathbf{A}, \tag{21}$$

where $\mathbf{J} \in \mathbb{C}^{n_\infty\times n_\infty}$ is a Jordan block with zero eigenvalues. Then, the matrices $\mathbf{U}_f, \mathbf{V}_f \in \mathbb{C}^{n\times n_f}$ can be found by computing the null space of matrices $\mathbf{V}_\infty^*\mathbf{A}^*$ and, respectively, of $\mathbf{U}_\infty^*\mathbf{A}$:

$$\mathbf{U}_f^*\mathbf{A}\mathbf{V}_\infty = \mathbf{0}, \quad \mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_f = \mathbf{0}. \tag{22}$$

Then, construct the transformation matrices $\mathbf{U} = \begin{bmatrix}\mathbf{U}_f & \mathbf{U}_\infty\end{bmatrix}$ and $\mathbf{V} = \begin{bmatrix}\mathbf{V}_f & \mathbf{V}_\infty\end{bmatrix}$ by putting together the matrices in (21) and (22).

3.1.2 Additive decomposition of a transfer function no. 1

This approach was introduced in [30] Section 4.1. The first step is to apply the QZ (known also as generalized Schur) decomposition to the matrix pencil $\lambda\mathbf{E}-\mathbf{A}$. This is done by imposing that the finite eigenvalues of this pencil are located in the upper-left corner while the remaining infinite eigenvalues are in the lower-right corner:

$$\mathbf{Q}(\lambda\mathbf{E}-\mathbf{A})\mathbf{Z} = \begin{bmatrix}\mathbf{Q}_1\\ \mathbf{Q}_2\end{bmatrix}(\lambda\mathbf{E}-\mathbf{A})\begin{bmatrix}\mathbf{Z}_1 & \mathbf{Z}_2\end{bmatrix} = \begin{bmatrix}\lambda\mathbf{E}_{11}-\mathbf{A}_{11} & \lambda\mathbf{E}_{12}-\mathbf{A}_{12}\\ \mathbf{0} & \lambda\mathbf{E}_{22}-\mathbf{A}_{22}\end{bmatrix}. \tag{23}$$

Next, construct matrices $\mathbf{T}$ and $\mathbf{S}$ in the following way

$$\mathbf{T} = \begin{bmatrix}\mathbf{I}_{n_f} & \mathbf{L}\\ \mathbf{0} & \mathbf{I}_{n_\infty}\end{bmatrix} \; \mathbf{S} = \begin{bmatrix}\mathbf{I}_{n_f} & \mathbf{R}\\ \mathbf{0} & \mathbf{I}_{n_\infty}\end{bmatrix}, \text{ where } \begin{cases}\mathbf{E}_{11}\mathbf{R}+\mathbf{L}\mathbf{E}_{22} = -\mathbf{E}_{12},\\ \mathbf{A}_{11}\mathbf{R}+\mathbf{L}\mathbf{A}_{22} = -\mathbf{A}_{12}.\end{cases} \tag{24}$$

The transformed matrices $\mathbf{E}_1 = \mathbf{TQEZS}$, $\mathbf{A}_1 = \mathbf{TQAZS}$ are thus block diagonal. Numerical methods for solving coupled Sylvester equations as in (24) were proposed in [27, 29, 31]. The left and right transformation matrices can be explicitly constructed as follows $\mathbf{U} = (\mathbf{TQ})^*$, $\mathbf{V} = \mathbf{ZS}$.

3.1.3 Additive decomposition of a transfer function no. 2

Again, as in Section 3.1.2, the QZ factorization is applied to the matrix pencil $\lambda\mathbf{E} - \mathbf{A}$ with two different orderings of the eigenvalues (as in Section 4.2 in [30]). This is done by imposing that the finite and infinite eigenvalues of the pencil are located in the upper-left corner and in the lower-right corner, respectively:

$$\mathbf{Q}^u(\lambda\mathbf{E} - \mathbf{A})\mathbf{Z}^u = \begin{bmatrix} \mathbf{Q}_1^u \\ \mathbf{Q}_2^u \end{bmatrix} (\lambda\mathbf{E} - \mathbf{A}) \begin{bmatrix} \mathbf{Z}_1^u & \mathbf{Z}_2^u \end{bmatrix} = \begin{bmatrix} \lambda\mathbf{E}_{11}^u - \mathbf{A}_{11}^u & \lambda\mathbf{E}_{12}^u - \mathbf{A}_{12}^u \\ \mathbf{0} & \lambda\mathbf{E}_{22}^u - \mathbf{A}_{22}^u \end{bmatrix}, \tag{25}$$

$$\mathbf{Q}^l(\lambda\mathbf{E} - \mathbf{A})\mathbf{Z}^l = \begin{bmatrix} \mathbf{Q}_1^l \\ \mathbf{Q}_2^l \end{bmatrix} (\lambda\mathbf{E} - \mathbf{A}) \begin{bmatrix} \mathbf{Z}_1^l & \mathbf{Z}_2^l \end{bmatrix} = \begin{bmatrix} \lambda\mathbf{E}_{11}^l - \mathbf{A}_{11}^l & \lambda\mathbf{E}_{12}^l - \mathbf{A}_{12}^l \\ \mathbf{0} & \lambda\mathbf{E}_{22}^l - \mathbf{A}_{22}^l \end{bmatrix}. \tag{26}$$

The transformation matrices are

$$\mathbf{U} = \begin{bmatrix} \mathbf{Q}_2^l \\ \mathbf{Q}_2^u \end{bmatrix}^*, \qquad \mathbf{V} = \begin{bmatrix} \mathbf{Z}_1^u & \mathbf{Z}_1^l \end{bmatrix}. \tag{27}$$

3.1.4 Additive decomposition of a transfer function no. 3

The decompositions covered in Sections 3.1.2 and 3.1.3 rely on computing one or two QZ factorizations (as described in [30]).

Recently, a new method was proposed in [41] that overcomes the usage of such factorizations and, instead, uses the disk function method (briefly mentioned in Section 2.2.3) to construct projection matrices for subspace extraction.

The complete procedure is stated in Algorithm 10 from [41]. The first step is to apply the disk function method to the pencil $\lambda(\alpha\mathbf{A}) - \mathbf{E}$, where the real positive scalar α is chosen so that $\alpha < \max\left\{|\lambda| : \lambda \in \Lambda(\mathbf{A}, \mathbf{E}) \setminus \{\infty\}\right\}^{-1}$, and $\Lambda(\mathbf{A}, \mathbf{E})$ denotes the spectrum of $\lambda\mathbf{E} - \mathbf{A}$. Here, α is used as eigenvalue scaling such that the infinite eigenvalues become zeros and everything else is pushed outside the unit circle. Hence, the disk function method produces skew projections onto the corresponding deflating subspaces.

3.2 Singular pencils

The computation of the Kronecker canonical form (KCF) received considerable attention in the second half of the previous century. This follows from the importance of the KCF in the area of systems and control theory, i.e., realization theory and controllability. In what follows, we mention only a few of the contributions.

1. A generalization of the staircase algorithm to singular pencils using unitary equivalence transformations is proposed in [40].

2. The so-called AB algorithm for computing the right singular structure and the Jordan structure of the zero eigenvalue for a singular pencil is introduced in [32].
3. An algorithm based on repeated generalized singular value decompositions (or more precisely, cosine-sine decompositions of partitioned orthonormal matrices), in short referred to as RGSVD, is presented in [28].
4. A faster algorithm which requires $O(m^2n)$ operations when the original pencil is $m \times n$ is proposed in [8].
5. Robust algorithms for implementing the generalized Schur decomposition of an arbitrary pencil $A - \lambda B$ are presented in [19].
6. A procedure based on the so-called Wong sequences is proposed in order to derive a quasi-Kronecker form in [17].

The procedures mentioned above are quite involved and not straightforward to implement in a practical setup.

The toolbox GUPTRI (Generalized Upper TRIangular form) presented in [20] can be used to compute the generalized Schur decomposition of an arbitrary pencil $A - \lambda B$ (regular or singular), hence revealing the Kronecker structure of the pencil. The algorithms are implemented in Fortran (also in Matlab) and are available as an open source toolbox (see [21]).

Most of the numerical experiments presented in Section 5 are based on the following block diagonalization procedure (in Section 3.2.1). This is also used as a direct numerical tool for estimating the polynomial coefficients.

3.2.1 The proposed procedure

We are using a numerical procedure inspired from the decomposition that was briefly mentioned in Section 3.1.3. Originally proposed in [30], this procedure uses unitary transformations to block diagonalize a regular matrix pencil based on two re-orderings of the pencil's eigenvalues.

For the first re-ordering, the blocks that have eigenvalues belonging to a set Γ are placed in the lower-right corner of the transformed upper triangular matrices. For the second re-ordering, the order is reversed (the blocks that have eigenvalues belonging to the set Γ are placed in the upper-left corner).

The procedure is based on performing two QZ (or generalized Schur) decompositions in complex arithmetic for a regular matrix pencil $(\mathbf{A}, \mathbf{E})$.

We are dealing with singular pairs $(\mathbf{A}, \mathbf{E})$ that have eigenvalues at ∞. So instead of analyzing the spectrum of the pencil $(\mathbf{A}, \mathbf{E})$, we inspect the spectrum of the pencil $(\mathbf{E}, \mathbf{A})$. Hence, one quantity of interest is represented by the eigenvalues at 0 (instead of the eigenvalues at infinity). From a numerical point of view, this leads to a more stable procedure. The set of interest, denoted with Γ, contains the eigenvalues at 0. Let $|\Gamma| = k$.

Nevertheless, it is not straightforward to accurately identify the set Γ (as it will be made clear by the numerical examples). Since the pencil has singularities, the true zeros might get mixed with such singularities in the classification process. Hence, a reliable choice of k (that denotes the number of zeros) is sometimes not straightforward to find.

In what follows, we present the steps of the proposed procedure.

1). Initialization step: start with system matrices $(\mathbf{C}, \mathbf{E}, \mathbf{A}, \mathbf{B})$ that form a realization of a linear descriptor system as in (1). Then, let $\mathbf{C}_1 = \mathbf{C}$, $\mathbf{E}_1 = \mathbf{E}$, $\mathbf{A}_1 = \mathbf{A}$ and $\mathbf{B}_1 = \mathbf{B}$.

2). Perform a QZ (generalized Schur) decomposition of $(\mathbf{E}_1, \mathbf{A}_1)$:

$$\mathbf{Q}^d(\lambda\mathbf{A}_1 - \mathbf{E}_1)\mathbf{Z}^d = \lambda\mathbf{A}_1^d - \mathbf{E}_1^d, \tag{28}$$

so that the elements of the vector $\begin{bmatrix}\gamma_1^d & \gamma_2^d & \dots & \gamma_n^d\end{bmatrix} \in \mathbb{R}_+ \cup \{0\}$ are sorted in a *descending* order, i.e., $\gamma_1^d \geqslant \gamma_2^d \geqslant \cdots \geqslant \gamma_n^d$, where $\gamma_i^d = \frac{|\alpha_i^d|}{|\beta_i^d|+\tau}$. Here, $\alpha_i^d \in \mathbb{C}$ and $\beta_i^d \in \mathbb{C}$ represent the (i, i) entries of matrices $\mathbf{E}_1^d$ and, respectively, of $\mathbf{A}_1^d$. Also, $\tau \in (0, 1)$ is a small positive real number; in practical applications, it is chosen as the *eps* value in Matlab—the machine precision $\approx\ 2.2204 \cdot 10^{-16}$. The purpose of using τ is to avoid division by 0. Note that, in (28), matrices $\mathbf{E}_1^d, \mathbf{A}_1^d \in \mathbb{C}^{n\times n}$ are upper triangular, and $\mathbf{Q}^d, \mathbf{Z}^d \in \mathbb{C}^{n\times n}$ are unitary.

