Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 55-30 (2022) 377–382

# The one-sided Loewner framework and connections to other model reduction methods based on interpolation

**Ion Victor Gosea** * **Athanasios C. Antoulas** **

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

** *Department of Electrical and Computer Engineering, Rice University, Houston, Baylor College of Medicine, Houston, and Max Planck Institute, Magdeburg, Germany, (e-mail: aca@rice.edu)*

**Abstract:** We present an extension of the Loewner framework, an established data-driven reduction, and identification method. This will be referred to as the one-sided Loewner framework since only one set of interpolation conditions are explicitly and exactly matched. For the other set of conditions, approximated interpolation is imposed. We describe how to explicitly characterize new interpolation conditions, derived from the latter set. We also show connections to the iterative AAA algorithm. Typical applications include constructing reduced models from frequency response data measured from systems in electronics or mechanical engineering. We illustrate the application of the main method on a large-scale benchmark example.



*Keywords:* Model reduction, data-driven methods, transfer function, interpolation methods.

## 1. INTRODUCTION

Complex physical processes used in many real-world applications typically require as models dynamical systems that are simple/small enough to be simulated or to be used for control design purposes. Model order reduction (MOR) is a well-established technique used for reducing the computational complexity of mathematical models in numerical simulations. Many modern mathematical models of real-life processes pose challenges when used in numerical simulations, due to the large size, or its complexity. By means of MOR tools, the computational complexity of such problems is lowered, such as for simulations of large-scale dynamical control systems. Typically, most MOR methods aim at conserving some inherent properties and characteristics of the full order model. For example, these include stability or passivity in electronics engineering. Additionally, the surrogate models thus obtained typically inherit the structure of the original system, together with some relevant mathematical features, e.g., dominant poles in common, and similarity of the transfer functions.

Model order reduction (MOR) techniques Antoulas [2005], Benner et al. [2017], Antoulas et al. [2020] play a crucial role in obtaining such surrogates. MOR is usually based on the full knowledge of the system's structure, which can be derived from physics laws (so-called intrusive techniques). However, the availability of measured data and the rise of data-driven applications together with machine learning techniques require incorporating measurements when modeling or controlling a system. As such, we will concentrate in this work on data-driven methods. The Loewner framework (LF) Mayo and Antoulas [2007] is a non-intrusive data-driven reduction technique in the frequency-domain based on interpolation and is of particular relevance in this context. For recent tutorial papers on LF for linear systems, we refer the reader to Antoulas et al. [2017] and to Karachalios et al. [2021], and for an extension that uses time-domain data, we refer the reader to Peherstorfer and Willcox [2016]. In recent years, an iterative and adaptive extension of the LF was proposed in Nakatsukasa et al. [2018]. Additionally, the Loewner framework has been recently extended to certain classes of nonlinear systems, such as bilinear systems in Antoulas et al. [2016]. It is to be mentioned that another one-sided interpolation framework was proposed in Astolfi [2010].

In what follows, we analyze linear systems characterized in state-space realization by the following equations:

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

where $\mathbf{E}, \mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{B}, \mathbf{C}^T \in \mathbb{C}^n$ and $\mathbf{D} \in \mathbb{C}$ (we analyze here the SISO case, with $m = 1$ input and $p = 1$ output). From measured data, i.e., samples of the transfer function $\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$ evaluated at particular interpolation points, the goal is to compute reduced-order models (ROMs) as in (1), without having access to the original realization $(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$.

The goal of this paper is to present one-sided extensions of the LF, and to link these with the newly-proposed AAA algorithm in Nakatsukasa et al. [2018]. The latter is a very robust and effective method, that is based on applying LF iteratively, until a reasonable approximation on the data has been achieved. AAA explicitly enforces interpolation on the set of support points of the barycentric form (see Berrut and Trefethen [2004] for details) of the rational approximant, which are the so-called right interpolation points in the LF, see Antoulas et al. [2017]. We propose


10.1016/j.ifacol.2022.11.082

an one-sided Loewner framework (OSLF), that is based on a similar philosophy as AAA. However, the method proposed here is not iterative, and a novel result is the explicit characterization of "compressed" left interpolation points, that bridges the LF and AAA methodologies. After the introduction is established, Section 2 briefly reviews the LF and AAA methods. The theoretical discussion in Section 3 (which presents the main method and contribution) is illustrated by numerical examples in Section 4.

## 2. THE METHODS UNDER CONSIDERATION

### *2.1 The Loewner framework in Mayo and Antoulas [2007]*

In this section we present a short summary of the Loewner framework (LF), as introduced in Mayo and Antoulas [2007]. It is to mentioned that LF has its roots in the earlier work of Antoulas and Anderson [1986].

The LF is based on frequency-domain measurements $\mathbf{H}(\imath\omega_i)$ corresponding to the transfer function of the underlying (potentially unknown) system. Typically, frequency response data are used which can be inferred in practice from various experiments. The interpolation problem under hand is formulated as follows; we are given data values and data points, partitioned into two disjoint sets

$$\begin{aligned} &\underline{\text{right data}} : \{(\lambda_j, \mathbf{H}(\lambda_j))\,,\ j = 1, \ldots, k\},\ \text{and,} \\ &\underline{\text{left data}} : \{(\mu_i, \mathbf{H}(\mu_i))\,,\ i = 1, \ldots, q\}. \end{aligned} \tag{2}$$

For simplicity, all points are assumed distinct, and it considered that the left and right sets have the same dimension, i.e., $q = k$.

We seek to find a rational function $\hat{\mathbf{H}}(s)$, such that the interpolation conditions hold for all $i, j = 1, \ldots, k$:

$$\hat{\mathbf{H}}(\mu_i) = \mathbf{H}(\mu_i) := v_i, \quad \hat{\mathbf{H}}(\lambda_j) = \mathbf{H}(\lambda_j) := w_j. \tag{3}$$

The case of data subsets with different dimensions, i.e., in which $k \neq q$ is also compatible in the Loewner framework; see Antoulas et al. [2017], and Antoulas [2016].