Next, for $k \in \mathbb{N}$, we split the decomposition into two parts corresponding to blocks of dimension $n - k$, and respectively, k (with eigenvalues in the set Γ), as follows:

$$\mathbf{Q}^d(\lambda\mathbf{A}_1 - \mathbf{E}_1)\mathbf{Z}^d = \begin{bmatrix}\mathbf{Q}_1^d \\ \mathbf{Q}_2^d\end{bmatrix}(\lambda\mathbf{A}_1 - \mathbf{E}_1)\begin{bmatrix}\mathbf{Z}_1^d & \mathbf{Z}_2^d\end{bmatrix} = \begin{bmatrix}\lambda\mathbf{A}_{11}^d - \mathbf{E}_{11}^d & \lambda\mathbf{A}_{12}^d - \mathbf{E}_{12}^d \\ \mathbf{0} & \lambda\mathbf{A}_{22}^d - \mathbf{E}_{22}^d\end{bmatrix}, \tag{29}$$

where $\mathbf{E}_{11}^d, \mathbf{A}_{11}^d \in \mathbb{C}^{(n-k)\times(n-k)}$, $\mathbf{E}_{22}^d, \mathbf{A}_{22}^d \in \mathbb{C}^{k\times k}$ and $\mathbf{E}_{12}^d, \mathbf{A}_{12}^d \in \mathbb{C}^{(n-k)\times k}$. Additionally, $(\mathbf{Q}_1^d)^*, \mathbf{Z}_1^d \in \mathbb{C}^{n\times(n-k)}$, $(\mathbf{Q}_2^d)^*, \mathbf{Z}_2^d \in \mathbb{C}^{n\times k}$ are unitary matrices.

Perform another QZ (generalized Schur) decomposition of the pencil $(\mathbf{E}_1, \mathbf{A}_1)$:

$$\mathbf{Q}^a(\lambda\mathbf{A}_1 - \mathbf{E}_1)\mathbf{Z}^a = \lambda\mathbf{A}_1^a - \mathbf{E}_1^a, \tag{30}$$

so that, this time, the elements of the vector $\begin{bmatrix}\gamma_1^a & \gamma_2^a & \dots & \gamma_n^a\end{bmatrix} \in \mathbb{R}_+ \cup \{0\}$ are sorted in an *ascending* order, i.e., $\gamma_1^a \leqslant \gamma_2^a \leqslant \cdots \leqslant \gamma_n^a$, where $\gamma_i^a = \frac{|\alpha_i^a|}{|\beta_i^a|+\tau}$. As before, $\alpha_i^a \in \mathbb{C}$ and $\beta_i^a \in \mathbb{C}$ are the (i, i) entries of matrices $\mathbf{E}_1^a$, and of $\mathbf{A}_1^a$. The positive constant τ is chosen as before. Note that, in (30), matrices $\mathbf{E}_1^a, \mathbf{A}_1^a$ are upper triangular while $\mathbf{Q}^a, \mathbf{Z}^a$ are unitary.

Next, use the same value $k \in \mathbb{N}$ to split the decomposition into two parts corresponding to blocks of dimension k (with eigenvalues in the set Γ), and respectively $n - k$, as follows:

$$\mathbf{Q}^a(\lambda\mathbf{A}_1 - \mathbf{E}_1)\mathbf{Z}^a = \begin{bmatrix}\mathbf{Q}_1^a \\ \mathbf{Q}_2^a\end{bmatrix}(\lambda\mathbf{A}_1 - \mathbf{E}_1)\begin{bmatrix}\mathbf{Z}_1^a & \mathbf{Z}_2^a\end{bmatrix} = \begin{bmatrix}\lambda\mathbf{A}_{11}^a - \mathbf{E}_{11}^a & \lambda\mathbf{A}_{12}^a - \mathbf{E}_{12}^a \\ \mathbf{0} & \lambda\mathbf{A}_{22}^a - \mathbf{E}_{22}^a\end{bmatrix}, \tag{31}$$

where $\mathbf{E}_{11}^a, \mathbf{A}_{11}^a \in \mathbb{C}^{k\times k}$, $\mathbf{E}_{22}^a, \mathbf{A}_{22}^a \in \mathbb{C}^{(n-k)\times(n-k)}$ and $\mathbf{E}_{12}^a, \mathbf{A}_{12}^a \in \mathbb{C}^{k\times(n-k)}$. Additionally, $(\mathbf{Q}_1^a)^*, \mathbf{Z}_1^a \in \mathbb{C}^{n\times k}$, $(\mathbf{Q}_2^a)^*, \mathbf{Z}_2^a \in \mathbb{C}^{n\times(n-k)}$ are unitary matrices.

3). Proceed by putting together the transformation matrices

$$\mathbf{U} = \begin{bmatrix}\mathbf{Q}_2^a \\ \mathbf{Q}_2^d\end{bmatrix}^*, \qquad \mathbf{T} = \begin{bmatrix}\mathbf{Z}_1^d & \mathbf{Z}_1^a\end{bmatrix}. \tag{32}$$

It follows that the matrix pencil $\lambda\mathbf{A}_1 - \mathbf{E}_1$ can be block diagonalized as

$$\begin{aligned}\mathbf{U}^*(\lambda\mathbf{A}_1 - \mathbf{E}_1)\mathbf{T} &= \begin{bmatrix}\mathbf{Q}_2^a\\ \mathbf{Q}_2^d\end{bmatrix}(\lambda\mathbf{A}_1 - \mathbf{E}_1)\begin{bmatrix}\mathbf{Z}_1^d & \mathbf{Z}_1^a\end{bmatrix}\\ &= \begin{bmatrix}\lambda\mathbf{Q}_2^a\mathbf{A}_1\mathbf{Z}_1^d - \mathbf{Q}_2^a\mathbf{E}_1\mathbf{Z}_1^d & \lambda\mathbf{Q}_2^a\mathbf{A}_1\mathbf{Z}_1^a - \mathbf{Q}_2^a\mathbf{E}_1\mathbf{Z}_1^a\\ \lambda\mathbf{Q}_2^d\mathbf{A}_1\mathbf{Z}_1^d - \mathbf{Q}_2^d\mathbf{E}_1\mathbf{Z}_1^d & \lambda\mathbf{Q}_2^d\mathbf{A}_1\mathbf{Z}_1^a - \mathbf{Q}_2^d\mathbf{E}_1\mathbf{Z}_1^a\end{bmatrix}\\ &= \begin{bmatrix}\lambda\mathbf{Q}_2^a\mathbf{A}_1\mathbf{Z}_1^d - \mathbf{Q}_2^a\mathbf{E}_1\mathbf{Z}_1^d & \mathbf{0}\\ \mathbf{0} & \lambda\mathbf{Q}_2^d\mathbf{A}_1\mathbf{Z}_1^a - \mathbf{Q}_2^d\mathbf{E}_1\mathbf{Z}_1^a\end{bmatrix}.\end{aligned}$$

Put together the transformed model as follows:

$$\mathbf{E}_2 = \mathbf{U}^*\mathbf{E}_1\mathbf{T},\ \mathbf{A}_2 = \mathbf{U}^*\mathbf{A}_1\mathbf{T},\ \mathbf{B}_2 = \mathbf{U}^*\mathbf{B}_1 \text{ and } \mathbf{C}_2 = \mathbf{C}_1\mathbf{T}.$$

4). Split the transformed linear model into two blocks:

$$\mathbf{E}_2 = \begin{bmatrix}\hat{\mathbf{E}}_2 & \mathbf{0}\\ \mathbf{0} & \tilde{\mathbf{E}}_2\end{bmatrix},\ \mathbf{A}_2 = \begin{bmatrix}\hat{\mathbf{A}}_2 & \mathbf{0}\\ \mathbf{0} & \tilde{\mathbf{A}}_2\end{bmatrix},\ \mathbf{B}_2 = \begin{bmatrix}\hat{\mathbf{B}}_2\\ \tilde{\mathbf{B}}_2\end{bmatrix},\ \mathbf{C}_2 = \begin{bmatrix}\hat{\mathbf{C}}_2 & \tilde{\mathbf{C}}_2\end{bmatrix}. \tag{33}$$

Use the second blocks to compute the polynomial coefficients:

$$p_\ell = -\tilde{\mathbf{C}}_2(\tilde{\mathbf{A}}_2^{\#}\tilde{\mathbf{E}}_2)^\ell\tilde{\mathbf{A}}_2^{\#}\tilde{\mathbf{B}}_2,\ \text{for } \ell \geqslant 0, \tag{34}$$

where $\tilde{\mathbf{A}}_2^{\#}$ represents the Moore–Penrose inverse of $\tilde{\mathbf{A}}_2$.

Remark 1 Note that the procedure described above represents a direct extension of the ADTF2 algorithm from [30] to the case of singular pencils.

Remark 2 The values α_i^d and β_i^d that appear in the second step of the proposed procedure are indeed of relevance. The reason is that they indicate the blocks corresponding to singularities. The proposed procedure could be altered so that the criterion for constructing the set Γ changes. For example, instead of computing the values γ_i^d, one can solely analyze the values α_i^d and β_i^d.

Example 1 We choose matrices $(\mathbf{C}, \mathbf{E}, \mathbf{A}, \mathbf{B})$ that form a non-minimal linear descriptor realization of the transfer function $\mathbf{H}(s) = 2s + 3 + \frac{4}{s-5}$, i.e.,

$$\mathbf{E} = \begin{bmatrix}-\frac{35}{18} & -\frac{52}{27} & -\frac{50}{27} & -\frac{14}{9}\\ -\frac{27}{14} & -\frac{40}{21} & -\frac{38}{21} & -\frac{10}{7}\\ -\frac{15}{8} & -\frac{11}{6} & -\frac{5}{3} & -1\\ -\frac{7}{4} & -\frac{5}{3} & -\frac{4}{3} & 0\end{bmatrix},\ \mathbf{A} = \begin{bmatrix}\frac{203}{18} & \frac{199}{27} & \frac{47}{27} & -\frac{7}{9}\\ \frac{103}{14} & \frac{73}{21} & -\frac{43}{21} & -\frac{29}{7}\\ \frac{13}{8} & -\frac{13}{6} & -\frac{22}{3} & -8\\ -\frac{7}{4} & -\frac{16}{3} & -\frac{29}{3} & -7\end{bmatrix}, \tag{35}$$
$$\mathbf{B} = \begin{bmatrix}-\frac{49}{9} & -\frac{11}{7} & 4 & 7\end{bmatrix}^T,\ \mathbf{C} = \begin{bmatrix}-\frac{7}{2} & \frac{1}{3} & \frac{17}{3} & 7\end{bmatrix}.$$

Note that the pencil $\lambda\mathbf{E} - \mathbf{A}$ is singular. Hence, the transfer function $\mathbf{H}$ of the system $(\mathbf{C}, \mathbf{E}, \mathbf{A}, \mathbf{B})$ is computed as in (4), i.e., $\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{\#}\mathbf{B}$.

Additionally, note that $\text{rank}(E) = 2$, while $\text{rank}(A) = 3$. Hence, the pencil $(\mathbf{E}, \mathbf{A})$ has only one nonzero finite eigenvalue at $\frac{1}{5}$, two eigenvalues at 0 and one singularity. We proceed with the proposed splitting method by first computing the QZ factorization corresponding to the descending ordering. For example, the values α_i^d, β_i^d, and γ_i^d are presented in Table 1.

Table 1 The diagonal entries α_i^d, β_i^d and coefficients γ_i^d for the descending order

α_i^d	β_i^d	γ_i^d
$3.9497 \cdot 10^{-1}$	1.9749	0.2
$1.9403 \cdot 10^{-15}$	$1.0792 \cdot 10^{-14}$	$1.7616 \cdot 10^{-1}$
$3.2466 \cdot 10^{-8}$	4.5442	$7.1445 \cdot 10^{-9}$
$5.0423 \cdot 10^{-8}$	7.0576	$7.1445 \cdot 10^{-9}$

Also, compute the QZ factorization corresponding to the ascending ordering. Next, for $k = 2$, we identify the matrices corresponding to the block diagonalized matrices $\mathbf{E}_2 = \begin{bmatrix} \hat{\mathbf{E}}_2 & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{E}}_2 \end{bmatrix}$ and $\mathbf{A}_2 = \begin{bmatrix} \hat{\mathbf{A}}_2 & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{A}}_2 \end{bmatrix}$ in the fourth step:

$$\hat{\mathbf{E}}_2 = 10^{-2} \begin{bmatrix} -1.8221 & -3.2120 \\ 7.5665 & 13.3376 \end{bmatrix}, \qquad \tilde{\mathbf{E}}_2 = \begin{bmatrix} -1.535910^{-8} & 6.6561 \cdot 10^{-1} \\ 9.8847 \cdot 10^{-16} & -1.9886 \cdot 10^{-8} \end{bmatrix},$$

$$\hat{\mathbf{A}}_2 = 10^{-1} \begin{bmatrix} -0.9110 & -1.6060 \\ 3.7832 & 6.688 \end{bmatrix}, \quad \tilde{\mathbf{A}}_2 = \begin{bmatrix} 2.1498 & -1.6813 \\ -1.2846 \cdot 10^{-7} & 2.7835 \end{bmatrix}.$$

By applying the formulas in (34), we correctly identify $p_0 = 3$, $p_1 = 2$, and $p_\ell = 0$ for $\ell \geqslant 2$. Next, we vary the value of k between 1 and 4 and collect the estimated polynomial coefficients for these four values. The results are presented in Table 2. Note that, for both $k \in \{2, 3\}$, the original coefficients are recovered.

4 Data-driven splitting of the strictly proper and polynomial part

4.1 The classical Loewner approach

In many situations, a model for the descriptor system $\boldsymbol{\Sigma}$ is not explicitly available; the system matrix realization in (1) can not be directly obtained. Instead of the system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{E}$, a black box is provided that produces input-output measurements. These situations include VLSI modeling from chips or real-time simulation of multi-body dynamics with constraints. The Loewner framework is used to construct a reduced-order system by means of the measured data. In this context, data consist of samples of the underlying transfer function $\mathbf{H}(s)$ at selected interpolation points (in some cases on the imaginary axis, when approximating the frequency response of the original system). A Loewner matrix pencil $\lambda\mathbb{L} - \mathbb{L}_s$ is composed of the Loewner matrix $\mathbb{L}$ and of the shifted Loewner matrix $\mathbb{L}_s$.

Table 2 The estimated polynomial coefficients

	p_0	p_1	p_2
$k = 4$	2.2	1.84	0.032
$k = 3$	3	2	0
$k = 2$	3	2	0
$k = 1$	$-\ 1.3997 \cdot 10^8$	1.0770	$-\ 8.2874 \cdot 10^{-9}$

The Loewner framework originates from rational approximation theory (see [6, 34]) and can be used to recover or reduce the original dynamical system by means of interpolating the transfer function. For simplicity, assume that the data set contains an even number of measurements denoted with $2N$. The next step is to partition the data measurements set into two disjoint data subsets, i.e., the left subset and the right subset as described in the following

$$\text{Left set: } (\mu_i, \mathbf{v}_i),\ i = 1, \ldots, N, \quad \text{Right set: } (\lambda_j, \mathbf{w}_j),\ j = 1, \ldots, N. \tag{36}$$

In (36), consider as given N distinct left interpolation points $\{\mu_i\}_{i=1}^N \subset \mathbb{C}$ and N left sample values: $\{\mathbf{v}_i\}_{i=1}^N \subset \mathbb{C}$. Additionally, consider N distinct right interpolation points $\{\lambda_i\}_{i=1}^N \subset \mathbb{C}$ and N right sample values $\{\mathbf{w}_i\}_{i=1}^N \subset \mathbb{C}$. Note that the sets of left and right interpolation points are assumed distinct, i.e., $\mu_i \neq \lambda_j,\ \forall i, j = 1, \ldots, N$.

We seek a linear descriptor system $\hat{\boldsymbol{\Sigma}}$ such that the associated transfer function $\hat{\mathbf{H}}(s)$ is an interpolant to the original one, $\mathbf{H}(s)$. More precisely, the following interpolation conditions should then hold:

$$\hat{\mathbf{H}}(\mu_i) = \mathbf{H}(\mu_i) = \mathbf{v}_i, \text{ and } \hat{\mathbf{H}}(\lambda_j) = \mathbf{H}(\lambda_j) = \mathbf{w}_j, \text{ for } \quad i, j = 1, \ldots, N. \tag{37}$$

Next, arrange the measured data into matrix format as follows:

$$\begin{aligned} \mathbf{M} &= \operatorname{diag}([\mu_1, \mu_2, \ldots, \mu_N]) \in \mathbb{C}^{N\times N}, & \Lambda &= \operatorname{diag}([\lambda_1, \lambda_2, \ldots, \lambda_N]) \in \mathbb{C}^{N\times N}, \\ \mathbb{V}^T &= \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_N \end{bmatrix} \in \mathbb{C}^{1\times N}, & \mathbb{W} &= \begin{bmatrix} \mathbf{w}_1 & \mathbf{w}_2 & \cdots & \mathbf{w}_N \end{bmatrix} \in \mathbb{C}^{1\times N}. \end{aligned} \tag{38}$$

The Loewner matrix $\mathbb{L} \in \mathbb{C}^{N\times N}$ and the shifted Loewner matrix $\mathbb{L}_s \in \mathbb{C}^{N\times N}$ are defined as follows:

$$\mathbb{L} = \begin{bmatrix} \frac{\mathbf{v}_1-\mathbf{w}_1}{\mu_1-\lambda_1} & \cdots & \frac{\mathbf{v}_1-\mathbf{w}_N}{\mu_1-\lambda_N} \\ \vdots & \ddots & \vdots \\ \frac{\mathbf{v}_N-\mathbf{w}_1}{\mu_N-\lambda_1} & \cdots & \frac{\mathbf{v}_N-\mathbf{w}_N}{\mu_N-\lambda_N} \end{bmatrix}, \quad \mathbb{L}_s = \begin{bmatrix} \frac{\mu_1\mathbf{v}_1-\lambda_1\mathbf{w}_1}{\mu_1-\lambda_1} & \cdots & \frac{\mu_1\mathbf{v}_1-\lambda_N\mathbf{w}_N}{\mu_1-\lambda_N} \\ \vdots & \ddots & \vdots \\ \frac{\mu_N\mathbf{v}_N-\lambda_1\mathbf{w}_1}{\mu_N-\lambda_1} & \cdots & \frac{\mu_N\mathbf{v}_N-\lambda_N\mathbf{w}_N}{\mu_N-\lambda_N} \end{bmatrix}, \tag{39}$$

and are the solutions of the following Sylvester equation

$$\begin{aligned} \mathbf{M}\mathbb{L} - \mathbb{L}\Lambda &= \mathbb{V}\mathbf{R} - \mathbf{L}\mathbb{W}, \\ \mathbf{M}\mathbb{L}_s - \mathbb{L}_s\Lambda &= \mathbf{M}\mathbb{V}\mathbf{R} - \mathbf{L}\mathbb{W}\Lambda, \end{aligned} \tag{40}$$

where $\mathbf{L}^T = \mathbf{R} = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \in \mathbb{C}^{1\times N}$.

After rearranging the data into matrix format, we construct with basically no computational cost a Loewner linear descriptor model described by the following realization $\boldsymbol{\Sigma}_L = (\mathbb{W}, -\mathbb{L}, -\mathbb{L}_s, \mathbb{V})$.

We distinguish two cases:

1. The *right* amount of data is available—in this case, one can exactly recover the original system;
2. A *redundant* amount of data is provided—in this case, one can approximate the original system.

In the first case, the matrix pencil $\lambda\mathbb{L} - \mathbb{L}_s$ is regular. Then, the transfer function of the Loewner model is $\hat{\mathbf{H}}(s) = -\mathbb{W}(s\mathbb{L} - \mathbb{L}_s)^{-1}\mathbb{V}$. This situation is encountered when the number of measurements is just enough to recover the original system (as explained in [34], one would require $2n$ measurements to recover a system of order n).

In the second case, the pencil $\lambda\mathbb{L} - \mathbb{L}_s$ is singular. Then, the definition of the transfer function is adapted as in [2], i.e., $\hat{\mathbf{H}}(s) = -\mathbb{W}(s\mathbb{L} - \mathbb{L}_s)^{\#}\mathbb{V}$.

In real-world applications, the reason behind the singularity of the matrix pencil $\lambda\mathbb{L} - \mathbb{L}_s$ is that data contain, in most cases, redundant or inaccurate measurements.

We focus on the second case, which is more relevant in practice. In this case, since the Loewner pencil is singular, we can use the singular value decomposition (SVD) to compress the data. The truncation order r can be chosen by inspecting the singular value decay of the data. Deciding on a specific value of r is usually done by making a trade-off between reduced-order model dimension and accuracy of fit. Consider the (short) SVDs of augmented Loewner matrices

$$[\mathbb{L} \quad \mathbb{L}_s] = \mathbf{Y}_1\mathbf{S}_1\mathbf{X}_1^*, \quad \begin{bmatrix} \mathbb{L} \\ \mathbb{L}_s \end{bmatrix} = \mathbf{Y}_2\mathbf{S}_2\mathbf{X}_2^*, \tag{41}$$

where $\mathbf{S}_1 \in \mathbb{R}^{N\times N}$, $\mathbf{S}_2 \in \mathbb{R}^{N\times N}$, $\mathbf{Y}_1, \mathbf{X}_2 \in \mathbb{C}^{N\times N}$. The singular values are nonnegative real scalars that give information on the numerical rank of matrices and are stored on the main diagonal of matrices $\mathbf{S}_1$ and $\mathbf{S}_2$ as $\mathbf{S}_1 = \mathrm{diag}([\sigma_1^{(1)}, \sigma_2^{(1)}, \ldots, \sigma_N^{(1)}])$ and $\mathbf{S}_2 = \mathrm{diag}([\sigma_1^{(2)}, \sigma_2^{(2)}, \ldots, \sigma_N^{(2)}])$.

The matrices $\mathbf{Y}, \mathbf{X} \in \mathbb{C}^{N\times r}$ are obtained by selecting the first r columns of the matrices $\mathbf{Y}_1$ and $\mathbf{X}_2$. The reduced Loewner system is constructed by projecting the matrices $\mathbb{L}, \mathbb{L}_s, \mathbb{V}, \mathbb{W}$ with $\mathbf{Y}^*$ and $\mathbf{X}$ to the left and respectively to the right as

$$\hat{\mathbb{L}} = \mathbf{Y}^*\mathbb{L}\mathbf{X}, \quad \hat{\mathbb{L}}_s = \mathbf{Y}^*\mathbb{L}_s\mathbf{X}, \quad \hat{\mathbb{V}} = \mathbf{Y}^*\mathbb{V}, \quad \hat{\mathbb{W}} = \mathbb{W}\mathbf{X}. \tag{42}$$

The projected Loewner model is given by $\hat{\boldsymbol{\Sigma}}_L = (\hat{\mathbb{W}}, -\hat{\mathbb{L}}, -\hat{\mathbb{L}}_s, \hat{\mathbb{V}})$. By means of compression, the pencil $\lambda\, \hat{\mathbb{L}} - \hat{\mathbb{L}}_s$ does not have any infinite eigenvalues. Hence, $\hat{\boldsymbol{\Sigma}}_L$ is a strictly proper system (its transfer function does not have a polynomial part). Hence, we can say that, in the modeling stage, the behavior at infinity of the underlying system is generically lost. We will address this issue in the next sections.

One can construct transformation matrices $\mathbf{U}_\infty$ and $\mathbf{V}_\infty$ as described in Section 3 that are used to block diagonalize the pencil $\lambda\mathbb{L} - \mathbb{L}_s$. The recovered polynomial coefficients can be computed similarly as introduced in Section 2.2.2, by replacing the inversion operator with the Moore–Penrose inversion operator, as follows

$$p_i = \mathbb{W}\mathbf{V}_\infty\Big((\mathbf{U}_\infty^*\mathbb{L}_s\mathbf{V}_\infty)^{\#}(\mathbf{U}_\infty^*\mathbb{L}\mathbf{V}_\infty)\Big)^i (\mathbf{U}_\infty^*\mathbb{L}_s\mathbf{V}_\infty)^{\#}\mathbf{U}_\infty^*\mathbb{V}, \ i \geqslant 0. \tag{43}$$

4.2 The main procedure

In the case of redundant data, significantly large amount of measurements are used in the modeling step. In this case, the matrix pencil $\lambda\mathbb{L} - \mathbb{L}_s$ is singular. Instead of compressing the raw Loewner model as described in (42), we can alternatively first perform a post-processing procedure to the data in order to extract the required information.