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

$$\mathbb{L}_{(i,j)} = \frac{v_i - w_j}{\mu_i - \lambda_j}, \quad \mathbb{L}_{s(i,j)} = \frac{\mu_i v_i - \lambda_j w_j}{\mu_i - \lambda_j}, \tag{4}$$

while the data vectors $\mathbb{V} \in \mathbb{C}^k, \mathbb{W}^T \in \mathbb{C}^k$ are as:

$$\mathbb{V}_{(i)} = v_i, \quad \mathbb{W}_{(j)} = w_j, \ \text{for } i = 1, \ldots, k,\ j = 1, \ldots, k. \tag{5}$$

The (double-sided) Loewner quadruple is composed of:

$$\hat{\mathbf{E}} = -\mathbb{L}, \quad \hat{\mathbf{A}} = -\mathbb{L}_s, \quad \hat{\mathbf{B}} = \mathbb{V}, \quad \hat{\mathbf{C}} = \mathbb{W}. \tag{6}$$

*Theorem 1.* Here, we assume that the Loewner pencil $(\mathbb{L}_s, \mathbb{L})$ is regular. The transfer function of the (double-sided) Loewner model in (6), given by: $\hat{\Sigma}_L : (\hat{\mathbf{E}}, \hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}})$ constructed above, matches both the left & right data, i.e.:

$$\hat{\mathbf{H}}(\mu_i) = v_i, \quad \hat{\mathbf{H}}(\lambda_j) = w_j, \quad \forall\ 1 \leq i, j \leq k, \tag{7}$$

where the transfer function is computed as follows:

$$\hat{\mathbf{H}}(s) = \hat{\mathbf{C}}\left(s\hat{\mathbf{E}} - \hat{\mathbf{A}}\right)^{-1}\hat{\mathbf{B}} = \mathbb{W}\left(-s\mathbb{L} + \mathbb{L}_s\right)^{-1}\mathbb{V}. \tag{8}$$

*Proof 1.* We refer the reader to the original contribution Mayo and Antoulas [2007] and to Antoulas et al. [2017].

Additionally, the next Sylvester equations in [Antoulas 2005, chap. 6] are satisfied by the Loewner and shifted Loewner matrices ($\mathbf{L} = \mathbf{R}^T = \mathbf{1}_k = [1\ \cdots\ 1]^T \in \mathbb{C}^k$):

$$\begin{cases} \mathbf{M}\mathbb{L} - \mathbb{L}\mathbf{\Lambda} = \mathbb{V}\mathbf{R} - \mathbf{L}\mathbb{W}, \\ \mathbf{M}\mathbb{L}_s - \mathbb{L}_s\mathbf{\Lambda} = \mathbf{M}\mathbb{V}\mathbf{R} - \mathbf{L}\mathbb{W}\mathbf{\Lambda}, \end{cases} \tag{9}$$

where $\mathbf{M} = \text{diag}(\mu_1, \cdots, \mu_k)$ and $\mathbf{\Lambda} = \text{diag}(\lambda_1, \cdots, \lambda_k)$. It also follows that:

$$\mathbb{L}_s = \mathbb{L}\mathbf{\Lambda} + \mathbb{V}\mathbf{R} = \mathbf{M}\mathbb{L} + \mathbf{L}\mathbb{W}. \tag{10}$$

In practical applications, the pencil $(\mathbb{L}_s, \mathbb{L})$ is often singular. In these cases, perform a rank revealing singular value decomposition (SVD) of the Loewner matrices. Then, compute projection matrices $\mathbf{X}_r, \mathbf{Y}_r \in \mathbb{C}^{k\times r}$ as the left, and respectively, the right truncated singular vector matrices. More details can be found in Antoulas et al. [2017]. Here, $r < n$ represents the truncation index. The system matrices corresponding to a projected Loewner model of dimension $r$ can be computed

$$\tilde{\mathbf{E}} = -\mathbf{X}_r^*\mathbb{L}\mathbf{Y}_r, \quad \tilde{\mathbf{A}} = -\mathbf{X}_r^*\mathbb{L}_s\mathbf{Y}_r, \quad \tilde{\mathbf{B}} = \mathbf{X}_r^*\mathbb{V}, \quad \tilde{\mathbf{C}} = \mathbb{W}\mathbf{Y}_r.$$

### *2.2 The AAA algorithm*

The AAA algorithm Nakatsukasa et al. [2018] represents an adaptive extension of the interpolation-based method in Antoulas and Anderson [1986]. It is a robust and fast method used for rational interpolation applications. AAA is a multi-step algorithm, that computes a rational approximant of order $(k-1, k-1)$ in barycentric representation at step $k > 1$ . Additionally, as for the Loewner method, we restrict the presentation to the SISO case and point to the MIMO extension in Gosea and Güttel [2021].

The algorithm enforces a data splitting at step $\ell$:

$$\begin{aligned} &\text{data points}: \ \{z_h\}_{h=1}^{2M} = \{\nu_j\}_{j=1}^{\ell} \cup \{\eta_i\}_{i=1}^{2M-\ell}, \\ &\text{data values}: \ \{f_h\}_{h=1}^{2M} = \{h_j\}_{j=1}^{\ell} \cup \{g_i\}_{i=1}^{2M-\ell}. \end{aligned} \tag{11}$$

Note that the values $f_h$ in (11) represent the measurements evaluated at the points $z_h$, while $h_j$ and $g_i$ are the ones evaluated at $\nu_j$, and respectively at $\eta_i$.

The rational interpolant $\mathbf{H}_k$, obtained after $k$ iterations of the AAA algorithm, is written in the barycentric form as

$$\mathbf{H}_k(s) = \sum_{j=1}^{k} \frac{\alpha_j^{(k)} h_j}{s - \nu_j} \Big/ \sum_{j=1}^{k} \frac{\alpha_j^{(k)}}{s - \nu_j}, \tag{12}$$

with nonzero barycentric weights $\alpha_j^{(k)} \in \mathbb{C}$, pairwise distinct support points $\nu_j \in \mathbb{C}$, and function values $h_j$. Based on (12), interpolation is enforced at the points $\{\nu_j\}_{j=1}^{k}$, i.e., $\mathbf{H}_k(\nu_j) = h_j$ for $1 \leq j \leq k$. In order to completely determine the rational approximant $\mathbf{H}_k$, one needs to also find the barycentric weights. This is done by solving a least squares problem; the next support point is chosen by means of a greedy selection.

Based on the representation of the transfer function at step $\ell$ in (12), one can formulate the minimization problem as

$$\begin{aligned} &\min_{\alpha_1^{(\ell)}, \ldots, \alpha_\ell^{(\ell)}} \sum_{k=1}^{2M} (\mathbf{H}_\ell(z_k) - f_k)^2 \\ &\Leftrightarrow \min \sum_{k=1}^{2M} \left( \frac{\sum_{j=1}^{\ell} \frac{\alpha_j^{(\ell)}(f_k - h_j)}{z_k - \nu_j}}{\sum_{j=1}^{\ell} \frac{\alpha_j^{(\ell)}}{z_k - \nu_j}} \right)^2. \end{aligned} \tag{13}$$

Instead of solving the nonlinear problem in (13), one solves a linearized problem derived from it. For more details and exact implementation, see the Nakatsukasa et al. [2018].