The main procedure which adapts the Loewner framework to descriptor systems by preserving the DAE structure is given in Algorithm 1.

Algorithm 1 The new modified Loewner procedure for preserving the DAE structure in data - Loewner$^{\infty}$.

```
Input: A data set composed of sample points and values of the transfer function
    evaluated at these points which is partitioned into two subsets:
        $\begin{array}{l} \text{left}: (\mu_i, \mathbf{v}_i), \;\; i = 1, \ldots, N, \\ \text{right}: \;(\lambda_j, \mathbf{w}_j), \;\; j = 1, \ldots, N. \end{array}$    (44)
1: Rearrange the data into matrix format and construct a raw (possibly) singular
   Loewner model $\boldsymbol{\Sigma}_L$ given by $(\mathbb{W}, -\mathbb{L}, -\mathbb{L}_s, \mathbb{V})$ as in (38–39).
2: Perform a generalized block diagonalization as in Section 3. If the Loewner pencil
   is regular, use methods from Section 3.1. If the pencil is singular, use the
   procedure from Section 3.2.1 for the quadruple $(\mathbb{W}, -\mathbb{L}, -\mathbb{L}_s, \mathbb{V})$ .
3: Set up a trust interval $\mathcal{I} = [a, b] \subseteq [1, N]$. For each index $k_0 \in \mathcal{I}$, the estimated
   polynomial coefficients are computed.
4: Fix one value $k \in \mathcal{I}$ for the next steps. For example, one can choose the center
   of the interval $\mathcal{I}$. Here, $k$ represents the estimated dimension of the subsystem
   corresponding to the polynomial part.
5: Denote with $\{p_1, p_2, \ldots, p_h\}$ the polynomial coefficients corresponding to the
   chosen index $k$. The value $h \in \mathbb{N}$ is the maximum integer so that $|p_i| > \rho$ for all
   $0 \leqslant i \leqslant h$ (here $\rho > 0$ is a small tolerance value).
6: For all values of $\omega \in \{\mu_i | 1 \leqslant i \leqslant N\} \cup \{\lambda_j | 1 \leqslant j \leqslant N\}$ (or for another
   frequency range), subtract the polynomial part $\mathbf{H}_{\text{poly}}(\omega) = \sum_{i=0}^{g} p_i \omega^i$ from the
   original measurements. The post-processed data set can be written as
        $\begin{array}{l} \text{left}: (\mu_i, \mathbf{v}_i - \mathbf{H}_{\text{poly}}(\mu_i)), \;\; i = 1, \ldots, N, \\ \text{right}: \;(\lambda_j, \mathbf{w}_j - \mathbf{H}_{\text{poly}}(\lambda_j)), \;\; j = 1, \ldots, N. \end{array}$    (45)
7: Use the new modified measurements points and values in (45) to construct
   another Loewner model $\overline{\boldsymbol{\Sigma}}_L$ of dimension N, given by $(\overline{\mathbb{W}}, -\overline{\mathbb{L}}, -\overline{\mathbb{L}}_s, \overline{\mathbb{V}})$.
8: Project the system $\overline{\boldsymbol{\Sigma}}_L$ as in (42) and construct a new lower dimensional Loewner
   model $\hat{\boldsymbol{\Sigma}}_L$ (let $r$ be its dimension).
9: Reattach a minimal realization $\boldsymbol{\Sigma}_{\text{poly}}$ of the polynomial part $\mathbf{H}_{\text{poly}}(s)$ to the
   system $\hat{\boldsymbol{\Sigma}}_L$ and obtain the final version of the post-processed Loewner model
   (denoted with $\hat{\boldsymbol{\Sigma}}_L^{\text{new}}$).
Output: The reduced-order Loewner model $\hat{\boldsymbol{\Sigma}}_L^{\text{new}}$ via the new proposed procedure
    Loewner$^{\infty}$.
```

Lemma 2 *The transfer function of the Loewner model* $\overline{\boldsymbol{\Sigma}}_L$ *in Step 7, computed as* $\overline{\mathbb{W}}(\overline{\mathbb{L}}_s - s\overline{\mathbb{L}})^{\#}\overline{\mathbb{V}}$*, can be written in terms of the transfer function of the Loewner model* $\boldsymbol{\Sigma}_L$ *in Step 1, i.e.,* $\mathbb{W}(\mathbb{L}_s - s\mathbb{L})^{\#}\mathbb{V}$*, as follows:*

$$\overline{\mathbb{W}}(\overline{\mathbb{L}}_s - s\overline{\mathbb{L}})^{\#}\overline{\mathbb{V}} = \mathbb{W}(\mathbb{L}_s - s\mathbb{L})^{\#}\mathbb{V} - \mathbf{H}_{poly}(s), \tag{46}$$

for all $s \in \{\mu_i | 1 \leqslant i \leqslant N\} \cup \{\lambda_j | 1 \leqslant j \leqslant N\}$.

Proof For $i, j \in \{1, \ldots, N\}$, let $\overline{\mathbf{v}}_i = \mathbf{v}_i - \mathbf{H}_{poly}(\mu_i)$, and $\overline{\mathbf{w}}_j = \mathbf{w}_j - \mathbf{H}_{poly}(\lambda_j)$. Then, consider the following modified measurement vectors:

$$\overline{\mathbb{V}}^T = \left[\overline{\mathbf{v}}_1\ \overline{\mathbf{v}}_2\ \cdots\ \overline{\mathbf{v}}_N\right] \in \mathbb{C}^{1\times N}, \quad \overline{\mathbb{W}} = \left[\overline{\mathbf{w}}_1\ \overline{\mathbf{w}}_2\ \cdots\ \overline{\mathbf{w}}_N\right] \in \mathbb{C}^{1\times N},$$

the Loewner matrices corresponding to the data in (45) constructed as in (39):

$$\overline{\mathbb{L}}_{i,j} = \frac{\overline{\mathbf{v}}_i - \overline{\mathbf{w}}_j}{\mu_i - \lambda_j}, \quad \overline{\mathbb{L}}_{s\,i,j} = \frac{\mu_i\overline{\mathbf{v}}_i - \lambda_j\overline{\mathbf{w}}_j}{\mu_i - \lambda_j},$$

and the matrices $\mathbf{M}, \Lambda, \mathbb{L}, \mathbb{R}$ defined as before. The Loewner matrices $\overline{\mathbb{L}}$ and $\overline{\mathbb{L}}_s$ satisfy Sylvester equations as in (40), i.e.,

$$\begin{aligned} \mathbf{M}\overline{\mathbb{L}} - \mathbb{L}\Lambda &= \overline{\mathbb{V}}\mathbf{R} - \mathbf{L}\overline{\mathbb{W}}, \\ \mathbf{M}\overline{\mathbb{L}}_s - \overline{\mathbb{L}}_s\Lambda &= \mathbf{M}\overline{\mathbb{V}}\mathbf{R} - \mathbf{L}\overline{\mathbb{W}}\Lambda. \end{aligned} \tag{47}$$

By multiplying the first Sylvester equation in (47) by s and subtracting it from the second equation in (47), we get

$$\mathbf{M}(\overline{\mathbb{L}}_s - s\overline{\mathbb{L}}) - (\overline{\mathbb{L}}_s - s\overline{\mathbb{L}})\Lambda = (\mathbf{M} - s\mathbf{I})\overline{\mathbf{V}} - \mathbf{LW}(\Lambda - s\mathbf{I}). \tag{48}$$

Next, setting $s = \lambda_j$ and multiplying equation (48) with the jth unit vector $\mathbf{e}_j$ on the right, we write:

$$\begin{aligned} (\mathbf{M} - \lambda_j\mathbf{I})(\overline{\mathbb{L}}_s - \lambda_j\overline{\mathbb{L}})\mathbf{e}_j \ &= \ (\mathbf{M} - \lambda_j\mathbf{I})\overline{\mathbb{V}} \Rightarrow (\overline{\mathbb{L}}_s - \lambda_j\overline{\mathbb{L}})\mathbf{e}_j = \overline{\mathbb{V}} \\ &\Rightarrow \overline{\mathbb{W}}(\overline{\mathbb{L}}_s - \lambda_j\overline{\mathbb{L}})^{\#}\overline{\mathbb{V}} = \overline{\mathbb{W}}\mathbf{e}_j = \overline{w}_j = w_j - \mathbf{H}_{poly}(\lambda_j) \\ &\Rightarrow \overline{\mathbb{W}}(\overline{\mathbb{L}}_s - \lambda_j\overline{\mathbb{L}})^{\#}\overline{\mathbb{V}} = \mathbb{W}(\mathbb{L}_s - \lambda_j\mathbb{L})^{\#}\mathbb{V} - \mathbf{H}_{poly}(\lambda_j). \end{aligned} \tag{49}$$

A similar reasoning is used to show that the equality in (46) also holds for the left interpolation points μ_i. □

Remark 3 A trust interval $\mathcal{I}$ (as used in Step 3 of Algorithm 1) is an interval such that for any $k_1, k_2 \in [a, b]$, the polynomial coefficient estimates corresponding to k_1, i.e., $p_\ell^{(k_1)}$, and the ones corresponding to k_2, i.e., $p_\ell^{(k_2)}$ have similar values, i.e., $|p_\ell^{(k_1)} - p_\ell^{(k_2)}| < \zeta$, where $\zeta > 0$ is a small tolerance value. As an example, in Fig. 7, one can choose the confidence interval to be $\mathcal{I} = [25, 55]$, based on the values of the estimated coefficients.

Remark 4 In Step 4, k is the estimated value denoting the dimension of the subsystem corresponding to the polynomial part. It is to be noted that this subsystem is not necessarily in a minimal realization format. In practice, e.g., the MNA example in Section 5.3.1, this value is chosen to be $k = 17$, although a minimal realization would only be two dimensional.

Remark 5 Descriptor systems appear in practical applications mostly with index $\nu \in \{1, 2, 3\}$ (the polynomial part is up to a quadratic term). Hence, in such cases, the value $h \in \mathbb{N}$ would be less or equal than 2. In all numerical examples in Section 5, we displayed only the first three estimated coefficients, i.e., p_0, p_1, and p_2. For the MNA example, by choosing a tolerance value of $\rho = 10^{-10}$, we included in the calculations only the first 2 polynomial coefficients; hence, $h = 1$ was considered (since $p_2 < \rho$).

Remark 6 The Loewner system $\boldsymbol{\Sigma}_L$ corresponding to the post-processed data in (45) is strictly proper provided that the polynomial coefficients are perfectly estimated (exactly the same as the true coefficients corresponding to the original system $\boldsymbol{\Sigma}$).

Remark 7 Provided that the value h is accurately chosen, it is hence assumed that the polynomial coefficients are given by $p_0, p_1, \dots p_h$ and $p_i = 0$, $\forall i > h$. A minimal linear descriptor realization $\boldsymbol{\Sigma}_{\text{poly}} = (\mathbf{C}_{\text{poly}}, \mathbf{E}_{\text{poly}}, \mathbf{A}_{\text{poly}}, \mathbf{B}_{\text{poly}})$ of order $h+1$ corresponding to the polynomial part $\mathbf{H}_{\text{poly}}(s)$ is constructed as

$$\mathbf{E}_{\text{poly}} = \mathbf{J}_{h+1},\ \mathbf{A}_{\text{poly}} = \mathbf{I}_{h+1},\ \mathbf{B}_{\text{poly}} = [0\,0\ \cdots\ 0\ 1]^T,\ \mathbf{C}_{\text{poly}} = -[p_h\ p_{h-1}\ \cdots\ p_0],$$

where $\mathbf{J}_{h+1}$ is a Jordan block of dimension $(h+1) \times (h+1)$ with 0 eigenvalues, and $\mathbf{I}_{h+1}$ is the identity matrix of the same dimension.

5 Experimental results

In this section, we provide numerical results for various descriptor systems by implementing the algorithms from previous sections. We consider two simple low-dimensional examples as well as three larger scale benchmarks examples.

5.1 A simple example

Consider the simple transfer function $\mathbf{H}(s) = 2s + 2 + \frac{4}{s-5}$, the same as the one mentioned in example 1. First, choose the following interpolation points $\boldsymbol{\mu}^T = \left[-4, -2, 1, 3\right]$ and $\boldsymbol{\lambda} = \left[-3, -1, 2, 4\right]$. It hence follows that the matrices in (35), example 1 is actually Loewner matrices associated with the left and right sampling points and transfer function above, i.e., $\mathbf{E}_1 = -\mathbb{L}$, $\mathbf{A}_1 = -\mathbb{L}_s$.

Next, choose the same interpolation points but ordered differently, as $\boldsymbol{\mu}^T = \left[-1, -2, -3, -4\right]$, and $\boldsymbol{\lambda} = \left[1, 2, 3, 4\right]$. Then, put together the Loewner matrices and restart the procedure for the new matrices $\mathbf{E}_1$ and $\mathbf{A}_1$, where

$$\mathbf{E}_1 = -\mathbb{L} = \begin{bmatrix} -\frac{11}{6} & -\frac{16}{9} & -\frac{5}{3} & -\frac{4}{3} \\ -\frac{13}{7} & -\frac{38}{21} & -\frac{12}{7} & -\frac{10}{7} \\ -\frac{15}{8} & -\frac{11}{6} & -\frac{7}{4} & -\frac{3}{2} \\ -\frac{17}{9} & -\frac{50}{27} & -\frac{16}{9} & -\frac{14}{9} \end{bmatrix}, \quad \mathbf{A}_1 = -\mathbb{L}_s = \begin{bmatrix} -\frac{13}{6} & -\frac{35}{9} & -\frac{16}{3} & -\frac{17}{3} \\ -\frac{2}{7} & -\frac{43}{21} & -\frac{25}{7} & -\frac{29}{7} \\ \frac{13}{8} & -\frac{1}{6} & -\frac{7}{4} & -\frac{5}{2} \\ \frac{32}{9} & \frac{47}{27} & \frac{1}{9} & -\frac{7}{9} \end{bmatrix}.$$

Additionally, consider the vectors $\mathbf{B}_1 = \mathbb{V} = \left[\,\frac{1}{3}\ -\frac{11}{7}\ -\frac{7}{2}\ -\frac{49}{9}\,\right]^T$ and also $\mathbf{C}_1 = \mathbb{W} = \left[\,4\ \frac{17}{3}\ 7\ 7\,\right]$. Applying again the procedure in Section 3.2.1, we start by analyzing the entries of the main diagonal of matrices $\mathbf{E}_1^d$ and $\mathbf{A}_1^d$. Note that, by inspecting Tables 1 and 3, the position of the entries corresponding to the true finite eigenvalue ($\lambda = 0.2$) has shifted from the first to the second place (for the descending ordering).

Table 3 The diagonal entries α_i^d, β_i^d and coefficients γ_i^d for the descending order

α_i^d	β_i^d	γ_i^d
$3.5637 \cdot 10^{-14}$	$1.4881 \cdot 10^{-13}$	$2.3912 \cdot 10^{-1}$
$5.3495 \cdot 10^{-3}$	$2.6747 \cdot 10^{-2}$	0.2
$2.6742 \cdot 10^{-9}$	$2.3409 \cdot 10^{-1}$	$1.1424 \cdot 10^{-8}$
$4.8250 \cdot 10^{-9}$	$4.2237 \cdot 10^{-1}$	$1.1424 \cdot 10^{-8}$

The fact that the singularity corresponding to $\gamma_1 = 2.3912 \cdot 10^{-1}$ is located on the first position for descending ordering (and, respectively on the last for the ascending ordering) influences the accurate recovery of the coefficients. Note that the results in Table 4 are correct only for $k = 2$, while in the previous experiment, they were correct for both $k \in \{2, 3\}$ (see Table 2).

This shows, in particular, the sensitivity of the Loewner framework for different data sets in addition to the numerical sensitivity of the splitting scheme. That is why, for the large-scale examples, we will use so-called trust intervals for the k value. This is done in order to estimate the polynomial coefficients (for different values of k).

Finally, we assume that the sample values of the transfer function $\mathbf{H}(s) = 2s + 3 + \frac{4}{s-5}$, evaluated at $\boldsymbol{\mu}^T \cup \boldsymbol{\lambda}$, are affected by additive noise. We assume the noise level to be 10^{-2} and the noisy values to follow a standard normal distribution. We repeat the experiment described above and compute the estimated polynomial coefficients for $k = 2$ (the value that should provide correct estimates): $p_0 = 2.0623$, $p_1 = 1.8028$, $p_2 = -0.0428$, $p_3 = -0.0042$, and $p_4 = 0.0002$. Hence, we conclude that the polynomial part extraction is indeed sensitive to noise.

5.2 An order $n = 7$ example

Consider a descriptor system of dimension $n = 7$ described by $\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, $\mathbf{y} = \mathbf{C}\mathbf{x}$, where

$$\mathbf{E} = \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0\\ 0&0&1&0&0&0&0\\ 0&0&0&1&0&0&0\\ 0&0&0&0&2&0&0\\ 0&0&0&0&0&3&0\\ 0&0&0&0&0&0&0 \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0&0&0&1&0&0&0\\ 0&0&0&0&1&0&0\\ 0&0&1&0&0&1&0\\ -2&1&0&-2&1&0&-1\\ 1&-2&1&1&-3&1&0\\ 0&1&-2&0&1&-2&1\\ 1&0&-1&0&0&0&0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1\\1\\0\\1\\1\\1\\1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} -1\\0\\1\\-1\\0\\0\\0 \end{bmatrix}. \tag{50}$$

Table 4 The estimated polynomial coefficients

	p_0	p_1	p_2
$k = 4$	2.2	1.84	-0.032
$k = 3$	2.2	1.84	-0.032
$k = 2$	3	2	0
$k = 1$	$-8.7536 \cdot 10^7$	2.4374	$-6.7854 \cdot 10^{-8}$

The transfer function can be explicitly computed as follows:

$$\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} = \frac{3s}{4} + \frac{3}{2} - \frac{5s^3 + 9s^2 + 5s - 2}{8s^4 + 20s^3 + 24s^2 + 12s + 4}. \tag{51}$$

Hence, identify the coefficients of the polynomial part as $p_0 = \frac{3}{2}$, $p_1 = \frac{3}{4}$, and $p_k = 0$ for $k > 1$. Note that the pencil $\lambda\mathbf{E} - \mathbf{A}$ has three eigenvalues at ∞ and four finite eigenvalues given by the two complex conjugate pairs: $-0.9712 \pm 0.8138i$ and $-0.2787 \pm 0.4834i$.

Using the Jordan chain method in Section 3.1.1 for $k = 3$, the transformation matrices $\mathbf{U}_\infty$ and $\mathbf{V}_\infty$ can be explicitly obtained as

$$\mathbf{U}_\infty = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}, \quad \mathbf{V}_\infty = \begin{bmatrix} 0 & 0 & -9 \\ 0 & 0 & 0 \\ 0 & 0 & 3 \\ 0 & 3 & 18 \\ 0 & 0 & -3 \\ 0 & -1 & -2 \\ 1 & 0 & 0 \end{bmatrix}. \tag{52}$$

We can obtain the coefficients of polynomial part as follows:

$$\begin{aligned} p_0 &= -\mathbf{C}\mathbf{V}_\infty(\mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{B} = \tfrac{3}{2}, \\ p_1 &= -\mathbf{C}\mathbf{V}_\infty(\mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{E}\mathbf{V}_\infty(\mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{B} = \tfrac{3}{4}, \\ p_k &= -\mathbf{C}\mathbf{V}_\infty((\mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{E}\mathbf{V}_\infty)^k(\mathbf{U}_\infty^*\mathbf{A}\mathbf{V}_\infty)^{-1}\mathbf{U}_\infty^*\mathbf{B} = 0, \quad k > 1. \end{aligned}$$

Additionally, the ADTF1 method from Section 3.1.2 is applied to the original matrices in (50). The results can be found in Table 5. For $k \in \{0, 1, 2, 3\}$, we perfectly recover the true polynomial coefficients. For $k \in \{4, 5, 6\}$, the recovered values are not accurate.

Finally, in what follows, we apply this framework for data-driven splitting of the polynomial part. We show that the polynomial coefficients can be recovered from measurements of the transfer function. For example, choose 11 left and 11 right sample points on the real axis, collected in the vectors $\boldsymbol{\mu}, \boldsymbol{\lambda}^T \in \mathbb{R}^{11}$,

$$\boldsymbol{\mu}^T = -\tfrac{1}{2}\left[11, 10, \ldots, 1\right], \qquad \boldsymbol{\lambda} = \tfrac{1}{2}\left[0, 1, \ldots, 10\right]. \tag{53}$$

By means of the classical Loewner framework in [6], construct an 11th order system from measurements, characterized by a singular pencil $(\mathbf{A}, \mathbf{E})$. Then apply the proposed splitting technique to separate this system into two subsystems. We denote with k the dimension of the subsystem corresponding to the infinite eigenvalues.

Table 5 The estimated polynomial coefficients using the ADTF1 method in Section 3.1.2

ADTF1	p_0	p_1	p_2
$k = 6$	1.8716 + 0.0500i	−0.8709 − 0.0420i	1.3078 + 1.9765i
$k = 5$	1.4131 − 0.0053i	0.7261 − 0.0077i	0.0548 + 0.028i
$k = 4$	1.4250 − 0.0841i	0.7242 + 0.0212i	0.0515 + 0.0150i
$0 \le k \le 3$	1.5	0.75	0

Table 6 The estimated polynomial coefficients using the procedure in Section 3.2.1

ADTF2	p_0	p_1	p_2
$k = 10$	1.6747 − 0.7382i	−0.5628 + 0.1558i	1.4716 + 2.0918i
$k = 9$	1.3495	0.8744	− 0.0568
$k = 8$	1.4247 + 0.0329i	0.8122 + 0.0182i	− 0.0284 − 0.0426i
$2 \leq k \leq 7$	1.5	0.75	0

We vary this value from 2 to 10. The results are collected in Table 6. Note that for $k \in \{2, 3, \ldots, 7\}$, we perfectly recover the coefficients.

5.3 Descriptor system benchmarks

In this section, we apply the proposed method in Algorithm 1 to the following benchmarks problems: modified nodal analysis model 1 (MNA-1) [18], semi-discretized Oseen equations (Oseen) [26], constrained damped mass-spring system (CDMS system) [35]. The dimension and the index of these systems are collected in Table 7.

In what follows, we apply the proposed procedure in Section 3.2.1 for singular pencils.

5.3.1 MNA1 model

Modified nodal analysis (MNA) is a procedure used in the field of electrical engineering that determines the circuit's node voltages and branch currents. It was developed as a means to overcome the difficulty of representing voltage-defined components in classical nodal analysis (e.g., voltage-controlled sources).

As described in [18], voltage sources are connected to the network ports to obtain the admittance matrix of a multiport. The MNA equations are written as $\mathbf{E}\dot{\mathbf{x}}_n = \mathbf{A}\mathbf{x}_n + \mathbf{B}\mathbf{u}_p, \ \ \mathbf{i}_p = \mathbf{C}\mathbf{x}_n$. The $\mathbf{i}_p$ and $\mathbf{u}_p$ vectors denote the port currents and voltages, respectively, while $\mathbf{v}$ and $\mathbf{i}$ are the MNA variables corresponding to the node voltages, inductor and voltage source currents, respectively. The conductance matrix $\mathbf{A}$ and the susceptance matrix $\mathbf{E}$ are given

$$\mathbf{A} = \begin{bmatrix} -\mathbf{N} & -\mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{H} \end{bmatrix}, \quad \mathbf{x}_n = \begin{bmatrix} \mathbf{v} \\ \mathbf{i} \end{bmatrix}. \tag{54}$$

where $\mathbf{N}$, $\mathbf{L}$, and $\mathbf{H}$ are the matrices containing the resistors, capacitors, and inductors values, respectively. Also, $\mathbf{G}$ consists of 1, − 1, and 0 entries only. In this framework,

Table 7 Properties of the benchmark examples

	MNA-1	CDMS	Oseen
Dimension n	578	1001	1557
Index ν	2	3	2

if the circuit contains passive linear elements only, it follows that the $\mathbf{E}$ matrix is symmetric and positive semidefinite, while $\mathbf{B} = \mathbf{C}^T$.