## 3. THE ONE-SIDED LOEWNER FRAMEWORK

In this section we will outline the main procedure proposed in this work. We will start by introducing the quantities of interest, that will be used in computing rational approximants for the given data. The first scenario treated here (in Section 3.1) is the one for which the right data has one entry more than the left data. We show that interpolation is enforced on all the sampling points. The more relevant case for this contribution will be treated in the second part of this section, i..e, in Section 3.2. There, the left data set is (much) larger than the right set, and we will show how to impose explicit interpolation on a smaller (compressed) left data set. This scenario is of particular relevance since it appears in the AAA algorithm.

As in classical formulation of LF in Section 2.1, the Loewner matrix $\mathbb{L} \in \mathbb{C}^{q\times k}$ stays unchanged, i.e., $\mathbb{L}_{(i,j)} = \frac{v_i - w_j}{\mu_i - \lambda_j}$ while the one-sided shifted Loewner matrices ${\mathbb{L}_s}^{\mu} \in \mathbb{C}^{q\times k}$ (for the left side) and ${\mathbb{L}_s}^{\lambda} \in \mathbb{C}^{q\times k}$ (for the right side), are defined as follows Antoulas et al. [2017]:

$$ {\mathbb{L}_s}^{\mu}_{(i,j)} = \mu_i \frac{v_i - w_j}{\mu_i - \lambda_j},\ {\mathbb{L}_s}^{\lambda}_{(i,j)} = \frac{v_i - w_j}{\mu_i - \lambda_j}\lambda_j, \tag{14} $$

for all $i = 1, \ldots, q,\ j = 1, \ldots, k$. Note that the newly-introduced matrices in (14) can be factored in terms of the Loewner matrix as:

$$ {\mathbb{L}_s}^{\mu} = \mathbf{M}\mathbb{L},\ \ {\mathbb{L}_s}^{\lambda} = \mathbb{L}\mathbf{\Lambda}. \tag{15} $$

Additionally, note that the data vectors $\mathbb{V} \in \mathbb{R}^q, \mathbb{W}^T \in \mathbb{R}^k$ are defined as in (5). Here, we concentrate mostly on the case of one-sided right interpolant, hence we are using ${\mathbb{L}_s}^{\lambda}$ (the converse results also apply to ${\mathbb{L}_s}^{\mu}$).

Let $\mathbf{J}_{k-1} \in \mathbb{C}^{k\times(k-1)}$ so that $\mathbf{J}_{k-1} = \begin{bmatrix} \mathbf{I}_{k-1} \\ -\mathbf{1}_{k-1}^T \end{bmatrix}$, where $\mathbf{1}_{k-1} = [1\ \cdots\ 1]^T \in \mathbb{C}^{k-1}$.

Intuitively, by multiplying a matrix $\mathbf{X} \in \mathbb{C}^{k\times k}$ with matrix $\mathbf{J}_{k-1}$ to the right has the effect of subtracting the $k$th column of $\mathbf{X}$ from the previous $k-1$ columns, i.e. for $\mathbf{X} = [\mathbf{X}_1\ \mathbf{X}_2\ \cdots\ \mathbf{X}_k]$, we have that:

$$ \mathbf{X}\mathbf{J}_{k-1} = [\mathbf{X}_1 - \mathbf{X}_k\ \ \mathbf{X}_2 - \mathbf{X}_k\ \cdots\ \mathbf{X}_{k-1} - \mathbf{X}_k] \in \mathbb{C}^{k\times(k-1)}. $$

Since $\mathbf{R} = \mathbf{1}_k^T$, we get that $\mathbf{R}\mathbf{J}_{k-1} = \mathbf{0}_{k-1}$. From (9) and (15), it directly follows that the Loewner matrix $\mathbb{L}$ satisfies the following identity:

$$ \mathbf{M}\mathbb{L} - {\mathbb{L}_s}^{\lambda} = \mathbb{V}\mathbf{R} - \mathbf{L}\mathbb{W}. \tag{16} $$

*Definition 1.* We introduce new matrices, computed from the original ones defined earlier, by multiplying them with the matrix $\mathbf{J}_{k-1}$, to the right, as:

$$ \hat{\mathbb{L}} = \mathbb{L}\mathbf{J}_{k-1},\ \ \hat{\mathbb{L}}_s{}^{\lambda} = {\mathbb{L}_s}^{\lambda}\mathbf{J}_{k-1},\ \text{and}\ \hat{\mathbb{W}} = \mathbb{W}\mathbf{J}_{k-1}. \tag{17} $$

By multiplying (16) with $\mathbf{J}_{k-1}$ to the right, we get that:

$$ \mathbf{M}\mathbb{L}\mathbf{J}_{k-1} - {\mathbb{L}_s}^{\lambda}\mathbf{J}_{k-1} = \mathbb{V}\mathbf{R}\mathbf{J}_{k-1} - \mathbf{L}\mathbb{W}\mathbf{J}_{k-1}, $$

$$ \Rightarrow \boxed{\hat{\mathbb{L}}_s{}^{\lambda} = \mathbf{M}\,\hat{\mathbb{L}} + \mathbf{L}\hat{\mathbb{W}}} \tag{18} $$

We will show that such a relation, i.e., as in (18), holds also for the case in which the left interpolation points are compressed (by means of projecting).