In this section, we analyze the MNA-1 system of dimension $n = 578$ from [18]. All system matrices of the original system are sparse and the number of inputs and outputs is $m = p = 9$. Next, consider instead a SISO-modified system $(\breve{\mathbf{A}}, \breve{\mathbf{B}}, \breve{\mathbf{C}}, \mathbf{0}, \breve{\mathbf{E}})$ for which the system matrices are written in terms of the original ones, as $\breve{\mathbf{A}} = 10^{-4}\mathbf{A}$, $\breve{\mathbf{E}} = 10^{9}\mathbf{E}$, $\breve{\mathbf{B}} = \mathbf{B}\mathbf{e}_2$, $\breve{\mathbf{C}} = \mathbf{e}_2^T\mathbf{C}$. Basically, we do a scaling of the original pencil $\lambda\mathbf{E} - \mathbf{A}$ and consider only the second input and output. We can directly compute the coefficients of the polynomial part of the transfer function as $p_0^{\text{true}} = 5.5047 \cdot 10^6$, $p_1^{\text{true}} = 2.3010 \cdot 10^3$ and $p_j^{\text{true}} = 0$, $j > 1$. Hence, the index of the system is $\nu = 2$.

First, depict the frequency response of the system in Fig. 1, i.e., the magnitude of the transfer function $\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}$ evaluated on the imaginary axis, in the interval $[10^{-2}, 10^{1}]i$.

By inspecting the frequency response, we observe that the dominant oscillations mainly occur in the frequency range $[10^{-2}, 10^{0}]i$. The next step is to take measurements in a high frequency range and extract the polynomial coefficients from this data.

Consequently, take $N = 40$ measurements in the frequency range $[10^2, 10^6]i$ and construct a real-valued 40th order Loewner model (by including the complex conjugate values in the data set, as performed in [6], page 360). By varying the value of the index k from 1 to 40, compute the values of p_j, $j \in \{0, 1, 2\}$ for each k. Afterwards, choose the trust interval $[10, 25]$ in which the values of the estimated polynomial coefficients have similar values for all k's, as can be observed in Fig. 2.

The relative deviation between the true and estimated coefficients is depicted in Fig. 3 for p_0 and p_1 only. The reason for this is because $p_2^{\text{true}} = 0$.

We select the estimated polynomial coefficient values corresponding to the center of the interval, i.e., to $k = 17$.

Next, proceed with the remaining steps of Algorithm 1. In the next modeling step, we are using a different set of measurements. In order to match the oscillatory behavior of the system in the lower frequency range, select 600 logarithmically spaced points in the interval $[10^{-2}, 10^{2}]i$. Again, construct a real Loewner model of

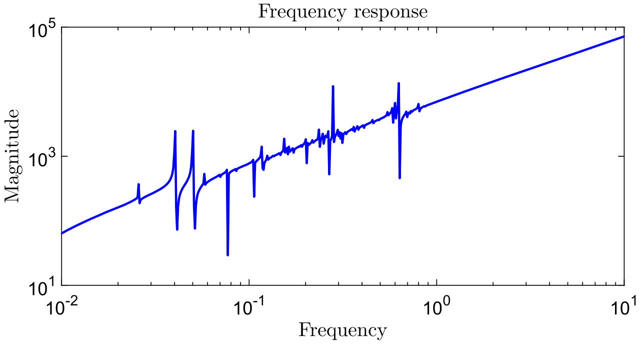

Fig. 1 The frequency response of the descriptor system

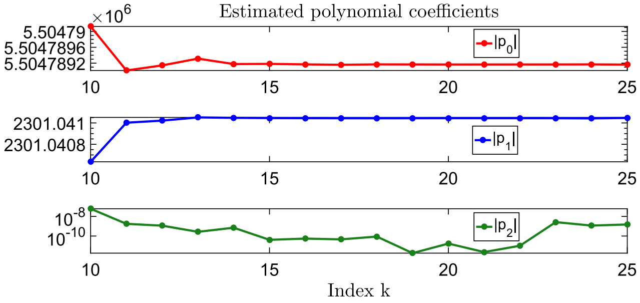

Fig. 2 The estimated polynomial coefficients of the transfer function for $k \in [10, 25]$

dimension 600. The decay of the normalized singular values corresponding to the augmented Loewner matrices is presented in Fig. 4.

Finally, we subtract the estimated polynomial part from this Loewner model, compress the model to order $r = 28$ (corresponding to $\sigma_{28} \approx 10^{-2}$ singular value), and then reattach the polynomial part given by the estimated polynomial coefficients—a minimal realization is considered. The dimension of the resulting system after reattaching the polynomial part is hence $\tilde{r} = 31$ (p_2 is also taken into consideration). This is true for the other experiments in this section (in Sections 5.3.2 and 5.3.3).

The frequency responses of the original large-scale model and of the two reduced-order Loewner models (with and without performing the post-processing step) are presented in Fig. 5. Notice that there is a clear mismatch between the original response and the one produced with the classical Loewner approach (which is visible starting at the frequency value 10^3).

Additionally, the relative approximation error is depicted in Fig. 5. This was computed using the formula $\|\mathbf{H}(i\omega) - \hat{\mathbf{H}}(i\omega)\|/\|\mathbf{H}(i\omega)\|$, for $\omega \in [10^{-2}, 10^6]$. For the two other numerical examples treated in this section, the relative approximation error in the frequency domain is computed using the same formula (but for different frequency ranges).

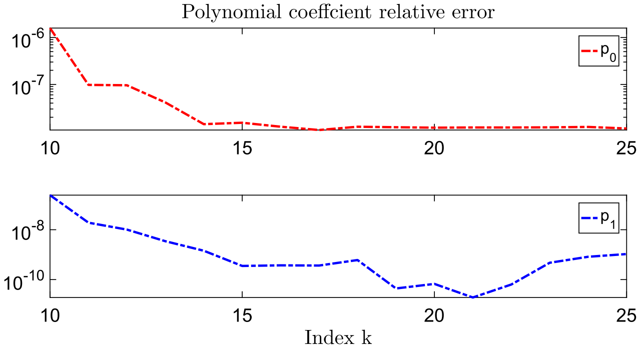

Fig. 3 Deviation between the true and estimated polynomial coefficients p_0 and p_1

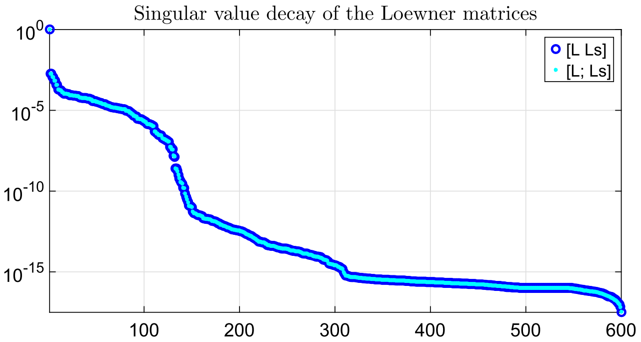

Fig. 4 Singular value decay of the augmented matrices $[\mathbb{L}\ \mathbb{L}_s]$ and $[\mathbb{L}\ ;\ \mathbb{L}_s]$

Note that the advantage of using the new method consists in drastically lowering the deviation in the high frequency range.

5.3.2 Constrained damped mass-spring system

For the second benchmark example, we consider the holonomically constrained damped mass-spring system from [35]. Here, the ith mass of weight m_i is connected to the (i + 1)st mass by a spring and a damper with constants k_i and d_i, respectively, and also to the ground by a spring and a damper with constants ξ_i and δ_i, respectively. Additionally, the first mass is connected to the last one by a rigid bar and it is influenced by the control input signal $\mathbf{u}(t)$. The vibration of this system is described by a descriptor system as in equation (34) from [35].

We consider the same values for the ξ_i, δ_i, k_i, and d_i parameters as in [35] and choose $g = 500$ for the dimension of the variables. Hence, by grouping together the

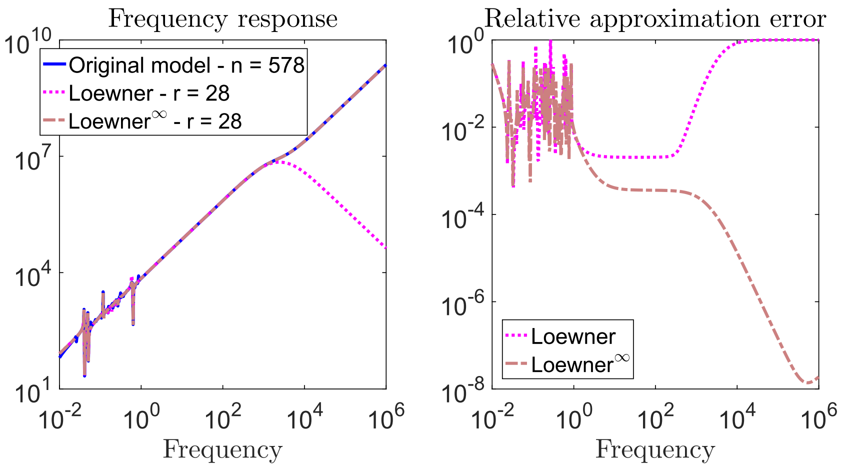

Fig. 5 Frequency response of the original/reduced models + relative approximation error

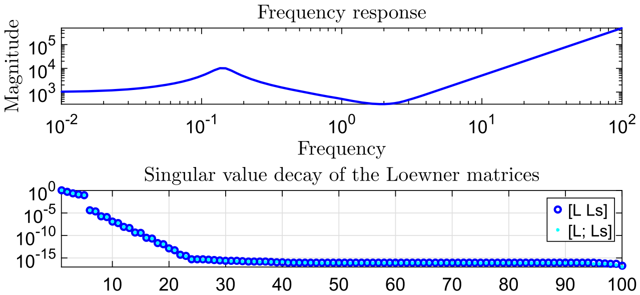

Fig. 6 The frequency response (up) and the decay of singular values (down)

position and the velocity vectors into the new $\mathbf{x}$ variable, we end up with a descriptor system of order $n = 1001$, described by $\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, $\mathbf{y} = \mathbf{C}\mathbf{x}$, where $\mathbf{E}, \mathbf{A} \in \mathbb{R}^{1001\times1001}$ and $\mathbf{B}, \mathbf{C}^T \in \mathbb{R}^{1001}$. As opposed to the original system in [35], where it was considered that $\mathbf{B} = \mathbf{e}_{g+1} \in \mathbb{R}^g$ and $\mathbf{C} = [\mathbf{e}_1\ \mathbf{e}_2\ \mathbf{e}_{g-1}]^T \in \mathbb{R}^{3\times g}$ for $g \geqslant 1$ (hence a system with one input and three outputs), we choose $\mathbf{B} = \mathbf{C}^T = [1\ 1\ \ldots\ 1]^T \in \mathbb{R}^{1001}$ and analyze a SISO descriptor system instead.

For the original 1001th order system constructed as described above, we can directly identify the following coefficients of the polynomial part

$$p_0^{\text{true}} = \frac{23}{8} = 2.875, \quad p_1^{\text{true}} = \frac{15}{2} = 7.5, \quad p_2^{\text{true}} = 50, \text{ and } p_k^{\text{true}} = 0, \text{ for } k > 2.$$

Hence, it follows that the original system is of index $\nu = 3$. Take measurements corresponding to $N = 100$ logarithmically spaced sampling points in the frequency range $[10^{-2}, 10^2]i$. Then, construct a real-valued 100th order Loewner model. The frequency response (restricted to the above sampling interval) of the system is presented in Fig. 6 together with the singular value decay of the augmented Loewner matrices .

Again, we vary the value of the index k from 1 to 100 and compute the values of polynomial coefficients p_j, $j \in \{0, 1, 2\}$ for each k. The results are gathered in Fig. 7. By inspecting this figure, we observe that one can choose, for example, the trust interval [25, 55].

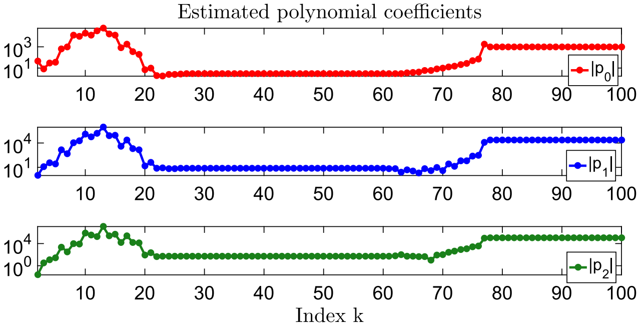

Fig. 7 The estimated polynomial coefficients of the transfer function for $k \in [1, 100]$

Next, Fig. 8 shows the deviation between the true polynomial coefficients and the estimated ones for values of k restricted to the chosen trust interval. Next, we will apply Algorithm 1 for two such values, corresponding to $k_1 = 27$ and $k_2 = 40$. Denote with $p_j^{(\ell)}$ the estimated polynomial coefficients corresponding to index k_ℓ for $\ell \in \{1, 2\}$ and $j \in \{0, 1, 2\}$. The absolute value of the deviation is computed as $\boldsymbol{\epsilon}^{(\ell)} = \left[|p_1^{(\ell)} - p_1^{\text{true}}| \ |p_2^{(\ell)} - p_2^{\text{true}}| \ |p_3^{(\ell)} - p_3^{\text{true}}|\right]$

$$\begin{cases} \boldsymbol{\epsilon}^{(1)} = [1.1600 \ \ 1.6904 \ \ 1.933] \cdot 10^{-2}, \ \text{for } k_1 = 27, \\ \boldsymbol{\epsilon}^{(2)} = [8.4913 \ \ 3.2977 \ \ 3.1873] \cdot 10^{-8}, \ \text{for } k_2 = 40. \end{cases}$$

In what follows, we construct reduced-order models of order $r = 8$ (which corresponds to the singular value $\sigma_8 \approx 10^{-5}$) For each of the values k_1 and k_2, we compute the modified Loewner reduced model of order $r = 8$ via the procedure in Algorithm 1 as well as the reduced model using the classical Loewner model (of the same order). We also calculate the relative approximation error between the frequency response of the original large-scale system and that of the reduced models. The results are depicted in Fig. 9.

As expected, the approximation error is highly dependent on how well the polynomial coefficients are approximated. As it can be observed in the left side of Fig. 9, the approximation error is significantly larger than that in the right side of the figure (when applying Loewner$^\infty$).

5.3.3 Semi-discretized Oseen equations model

In this section, we apply the new proposed method to a descriptor system obtained from semi-discretization of the Oseen equations, as performed and thoroughly explained in [26]. The Oseen equations are related to the linearized Navier–Stokes equations. The study of these equations is of importance in many flow systems where one is interested in the behavior of the linearized flow around a steady state.

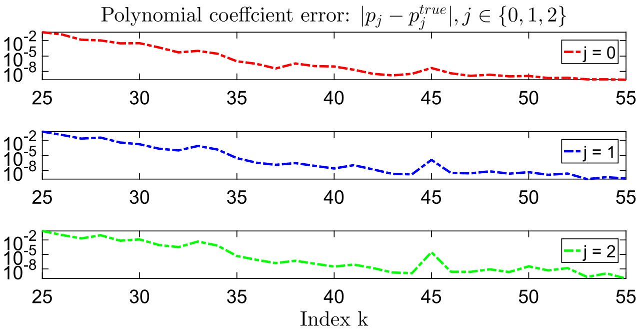

Fig. 8 The error between the true and estimated polynomial coefficients for $k \in [25, 55]$

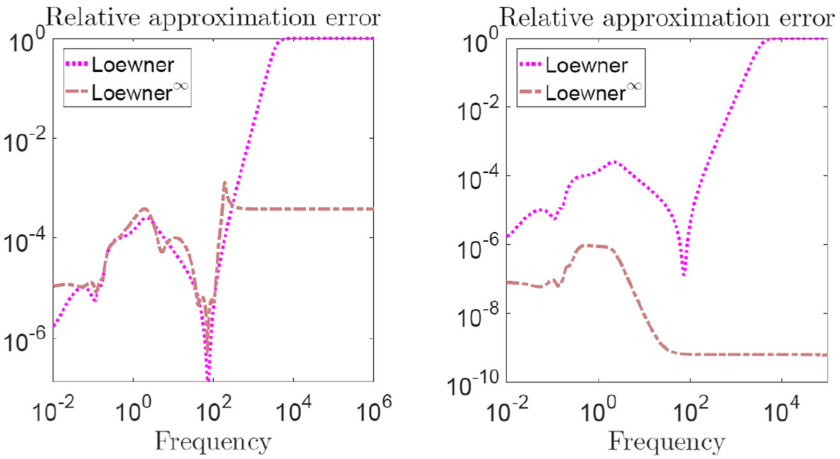

Fig. 9 The relative approximation error of the original frequency response when using the coefficients corresponding to $k_1 = 27$ (left sub-figure) and to $k_2 = 40$ (right sub-figure)

As presented in [26], after performing a spatial discretization, the dynamics is described in generalized state space by the equations

$$\begin{aligned}\mathbf{E}_{11}\frac{d}{dt}\mathbf{v}(t) &= \mathbf{A}_{11}\mathbf{v}(t) + \mathbf{A}_{12}\mathbf{p}(t) + \mathbf{B}_1\mathbf{g}(t),\\ \mathbf{0} &= \mathbf{A}_{12}^T\mathbf{v}(t) + \mathbf{B}_2\mathbf{g}(t),\\ \mathbf{y}(t) &= \mathbf{C}_1\mathbf{v}(t) + \mathbf{C}_2\mathbf{p}(t) + \mathbf{D}\mathbf{g}(t), \end{aligned} \tag{55}$$

where $\mathbf{v} \in \mathbb{R}^{n_v}$, $\mathbf{p} \in \mathbb{R}^{n_p}$ are the states, $\mathbf{g} \in \mathbb{R}^{n_g}$ are the inputs, and $\mathbf{y} \in \mathbb{R}^{n_y}$ are the outputs, and $\mathbf{E}_{11}$ is a symmetric positive definite matrix. The next step is to rewrite the system in (55) as a descriptor system in (1) of dimension $n_v + n_p$, by employing the following notation

$$\mathbf{E} = \begin{pmatrix} \mathbf{E}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^T & \mathbf{0} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} \mathbf{C}_1 & \mathbf{C}_2 \end{pmatrix}, \tag{56}$$

and $\mathbf{x}(t) = \begin{pmatrix} \mathbf{v}(t) \\ \mathbf{p}(t) \end{pmatrix}$. Hence, rewrite (55) as

$$\mathbf{E}\frac{d}{dt}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{g}(t), \quad \mathbf{E}\mathbf{x}(0) = \mathbf{0}, \quad \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{g}(t). \tag{57}$$

In the following, perform a discretization scheme corresponding to the finite element grid in the upper part of Fig. 12 (as in [26]). This corresponds to the following dimension values of the variables in (55): $n_v = 1352$, $n_p = 205$, $n_g = 6$ and $n_y = 2$.

Hence, the original descriptor model constructed as in [26] has dimension $n = 1557$, 6 inputs, 2 outputs, and is of Hessenberg index $\nu = 2$. The coefficients of the

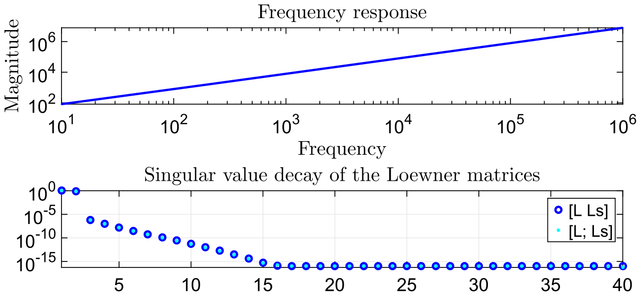

Fig. 10 The frequency response (up) and the decay of singular values (down)

polynomial part of the transfer function can be explicitly computed in terms of the matrices in (56) as

$$\begin{aligned} p_0^{\text{true}} &= \mathbf{D} - \mathbf{C}_1\mathbf{E}_{11}^{-1}\mathbf{A}_{12}(\mathbf{A}_{12}^T\mathbf{E}_{11}^{-1}\mathbf{A}_{12})^{-1}\mathbf{B}_2 - \mathbf{C}_2(\mathbf{A}_{12}^T\mathbf{E}_{11}^{-1}\mathbf{A}_{12})^{-1}\mathbf{A}_{12}^T\mathbf{E}_{11}^{-1}\mathbf{B}, \\ p_1^{\text{true}} &= -\mathbf{C}_2(\mathbf{A}_{12}^T\mathbf{E}_{11}^{-1}\mathbf{A}_{12})^{-1}\mathbf{B}_2, \qquad p_k^{\text{true}} = \mathbf{0},\ k \geqslant 2. \end{aligned} \tag{58}$$

In the numerical simulations, consider only the first input and the second output. Hence, we analyze an index 2 descriptor system with original polynomial coefficients $p_0^{\text{true}} = 16.6241$ and $p_1^{\text{true}} = 7.9211$.

Proceed to choosing data as transfer function measurements corresponding to $N = 40$ logarithmically spaced sampling points in the frequency range $[10^1, 10^6]i$. Afterwards, construct a real-valued 40th order Loewner model. The frequency response (restricted to the above sampling interval) of the system is presented in Fig. 10 together with the singular value decay of the augmented Loewner matrices.

Next, vary the value of the index k from 1 to 40 and compute the values of polynomial coefficients $p_j,\ j \in \{0, 1, 2\}$ for each k. The results are restricted to the trust interval [6, 26] and are depicted in Fig. 11.

Choose the polynomial coefficients corresponding to $k = 16$ (middle of the interval). Then, the deviation between the true coefficients and the estimated ones is computed as $\boldsymbol{\epsilon}^{(1)} = [7.1182 \cdot 10^{-4} \;\; 2.4363 \cdot 10^{-10} \;\; 1.0523 \cdot 10^{-14}]$. We compute

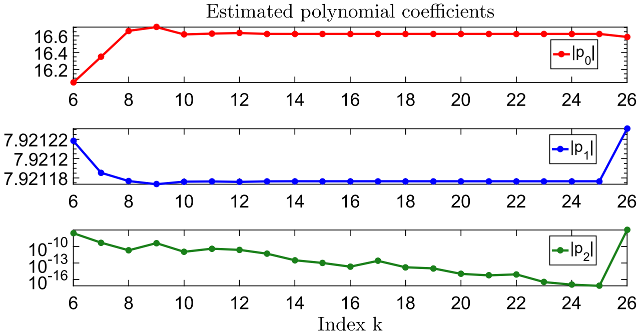

Fig. 11 The estimated polynomial coefficients for $k \in [6, 26]$

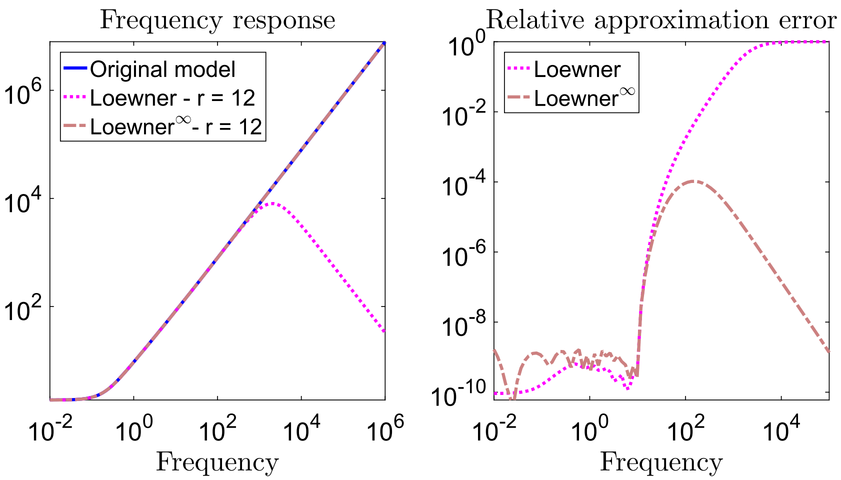

Fig. 12 Frequency response comparison and approximation error

reduced-order models of order $r = 12$ (which corresponds to the singular value $\sigma_{12} \approx 10^{-12}$). We compare the frequency response of the original system with that of the two reduced models of order 12 (computed via the classical Loewner approach and the new modified approach) in Fig. 12. Again, notice that there is a clear mismatch between the original response and the one of the classical Loewner (which is visible starting at around 10^3i). The response computed with the new method follows the original curve. Moreover, in the right side of Fig. 12, we present the relative approximation error between the original and reduced-order models. Clearly, the new method performs better in the high frequency range.