### 3.1 The case with $q = k - 1$

*Theorem 2.* Let $q = k-1$ and define the following rational function in terms of the quantities from before:

$$ \hat{\mathbf{H}}^{\lambda}(s) = w_k - (s - \lambda_k)\hat{\mathbb{W}}\left(s\,\hat{\mathbb{L}} - \ \hat{\mathbb{L}}_s{}^{\lambda}\right)^{-1}\mathbb{L}\mathbf{e}_k, \tag{19} $$

where $\mathbf{e}_k = [0\ 0\ \cdots\ 1]^T \in \mathbb{C}^k$. It follows that $\hat{\mathbf{H}}^{\lambda}(s)$ in (19) satisfies the $2k-1$ interpolation conditions in (2), i.e.:

$$ \hat{\mathbf{H}}^{\lambda}(\mu_i) = \mathbf{H}(\mu_i),\ \ i = 1, \ldots, k-1, \tag{20} $$

$$ \hat{\mathbf{H}}^{\lambda}(\lambda_j) = \mathbf{H}(\lambda_j),\ \ j = 1, \ldots, k. \tag{21} $$

*Proof 2.* We start with proving the first set of interpolation conditions in (20). For $1 \le i \le k-1$ we can write:

$$ \begin{aligned} \hat{\mathbf{H}}^{\lambda}(\mu_i) &= w_k - (\mu_i - \lambda_k)\hat{\mathbb{W}}\left(\mu_i\,\hat{\mathbb{L}} - \ \hat{\mathbb{L}}_s{}^{\lambda}\right)^{-1}\mathbb{L}\mathbf{e}_k \\ &= w_k - (\mu_i - \lambda_k)\hat{\mathbb{W}}\left(\mu_i\,\hat{\mathbb{L}} - \mathbf{M}\,\hat{\mathbb{L}} - \mathbf{L}\hat{\mathbb{W}}\right)^{-1}\mathbb{L}\mathbf{e}_k. \end{aligned} $$

Now, since $\mathbf{M} = \mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_{k-1})$, it follows that $\mathbf{e}_i^T\left(\mu_i\,\hat{\mathbb{L}} - \mathbf{M}\,\hat{\mathbb{L}} - \mathbf{L}\hat{\mathbb{W}}\right) = -\hat{\mathbb{W}}$, and we can hence write:

$$ \hat{\mathbb{W}}\left(\mu_i\,\hat{\mathbb{L}} - \mathbf{M}\,\hat{\mathbb{L}} - \mathbf{L}\hat{\mathbb{W}}\right)^{-1} = -\mathbf{e}_i^T. \tag{22} $$

Finally, since $\mathbb{L}_{(i,k)} = \frac{v_i - w_k}{\mu_i - \lambda_k}$ for $1 \le i \le k-1$, it follows:

$$ \hat{\mathbf{H}}^{\lambda}(\mu_i) = w_k + (\mu_i - \lambda_k)\mathbf{e}_i^T\mathbb{L}\mathbf{e}_k = w_k + (v_i - w_k) = v_i. $$

Now, we continue with the second set of interpolation conditions in (21). Obviously, for $s = \lambda_k$, it follows that $\hat{\mathbf{H}}^{\lambda}(\lambda_k) = w_k$. Choose now $1 \le j \le k-1$ and proceed as:

$$ \begin{aligned} \hat{\mathbf{H}}^{\lambda}(\lambda_j) &= w_k - (\lambda_j - \lambda_k)\hat{\mathbb{W}}\left(\lambda_j\,\hat{\mathbb{L}} - \ \hat{\mathbb{L}}_s{}^{\lambda}\right)^{-1}\mathbb{L}\mathbf{e}_k \\ &= w_k - (\lambda_j - \lambda_k)\hat{\mathbb{W}}\left(\lambda_j\mathbb{L}\mathbf{J}_{k-1} - {\mathbb{L}_s}^{\lambda}\mathbf{J}_{k-1}\right)^{-1}\mathbb{L}\mathbf{e}_k. \end{aligned} $$

By using that:

$$ \lambda_j\mathbb{L}\mathbf{J}_{k-1} - {\mathbb{L}_s}^{\lambda}\mathbf{J}_{k-1} = \mathbb{L}\left(\lambda_j\mathbf{I} - \mathbf{\Lambda}\right)\mathbf{J}_{k-1}, $$

it follows that we can write:

$$ \left(\lambda_j\mathbb{L}\mathbf{J}_{k-1} - {\mathbb{L}_s}^{\lambda}\mathbf{J}_{k-1}\right)^{-1}\mathbb{L}\mathbf{e}_k = \frac{1}{\lambda_j - \lambda_k}\mathbf{e}_j. $$

Then, by using the previous identity, we get that:

$$ \begin{aligned} \hat{\mathbf{H}}^{\lambda}(\lambda_j) &= w_k - (\lambda_k - \lambda_j)\hat{\mathbb{W}}\frac{1}{\lambda_j - \lambda_k}\mathbf{e}_j \\ &= w_k + \hat{\mathbb{W}}\mathbf{e}_j = w_k + (w_j - w_k) = w_j. \end{aligned} $$

*Connections to barycentric forms* In what follows, we connect the barycentric representation of the rational interpolant as in Antoulas and Anderson [1986], to the previously-defined representation in (19).

We compute the SVD of the Loewner matrix $\mathbb{L} \in \mathbb{C}^{(k-1)\times k}$

$$ \mathbb{L} = \mathbf{X}\mathbf{S}\mathbf{Y}^*. \tag{23} $$

Now, since the matrix $\mathbb{L}$ has one more columns than rows, it admits (at least) one vector in the right null-space (given by the last column of $\mathbf{Y}$), i.e., $\mathbb{L}\mathbf{Y}\mathbf{e}_k = \mathbf{0}_{k-1}$.

*Lemma 3.* The following identity holds true for all $s \in \mathbb{C}$, i.e., the rational interpolant in (19) has the following barycentric representation:

$$ w_k - (s - \lambda_k)\hat{\mathbb{W}}\left(s\,\hat{\mathbb{L}} - \ \hat{\mathbb{L}}_s{}^{\lambda}\right)^{-1}\mathbb{L}\mathbf{e}_k = \frac{\sum_{j=1}^{k}\frac{y_j w_j}{s-\lambda_j}}{\sum_{j=1}^{k}\frac{y_j}{s-\lambda_j}}, $$

where the "weights" values $y_i$'s are entries of the last right singular vector of the Loewner matrix $\mathbb{L}$, i.e., the last column of matrix $\mathbf{Y}$: $[y_1 \ y_2 \ \cdots \ y_k]^T = \mathbf{Y}\mathbf{e}_k$.

*3.2 The case with $q \geq k$ (more interpolation conditions in the left data subset)*

Assume now that $q \geq k$, i.e., the left interpolation conditions are more numerous or the same number as the right ones (as it is typical the case for AAA). The first step is to compute a compact SVD of the matrix $\hat{\mathbb{L}} = \mathbb{L}\mathbf{J}_{k-1} \in \mathbb{C}^{q\times(k-1)}$ as

$$\hat{\mathbb{L}} = \hat{\mathbf{X}}\hat{\mathbf{S}}\hat{\mathbf{Y}}^*, \tag{24}$$

with unitary matrices $\hat{\mathbf{X}} \in \mathbb{C}^{q\times(k-1)}$, and $\hat{\mathbf{Y}} \in \mathbb{C}^{(k-1)\times(k-1)}$, i.e., satisfying $\hat{\mathbf{X}}^T\hat{\mathbf{X}} = \mathbf{I}_{k-1}$ and $\hat{\mathbf{Y}}^*\hat{\mathbf{Y}} = \hat{\mathbf{Y}}\hat{\mathbf{Y}}^* = \mathbf{I}_{k-1}$, and with the diagonal matrix $\hat{\mathbf{S}} \in \mathbb{C}^{(k-1)\times(k-1)}$.

*Definition 2.* We define the left-compressed Loewner quantities, as:

$$\tilde{\mathbb{L}} = \hat{\mathbf{X}}^* \hat{\mathbb{L}}, \quad \tilde{\mathbb{L}}_s{}^{\lambda} = \hat{\mathbf{X}}^* \hat{\mathbb{L}}_s{}^{\lambda}, \quad \tilde{\mathbf{L}} = \hat{\mathbf{X}}^*\mathbf{L}, \tag{25}$$

Now, by multiplying the identity in (18) with $\hat{\mathbf{X}}^*$ ot the left, and by using the notations in (24) and (25), it follows:

$$\begin{aligned} \hat{\mathbf{X}}^* \ \hat{\mathbb{L}}_s{}^{\lambda} &= \hat{\mathbf{X}}^*\mathbf{M} \underbrace{\hat{\mathbb{L}}}_{\hat{\mathbf{X}}\hat{\mathbf{S}}\hat{\mathbf{Y}}^*} + \hat{\mathbf{X}}^*\mathbf{L}\hat{\mathbb{W}} \\ \Rightarrow \ \tilde{\mathbb{L}}_s{}^{\lambda} &= \left(\hat{\mathbf{X}}^*\mathbf{M}\hat{\mathbf{X}}\right)\hat{\mathbf{S}}\hat{\mathbf{Y}}^* + \tilde{\mathbf{L}}\hat{\mathbb{W}}. \end{aligned} \tag{26}$$

*Definition 3.* Let $\tilde{\mathbf{M}} = \hat{\mathbf{X}}^*\mathbf{M}\hat{\mathbf{X}} \in \mathbb{C}^{(k-1)\times(k-1)}$ be the matrix obtained by compressing the original diagonal matrix $\mathbf{M} \in \mathbb{C}^{q\times q}$ (that contains the left interpolation points on its main diagonal).

Now, since $\hat{\mathbf{X}}^*\hat{\mathbf{X}} = \mathbf{I}_{k-1}$, it follows that $\tilde{\mathbb{L}} = \hat{\mathbf{X}}^* \hat{\mathbb{L}} = \hat{\mathbf{X}}^* \left(\hat{\mathbf{X}}\hat{\mathbf{S}}\hat{\mathbf{Y}}^*\right) = \hat{\mathbf{S}}\hat{\mathbf{Y}}^*$. By combining this result with that in (26), it holds true that:

$$\boxed{\tilde{\mathbb{L}}_s{}^{\lambda} = \tilde{\mathbf{M}} \ \tilde{\mathbb{L}} + \ \tilde{\mathbf{L}}\hat{\mathbb{W}}} \tag{27}$$

*Definition 4.* The left compressed interpolation points are identified as the eigenvalues of the compressed matrix $\tilde{\mathbf{M}} = \hat{\mathbf{X}}^*\mathbf{M}\hat{\mathbf{X}} \in \mathbb{C}^{(k-1)\times(k-1)}$.

Finally, consider the EVD decomposition of matrix $\tilde{\mathbf{M}}$ as:

$$\tilde{\mathbf{M}} = \tilde{\mathbf{U}}\breve{\mathbf{M}}\tilde{\mathbf{U}}^{-1}, \tag{28}$$

with $\tilde{\mathbf{U}}, \breve{\mathbf{M}} \in \mathbb{C}^{(k-1)\times(k-1)}$ and $\breve{\mathbf{M}} = \mathrm{diag}(\breve{\mu}_1, \cdots, \breve{\mu}_{k-1})$. The left compressed interpolation points are hence denoted with $\breve{\mu}_1, \breve{\mu}_2, \ldots, \breve{\mu}_{k-1}$.

*Definition 5.* We define left-compressed Loewner quantities by applying a change of coordinates from (25), i.e., by multiplying those matrices with $\tilde{\mathbf{U}}^{-1}$ to the left, as:

$$\breve{\mathbb{L}} = \tilde{\mathbf{U}}^{-1} \ \tilde{\mathbb{L}}, \quad \breve{\mathbb{L}}_s{}^{\lambda} = \tilde{\mathbf{U}}^{-1} \ \tilde{\mathbb{L}}_s{}^{\lambda}, \quad \breve{\mathbf{L}} = \tilde{\mathbf{U}}^{-1}\tilde{\mathbf{L}}, \tag{29}$$

and also $\breve{\mathbf{M}} = \tilde{\mathbf{U}}^{-1}\tilde{\mathbf{M}}\tilde{\mathbf{U}}$.

By multiplying equality (27) with $\tilde{\mathbf{U}}^{-1}$ to the left, and substituting the notations in (29), it follows that the following equality holds true:

$$\boxed{\breve{\mathbb{L}}_s{}^{\lambda} = \breve{\mathbf{M}} \ \breve{\mathbb{L}} + \ \breve{\mathbf{L}}\hat{\mathbb{W}}}. \tag{30}$$

Additionally, we have that:

$$\breve{\mathbb{L}} = \tilde{\mathbf{U}}^{-1} \ \tilde{\mathbb{L}} = \tilde{\mathbf{U}}^{-1}\hat{\mathbf{X}}^* \ \hat{\mathbb{L}} = \tilde{\mathbf{U}}^{-1}\hat{\mathbf{X}}^*\mathbb{L}\mathbf{J}_{k-1}, \tag{31}$$

and let $\bar{\mathbb{L}} \in \mathbb{C}^{(k-1)\times k}$ be a matrix so that

$$\bar{\mathbb{L}} = \tilde{\mathbf{U}}^{-1}\hat{\mathbf{X}}^*\mathbb{L}. \tag{32}$$

Note that, from (31) and (32), it follows that $\breve{\mathbb{L}} = \bar{\mathbb{L}}\mathbf{J}_{k-1}$. The matrix defined in (32) will explicitly appear in the rational interpolant constructed in what follows. We are now ready to state the main result.