Finally, we perform a time domain simulation and compute the observed output $y(t)$ by means of an explicit forward Euler-type scheme. The control input is given by $u(t) = 0.2\sin(10\pi t) + 0.3\sin(4\pi t)e^{-t}$, while the time horizon is chosen to be $[0, 4]$s for this experiment. We consider zero initial conditions.

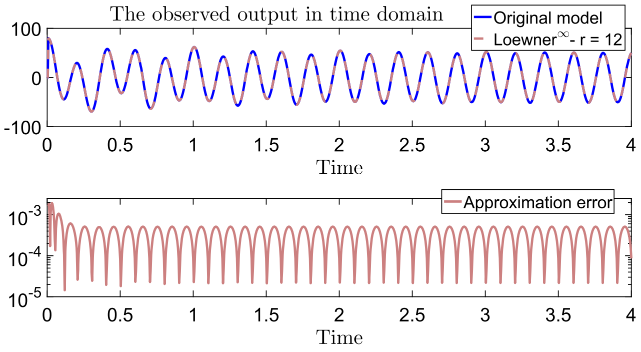

Fig. 13 Simulation in the time domain: the observed output (up) and the approximation error (down)

In the upper part of Fig. 13, we depict the observed output corresponding to the original large-scale system as well the output of the reduced-order system compute via the new proposed method. Notice that the two curves are practically indistinguishable. Additionally, we compute the absolute error between the two observed plots. The result is presented in the lower part of Fig. 13. We observe that the deviation between the two curves is indeed small. The purpose of this experiment was to test the accuracy of the proposed method outside the frequency domain for applications that are primarily considered in time domain (e.g., differential equations such as the Oseen equations).

6 Conclusion

The philosophy behind the classical Loewner framework is that by means of directly compressing the surrogate data model (put together solely from the available measurements), we obtain a reduced-order model that matches the interpolation conditions. In this work, we have addressed an open issue commonly encountered in this framework: the fact that the behavior at infinity of the original system is typically not matched by the reduced system.

We propose an algorithm that is based on existing methods of splitting the raw data Loewner model into two subsystems: one corresponding to the strictly proper part and one corresponding to the polynomial part. The splitting can be done for both regular and singular pairs of Loewner matrices.

Afterwards, we estimated the polynomial coefficients of the transfer function by setting up a trust interval (in which the coefficients have more or less constant values). We have shown that the choice of the coefficients plays a crucial role in the quality of approximation.

Three numerical test cases were chosen amongst well-established benchmark example from the literature. They show the practical applicability of the new proposed method to medium- and large-scale problems.

Finally, there are some open issues and possibly interesting research directions. These include investigating other splitting techniques (for singular pencils) as well as incorporating the case of noisy data in the process (and studying/quantifying the sensitivity to noise of the estimated polynomial coefficients). Finally, the proposed procedure could be extended to DAE systems with nonlinearities (e.g., bilinear control systems) .

Funding Information Open access funding provided by Projekt DEAL.

References

1. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. SIAM, Philadelphia (2005)
2. Antoulas, A.C.: The Loewner framework and transfer functions of singular/rectangular systems. Appl. Math. Lett. **54**, 36–47 (2016)
3. Antoulas, A.C., Beattie, C., Gugercin, S.: Interpolatory Model Reduction of Large-Scale Dynamical Systems, pp. 3–58. Efficient Modeling and Control of Large-Scale Systems. Springer, Boston (2010)
4. Antoulas, A.C., Beattie, C.A., Gugercin, S.: Interpolatory methods for model reduction. Computational Science and Engineering Series. SIAM (2019)
5. Antoulas, A.C., Gosea, I.V., Ionita, A.C.: Model reduction of bilinear systems in the Loewner framework. SIAM J. Sci. Comput. **38**(5), B889–B916 (2016)
6. Antoulas, A.C., Lefteriu, S., Ionita, A.C.: A tutorial introduction to the Loewner framework for model reduction. In: Benner, P., Cohen, A., Ohlberger, M., Willcox, K. (eds.) Model Reduction and Approximation for Complex Systems. Series: Computational Science & Engineering, pp. 335–376. SIAM (2017)
7. Baur, U., Benner, P., Feng, L.: Model order reduction for linear and nonlinear systems: a system-theoretic perspective. Archives of Computational Methods in Engineering **21**, 331–358 (2014)
8. Beelen, T., Van Dooren, P.: An improved algorithm for the computation of Kronecker's canonical form of a singular pencil. Linear Algebra Appl. **105**, 9–65 (1988)
9. Benner, P.: Partial stabilization of descriptor systems using spectral projectors. In: Van Dooren, P., Bhattacharyya, P.S., Chan, H.R., Olshevsky, V., Routray, A. (eds.) Numerical Linear Algebra in Signals, Systems and Control, pp. 55–76. Springer, Netherlands, Dordrecht (2011)
10. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. **57**(4), 483–531 (2015)
11. Benner, P., Losse, P., Mehrmann, V., Voigt, M.: Numerical linear algebra methods for linear differential-algebraic equations. In: Surveys in Differential-Algebraic Equations III, pp. 117–175. Springer (2015)
12. Benner, P., Mehrmann, V., Sorensen, D.C.: Dimension Reduction of Large-Scale Systems. Springer, Berlin (2005)
13. Benner, P., Sokolov, V.I.: Partial realization of descriptor systems. Systems & Control Letters **55**(11), 929–938 (2006)
14. Benner, P., Stykel, T.: Model order reduction of differential-algebraic equations: a survey. In: Ilchmann, A., Reis, T. (eds.) Surveys in Differential-Algebraic Equations IV. Differential-Algebraic Equations Forum, pp. 107–160. Springer International Publishing (2017)
15. Berger, T., Ilchmann, A., Trenn, S.: The quasi-Weierstrass form for regular matrix pencils. Linear Algebra Appl. **436**(10), 4052–4069 (2012)
16. Berger, T., Reis, T.: Controllability of linear differential-algebraic systems - a survey. In: Surveys in Differential-Algebraic Equations I, pp. 1–61. Springer, Berlin (2013)
17. Berger, T., Trenn, S.: The quasi-Kronecker form for matrix pencils. SIAM Journal on Matrix Analysis and Applications **33**(2), 336–368 (2012)
18. Chahlaoui, Y., Van Dooren, P.: A collection of benchmark examples for model reduction of linear time invariant dynamical systems. SLICOT Working Note 2002-2, http://slicot.org/20-site/126-benchmark-examples-for-model-reduction
19. Demmel, J., Kågström, B.: The generalized Schur decomposition of an arbitrary pencil A - zB: robust software with error bounds and applications. part i: theory and algorithms. ACM Trans. Math. Softw. **19**(2), 160–174 (1993)
20. Demmel, J., Kågström, B.: The generalized Schur decomposition of an arbitrary pencil A - zB,: robust software with error bounds and applications. part ii: software and applications. ACM Trans. Math. Softw. **19**(2), 175–201 (1993)
21. Demmel, J., Kågström, B.: Guptri Software for singular pencils. http://www8.cs.umu.se/research/nla/singular_pairs/guptri/ (1993)
22. Duan, G.R.: Analysis and Design of Descriptor Linear Systems. Springer, Berlin (2010)
23. Gosea, I.V., Antoulas, A.C.: Data-driven model order reduction of quadratic-bilinear systems. Special Issue of METT VII, Numercial Linear Algebra with Applications (NLAA), e2200 https://doi.org/10.1002/nla.2200 (2018)
24. Gosea, I.V., Petreczky, M., Antoulas, A.C.: Data-driven model order reduction of linear switched systems in the loewner framework. SIAM J. Sci. Comput. **40**(2), B572–B610 (2018)

25. Gugercin, S., Stykel, T., Wyatt, S.: Model reduction of descriptor systems by interpolatory projection methods. SIAM J. Sci. Comput. **35**(5), B1010–B1033 (2013)
26. Heinkenschloss, M., Sorensen, D.C., Sun, K.: Balanced truncation model reduction for a class of descriptor systems with application to the Oseen equations. SIAM J. Sci. Comput. **30**(2), 1038–1063 (2008)
27. Jonsson, I., Kågström, B.: Recursive blocked algorithms for solving triangular systems – part i: One-sided and coupled Sylvester-type matrix equations. ACM Trans. Math. Software **28**(4), 392–415 (2002)
28. Kågström, B.: RGSVD-An algorithm for computing the Kronecker structure and reducing subspaces of singular A - λ B pencils. SIAM J. Sci. Statist. Comput. **7**, 185–211 (1986)
29. Kågström, B., Poromaa, P.: Lapack-style algorithms and software for solving the generalized sylvester equation and estimating the separation between regular matrix pairs. ACM Trans. Math. Software **22**(1), 78–103 (1996)
30. Kågström, B., Van Dooren, P.: A generalized state-space approach for the additive decomposition of a transfer matrix. J Numer Linear Algebra Appl **1**, 165–181 (1992)
31. Kågström, B., Westin, L.: Generalized schur methods with condition estimators for solving the generalized sylvester equation. IEEE Trans. Automat. Control **34**, 745–751 (1989)
32. Kublanovskaya, V.N.: An approach to solving the spectral problem of $A-\lambda B$, Matrix Pencils, Lecture Notes in Mathematics. In: Kågström, B., Ruhe, A. (eds.), vol. 973, pp. 17–29. Springer (1983)
33. Kunkel, P., Mehrmann, V.: Differential-Algebraic Equations. Analysis and Numerical Solution. EMS Publishing House, Zürich (2006)
34. Mayo, A.J., Antoulas, A.C.: A framework for the generalized realization problem. Linear Algebra and Its Applications **426**, 634–662 (2007)
35. Mehrmann, V., Stykel, T.: Balanced truncation model reduction for large-scale systems in descriptor form. Dimension Reduction of Large-Scale Systems **45**, 83–115 (2005)
36. Mehrmann, V., Stykel, T.: Descriptor systems: a general mathematical framework for modelling, simulation and control. Automatisierungstechnik **54**(8), 405–415 (2006)
37. Seiwald, P., Castagnotto, A., Stykel, T., Lohmann, B.: $\mathcal{H}_2$ pseudo-optimal reduction of structured DAEs by rational interpolation. Tech. rep. https://arxiv.org/pdf/1804.08755.pdf (2018)
38. Stykel, T.: Gramian based model reduction for descriptor systems. Mathematics of Control, Signals, and Systems **16**, 297–319 (2004)
39. Trenn, S.: Solution concepts for linear DAEs: a survey. In: Surveys in Differential-Algebraic Equations I, pp. 137–172. Springer, Berlin (2013)
40. Van Dooren, P.: The computation of Kronecker's canonical form of a singular pencil. Linear Algebra Appl. **27**, 103–140 (1979)
41. Werner, S.: Hankel-norm approximation of descriptor systems. Ph.D. thesis, Department of Mathematics, Otto-von-Guerricke-University Magdeburg (2016)

Affiliations

Ion Victor Gosea[1] · Qiang Zhang[2] · Athanasios C. Antoulas[1,2,3]

Qiang Zhang
qz18@rice.edu

Athanasios C. Antoulas
aca@rice.edu

[1] Data-Driven System Reduction and Identification (DRI) Group, Max Planck Institute
for Dynamics of Complex Technical Systems, Sandtorstrasse 1, 39106, Magdeburg, Germany

[2] Department of Electrical and Computer Engineering, Rice University, 6100 Main St, MS-366, Houston, TX, 77005, USA

[3] Baylor College of Medicine, 1 Baylor Plaza, Houston, TX, 77030, USA