*Theorem 4.* For $q > k-1$ we introduce the following rational function in terms of the Loewner matrices and other matrices previously defined, as:

$$\breve{\mathbf{H}}^{\lambda}(s) = w_k - (s-\lambda_k)\hat{\mathbb{W}}\left(s \ \breve{\mathbb{L}} - \ \breve{\mathbb{L}}_s{}^{\lambda}\right)^{-1} \ \bar{\mathbb{L}}\mathbf{e}_k. \tag{33}$$

It follows that $\breve{\mathbf{H}}^{\lambda}(s)$ defined above satisfies the original $k$ right interpolation conditions:

$$\breve{\mathbf{H}}^{\lambda}(\lambda_j) = \mathbf{H}(\lambda_j), \quad j = 1, \ldots, k. \tag{34}$$

and also new $k-1$ left interpolation conditions:

$$\breve{\mathbf{H}}^{\lambda}(\breve{\mu}_i) = \breve{v}_i, \quad i = 1, \ldots, k-1. \tag{35}$$

*Proof 3.* The first set of interpolation conditions in (34) can be proven similarly as in the proof of Theorem 1. Hence, we proceed by explicitly proving the second set of conditions in (35). For $1 \leq i \leq k-1$ we can write:

$$\begin{aligned} \breve{\mathbf{H}}^{\lambda}(\breve{\mu}_i) &= w_k - (\breve{\mu}_i - \lambda_k)\hat{\mathbb{W}}\left(\breve{\mu}_i \ \breve{\mathbb{L}} - \ \breve{\mathbb{L}}_s{}^{\lambda}\right)^{-1} \ \bar{\mathbb{L}}\mathbf{e}_k \\ &= w_k - (\breve{\mu}_i - \lambda_k)\hat{\mathbb{W}}\left(\breve{\mu}_i \ \breve{\mathbb{L}} - \breve{\mathbf{M}} \ \breve{\mathbb{L}} - \ \breve{\mathbf{L}}\hat{\mathbb{W}}\right)^{-1} \ \bar{\mathbb{L}}\mathbf{e}_k. \end{aligned}$$

Now, since

$$\breve{\mathbf{M}} = \mathrm{diag}(\breve{\mu}_1, \breve{\mu}_2, \ldots, \breve{\mu}_{k-1}), \quad \breve{\mathbf{L}} = \begin{bmatrix} \breve{\ell}_1 & \cdots & \breve{\ell}_{k-1} \end{bmatrix},$$

it follows that $\mathbf{e}_i^T\left(\breve{\mu}_i \ \breve{\mathbb{L}} - \breve{\mathbf{M}} \ \breve{\mathbb{L}} - \ \breve{\mathbf{L}}\hat{\mathbb{W}}\right) = -\breve{\ell}_i\hat{\mathbb{W}}$, and we can hence write:

$$\hat{\mathbb{W}}\left(\breve{\mu}_i \ \breve{\mathbb{L}} - \breve{\mathbf{M}} \ \breve{\mathbb{L}} - \ \breve{\mathbf{L}}\hat{\mathbb{W}}\right)^{-1} = -\mathbf{e}_i^T/\breve{\ell}_i. \tag{36}$$

Finally, for all $1 \leq i \leq k-1$, the following holds true

$$\begin{aligned} \breve{\mathbf{H}}^{\lambda}(\breve{\mu}_i) &= w_k + (\breve{\mu}_i - \lambda_k)\mathbf{e}_i^T \ \bar{\mathbb{L}}\mathbf{e}_k/\breve{\ell}_i \\ &= w_k + (\breve{\mu}_i - \lambda_k) \ \bar{\mathbb{L}}_{i,k}/\breve{\ell}_i. \end{aligned}$$

This result provides the exact value that is matched, i.e. $\breve{v}_i = w_k + (\breve{\mu}_i - \lambda_k) \ \bar{\mathbb{L}}_{i,k}/\breve{\ell}_i$, for all $1 \leq i \leq k-1$.

*Remark 1.* Now, obviously, this interpolation value need not correspond to the evaluation of the original transfer function $\mathbf{H}(s)$ at the interpolation point $\breve{\mu}_i$, i.e.,

$$\mathbf{H}(\breve{\mu}_i) \neq \breve{v}_i. \tag{37}$$

However, as it was observed in practice (from various numerical examples, as shown also in Section 4.1), the values $\mathbf{H}(\breve{\mu}_i)$ and $\hat{\mathbf{H}}^{\lambda}(\breve{\mu}_i)$ are indeed similar (they are equal only when the original system is identified).

*Remark 2.* This property is relevant for the AAA framework since it shows that the systems fitted by AAA (see Section 2.2) are indeed (purely) interpolatory.

*Remark 3.* We note that the left compressed interpolation points are purely imaginary, provided that the original left ones are also purely imaginary. This follows from the identity $\tilde{\mathbf{M}} = \hat{\mathbf{X}}^*\mathbf{M}\hat{\mathbf{X}}$. This is mostly relevant for practical scenarios, for which frequency response data is typically the only (main) format available.

## 4. NUMERICAL EXAMPLES

*A toy example (small RLC network)* Consider the RLC circuit in Fig. 1. The control input $u$ is the voltage drop over the circuit, while the observed output $y$ is the current through the resistor $R$. The two state variables $x_1$ and $X_2$ are respectively, the voltage drop over the capacitor $C$, and the current through the inductor $L$. The following equations are derived from classical Kirchhoff circuit laws:

$$x_1 = L\dot{x}_2, \quad u = yR + x_1 \Rightarrow u = RCx_1 + Rx_2, \quad y = C\dot{x}_1 + x_2.$$

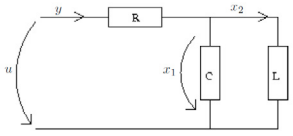


Fig. 1. A simple RLC circuit.

Hence, we are able to write a state-space representation of the underlying linear system that describes the dynamics:

$$\begin{cases} \dot{x}_1 = -\frac{1}{RC}x_1 - \frac{1}{C}x_2 + \frac{1}{RC}u \\ \dot{x}_2 = \frac{1}{L}x_2 \\ y = -\frac{1}{R}x_1 + \frac{1}{R}u \end{cases} \Rightarrow \begin{cases} \tilde{\mathbf{E}}\dot{\mathbf{x}} = \tilde{\mathbf{A}}\mathbf{x} + \tilde{\mathbf{B}}u \\ y = \tilde{\mathbf{C}}\mathbf{X} + \tilde{\mathbf{D}}u \end{cases}$$

By choosing $R = L = C = 1$ (for simplicity), the new system matrices can be written as

$$\mathbf{C} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}^T, \mathbf{E} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{A} = \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

and $\mathbf{D} = 1$. Note that the transfer function $\mathbf{H}(s)$ is computed as follows: $\mathbf{H}(s) = \frac{s^2+1}{s^2+s+1}$, $\forall\ s \in \mathbb{C}$. In what follows we discuss only the recovery of the linear model of the circuit (by means of the proposed methods), i.e., without reduction, which will be illustrated for a larger-scale benchmark example in Section 4.1.

*First experiment* In order to perfectly recover the original transfer function $\mathbf{H}(s)$, we need to choose $k = 3$ (three right interpolation points). Start by choosing $q = k-1 = 2$ in order to have a direct recovery (no compression on the left side). Let $\lambda_j = -j\imath$, where $\imath = \sqrt{-1}$, where $1 \le j \le 3$. Additionally, choose $q$ logarithmically-spaced left interpolation points in the interval $[0, 10]\imath$. Since $q = 2$, we have $\mu_1 = 1\imath$ and $\mu_2 = 10\imath$. The right data is given by

$$w_1 = 0, \ w_2 = \frac{3}{3+2\imath} \text{ and } w_2 = \frac{8}{8+3\imath}.$$

Next, compute a Loewner matrix $\mathbb{L} \in \mathbb{C}^{2\times 3}$ which is of rank $r = 2$ (the minimal number of variables needed to represent $\mathbf{H}(s)$). Note that $\mathbb{L}$ has indeed a right null-space given by vector $[y_1\ y_2\ y_3]^T \in \mathbb{C}^3$. Finally, the original TF is recovered exactly by computing $\hat{\mathbf{H}}^\lambda(s)$ as in (19). It is to be noted that the $\mathbf{D}$-term can be recovered as follows:

$$\mathbf{D}_{\text{rec}} = \frac{\sum_{j=1}^3 y_j w_j}{\sum_{j=1}^3 y_j} = 1 = \lim_{s\to\infty} \hat{\mathbf{H}}^\lambda(s). \tag{38}$$

*Second experiment* Next, we keep the same three right interpolation points as before, but we increase the number of left interpolation points (to 5, 50, 500, or even 5000), again logarithmically-spaced in the interval $[0, 10]\imath$. Then, we apply the procedure outlined in Section 3.2 to compute the two "compressed" points. The results are presented in Tab. 1. As shown there, as we are increasing the number of left interpolation points, it was observed that the two compressed values $\breve{\mu}_1$ and $\breve{\mu}_2$ tend to approach a particular value, i.e., stagnate with increasing the left points.

Table 1. The compressed left interpolation points.

| | $\breve{\mu}_1$ | $\breve{\mu}_2$ |
|---|---|---|
| $q = 5$ | $1.0530\imath$ | $3.2253\imath$ |
| $q = 50$ | $1.2215\imath$ | $3.3028\imath$ |
| $q = 500$ | $1.2422\imath$ | $3.3318\imath$ |
| $q = 5000$ | $1.2444\imath$ | $3.3348\imath$ |

### 4.1 MOR of a benchmark example (ISS 12A)

In this section, we analyze an established MOR benchmark example, i.e., the 12A component (Solar Arrays P3/P4) of the ISS (International Space Station) model. For more specific details on the model (the equations and how to derive them) we refer the reader to Gugercin et al. [2001]. In the numerical examples reported here, we have used the system matrices (MATLAB files) publicly available on the MOR repository in The MORwiki Community [2022]. The ISS12A model is a multi-input and multi-output linear time-invariant system (3 inputs and 3 outputs) characterized by $n = 1412$ state variables. In the experiments performed, we select the first input and first output, so that the analyzed model is written as in (1).

Since we are analyzing data-driven methods, the first step is to compute data. We start by choosing 200 interpolation points, logarithmically-spaced in the interval $[0.4, 40]\imath$, i.e., a range in the frequency response containing dominant oscillations. We sample the transfer function of the original large-scale model at these value, and obtain "the data". Using this set of measurements, we will compute three ROMs (of order $r = k - 1$, where $k = 40$ and $q = 160$) by means of the following methods:

(1) The Loewner framework outlined in Section 2.1 (LF);
(2) The AAA algorithm outlined in Section 2.2 (AAA);
(3) The one-sided LF introduced in Section 3.2 (OSLF).

First, we apply the OSLF framework as presented in Section 3.2, for $k = 40$ and $q = 160$ (in total 200 data points). Hence, the right data set contains 20% of the data values, while the left data set accounts for the remaining 80%. We use a sub-sampling approach for splitting the data set: one in five measurements pertains to the left data.

We compute a rational approximant on the data, denoted with $\breve{\mathbf{H}}^\lambda(s)$, as given by (33). The results are presented in Fig. 2. Clearly, by construction, the right data values are perfectly matched (as depicted in the upper part). Next, we show the fit on the left data set in the lower part of Fig. 2. We conclude that a reasonable fit was achieved.

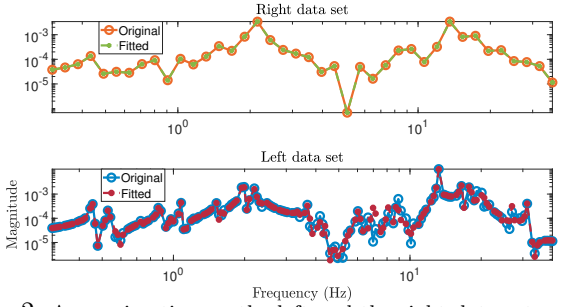


Fig. 2. Approximation on the left and the right data sets.

Next, we compute the left compressed interpolation points $\{\mu_i | 1 \le i \le k-1\}$ as in Definition 4. We evaluate $\breve{\mathbf{H}}^\lambda(s)$

at these points and compare these values with the original transfer function evaluations (to $\mathbf{H}(\mu_i)$ for $1 \leq i \leq k-1$). We depict the magnitude of these two quantities in Fig. 3. As expected, the values are indeed similar, in particular the ones of higher magnitude.

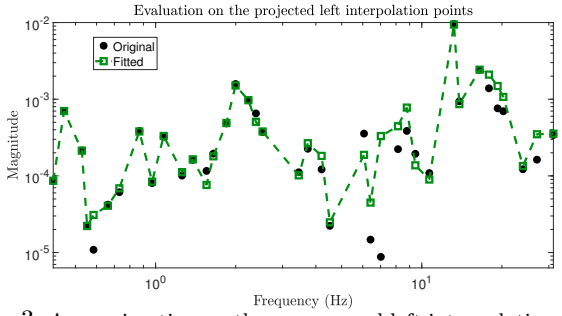


Fig. 3. Approximation on the compressed left interpolations points.

Next, we choose 400 logaritmically-spaced data points in the interval $[0.1, 100]\imath$. We evaluate the original transfer function $\mathbf{H}(s)$ at these points, and compare these samples to those computed by evaluating the transfer functions of the 3 ROMs. The results are depicted in Fig. 4. It can be observed that all three approximated transfer functions faithfully approximate the original response; in particular, the high amplitude peaks are accurately matched.

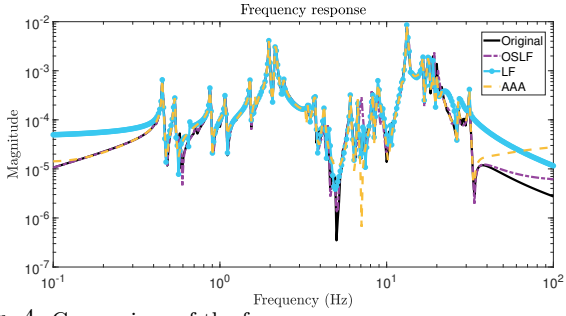


Fig. 4. Comparison of the frequency responses.

Finally, we depict the approximation errors generated by the ROMs computed with the 3 methods under consideration. Comparing the largest magnitude, all 3 error curves provide comparable results. However, in the high/low frequency ranges, OSLF seems to yield better approximation.

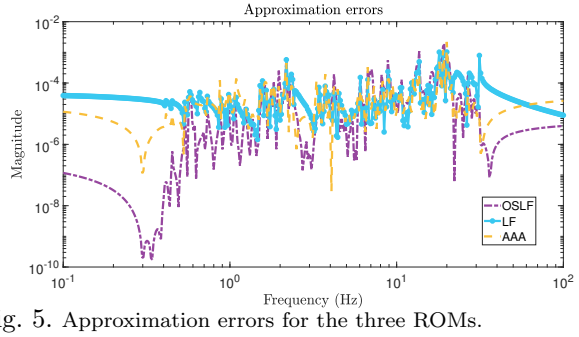


Fig. 5. Approximation errors for the three ROMs.

One bottleneck for LF is that it relies on a full SVD of potentially-large matrices while AAA relies on a heuristic for choosing right interpolation points, hence being quite fast (since it deals with "tall and skinny" Loewner matrices). On the other hand, the chosen heuristic may not always be optimal for the application under hand. For the OSLF (and LF), we need to rely on a particular data partition as in (2), that is decided a priori. This may constitute an advantage (to explicitly interpolate at measurements unaffected by noise, or of high magnitude), but could also be challenging when an obvious choice lacks.

## 5. CONCLUSION

We have proposed a data-driven model reduction method that is an extension of the classical (double-sided) Loewner framework. The new method allows to explicitly process data subsets with different dimension. In particular, we analyzed the case for which the left data set is larger than the right set, and interpolation is explicitly enforced on the latter. This approach has been performed before, being one kez concept of the AAA algorithm. Additionally, we have explicitly described the extra interpolation conditions (on the left side), that the rational approximant enforces. Finally, we have illustrated the theoretical findings on two numerical examples. Future research could include further elaborating on the connections to AAA, characterizing the compressed left/right points in more depth, imposing stability of the computed models, and finding parallels to the one-sided method in Astolfi [2010].

## REFERENCES

Antoulas, A.C. (2005). *Approximation of large-scale dynamical systems*. SIAM, Philadelphia.

Antoulas, A.C. and Anderson, B.D.O. (1986). On the scalar rational interpolation problem. *IMA Journal of Mathematical Control and Information*, 3(2-3), 61–88.

Antoulas, A.C., Beattie, C.A., and Gugercin, S. (2020). *Interpolatory Methods for Model Reduction*. Computational Science & Engineering. Society for Industrial and Applied Mathematics, Philadelphia, PA.

Antoulas, A.C., Gosea, I.V., and Ionita, A.C. (2016). Model reduction of bilinear systems in the Loewner framework. *SIAM Journal on Scientific Computing*, 38(5), B889–B916.

Antoulas, A.C., Lefteriu, S., and Ionita, A.C. (2017). A tutorial introduction to the Loewner framework for model reduction. In *Model Reduction and Approximation*, chapter 8, 335–376. SIAM.

Antoulas, A.C. (2016). The Loewner framework and transfer functions of singular/rectangular systems. *Applied Mathematics Letters*, 54, 36–47.

Astolfi, A. (2010). Model reduction by moment matching for linear and nonlinear systems. *Transactions on Automatic Control*, 55(10), 2321–2336. doi:10.1109/TAC.2010.2046044.

Benner, P., Ohlberger, M., Cohen, A., and Willcox, K. (2017). *Model Reduction and Approximation*. Society for Industrial and Applied Mathematics, Philadelphia, PA. doi:10.1137/1.9781611974829.

Berrut, J.P. and Trefethen, L.N. (2004). Barycentric lagrange interpolation. *SIAM review*, 46(3), 501–517.

Gosea, I.V. and Güttel, S. (2021). Algorithms for the rational approximation of matrix-valued functions. *SIAM Journal on Scientific Computing*, 43(5), A3033–A3054.

Gugercin, S., Antoulas, A.C., and Bedrossian, M. (2001). Approximation of the international space station 1R and 12A flex models. In *Proceedings of the IEEE CDC*, 1515–1516.

Karachalios, D., Gosea, I.V., and Antoulas, A.C. (2021). The Loewner framework for system identification and reduction. In *Model Order Reduction: Volume I: System-and Data-Driven Methods and Algorithms*, 181–228. De Gruyter.

Mayo, A.J. and Antoulas, A.C. (2007). A framework for the solution of the generalized realization problem. *Linear Algebra and Its Applications*, 425(2-3), 634–662.

Nakatsukasa, Y., Sete, O., and Trefethen, L.N. (2018). The AAA algorithm for rational approximation. *SIAM Journal on Scientific Computing*, 40(3), A1494–A1522.

Peherstorfer, B. and Willcox, K. (2016). Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306, 196–215. doi:10.1016/j.cma.2016.03.025.

The MORwiki Community (2022). MORwiki - Model Order Reduction Wiki. `http://modelreduction.org`.