Advances in Computational Mathematics (2024) 50:79
https://doi.org/10.1007/s10444-024-10166-z

Adaptive choice of near-optimal expansion points for interpolation-based structure-preserving model reduction

Quirin Aumann[1] · **Steffen W. R. Werner**[2]

Received: 30 April 2023 / Accepted: 10 June 2024

Abstract
Interpolation-based methods are well-established and effective approaches for the efficient generation of accurate reduced-order surrogate models. Common challenges for such methods are the automatic selection of good or even optimal interpolation points and the appropriate size of the reduced-order model. An approach that addresses the first problem for linear, unstructured systems is the iterative rational Krylov algorithm (IRKA), which computes optimal interpolation points through iterative updates by solving linear eigenvalue problems. However, in the case of preserving internal system structures, optimal interpolation points are unknown, and heuristics based on nonlinear eigenvalue problems result in numbers of potential interpolation points that typically exceed the reasonable size of reduced-order systems. In our work, we propose a projection-based iterative interpolation method inspired by IRKA for generally structured systems to adaptively compute near-optimal interpolation points as well as an appropriate size for the reduced-order system. Additionally, the iterative updates of the interpolation points can be chosen such that the reduced-order model provides an accurate approximation in specified frequency ranges of interest. For such applications, our new approach outperforms the established methods in terms of accuracy and computational effort. We show this in numerical examples with different structures.

Keywords Dynamical systems · Model order reduction · Structure preservation · Structured interpolation · Projection methods

Communicated by: Tobias Breiten

✉ Quirin Aumann
aumann@mpi-magdeburg.mpg.de

Steffen W. R. Werner
steffen.werner@vt.edu

[1] Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany

[2] Department of Mathematics and Division of Computational Modeling and Data Analytics, Academy of Data Science, Virginia Tech, Blacksburg, VA 24061, USA

Published online: 19 July 2024

Mathematics Subject Classification (2010) 30E05 · 41A30 · 65D05 · 93A15 · 93C80

1 Introduction

Simulation, control, and optimization of dynamical systems are essential for many applications. In this work, we consider structured linear dynamical systems in the frequency (Laplace) domain of the form

$$\Sigma\colon \begin{cases} \mathcal{K}(s)\boldsymbol{X}(s) = \mathcal{B}(s)\boldsymbol{U}(s), \quad \boldsymbol{Y}(s) = \mathcal{C}(s)\boldsymbol{X}(s), \end{cases} \tag{1}$$

where $\mathcal{K}\colon \mathbb{C} \to \mathbb{C}^{n\times n}$ describes the system dynamics, $\mathcal{B}\colon \mathbb{C} \to \mathbb{C}^{n\times m}$ the system's input, and $\mathcal{C}\colon \mathbb{C} \to \mathbb{C}^{p\times n}$ the system's output behavior; see [1] for the motivation of (1) and example systems of this form. The functions $\boldsymbol{X}\colon \mathbb{C} \to \mathbb{C}^n$, $\boldsymbol{U}\colon \mathbb{C} \to \mathbb{C}^m$, and $\boldsymbol{Y}\colon \mathbb{C} \to \mathbb{C}^p$ denote the internal states, inputs, and outputs, respectively. For all $s \in \mathbb{C}$ for which $\mathcal{K}$ is invertible, and $\mathcal{B}$ and $\mathcal{C}$ can be evaluated, the corresponding transfer function $\boldsymbol{H}\colon \mathbb{C} \to \mathbb{C}^{p\times m}$ directly relates the system's inputs to outputs:

$$\boldsymbol{H}(s) = \mathcal{C}(s)\mathcal{K}(s)^{-1}\mathcal{B}(s). \tag{2}$$

The most commonly considered structure of dynamical systems is given by equations that are linear in the frequency variable

$$\Sigma\colon \begin{cases} (s\boldsymbol{E} - \boldsymbol{A})\boldsymbol{X}(s) = \boldsymbol{B}\boldsymbol{U}(s), \quad \boldsymbol{Y}(s) = \boldsymbol{C}\boldsymbol{X}(s), \end{cases} \tag{3}$$

with the system matrices $\boldsymbol{A}, \boldsymbol{E} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times m}$, and $\boldsymbol{C} \in \mathbb{R}^{p\times n}$. Systems of the form (3) are also commonly referred to as *unstructured systems* due to them being considered as the standard case. Considering the general system class (1), we have

$$\mathcal{K}(s) = (s\boldsymbol{E} - \boldsymbol{A})^{-1}, \quad \mathcal{B}(s) = \boldsymbol{B}, \quad \mathcal{C}(s) = \boldsymbol{C},$$

and the corresponding transfer function $\boldsymbol{H}(s) = \boldsymbol{C}(s\boldsymbol{E} - \boldsymbol{A})^{-1}\boldsymbol{B}$. On the other hand, the modeling of specific physical phenomena hands down other differential structures into dynamical systems. The modeling of mechanical structures, structural vibrations, wave movement, or electrical circuits classically leads to transfer functions of the form

$$\boldsymbol{H}(s) = (\boldsymbol{C}_\mathrm{p} + s\boldsymbol{C}_\mathrm{v})(s^2\boldsymbol{M} + s\boldsymbol{D} + \boldsymbol{K})^{-1}\boldsymbol{B}; \tag{4}$$

see, for example, [2–4] and references therein. A different structure occurs in the modeling of incomplete systems resulting in delays, which are expressed as exponential terms in the frequency domain, e.g., with the transfer function

$$\boldsymbol{H}(s) = \boldsymbol{C}(s\boldsymbol{E} - \boldsymbol{A}_0 - \mathrm{e}^{-\tau s}\boldsymbol{A}_\mathrm{d})^{-1}\boldsymbol{B}, \tag{5}$$

for some constant delay $\tau > 0$; see, e.g., [5]. Many other structures exist in the literature, which are used to model, for example, poroelasticity [6], viscoelasticity [7], or interior acoustic problems [8]. While some structures, like second-order systems (4), can be reformulated into standard form (3), this is not necessarily possible for all occurring structures, including time-delay systems (5). Also, by reformulation into unstructured form, the number of states increases and structure-inherent properties are typically lost in subsequent computational procedures.

In general, there is a demand for highly accurate models in practical applications. As a result, the number of equations describing (1) vastly grows and the computational efficient solution of (1) in terms of resources such as time and memory is often impossible. Model order reduction methods are a remedy to this problem as they aim for the construction of cheap-to-evaluate yet accurate surrogate models that approximate the systems' input-to-output behavior while being described by a significantly smaller number of equations $r \ll n$, which eases the demand on computational resources required for the evaluation of the systems. Many model reduction techniques have been developed for unstructured systems (3); see, for example, [9]. In addition, the preservation of internal system structures such as (4) and (5) is desired as this typically yields more accurate approximations as well as the preservation of structure-inherent properties. Also, if the reduced-order model is to be coupled to other systems, preserving the structure is advantageous because the same coupling conditions as for the full-order model can be applied to the reduced surrogate [10].

Several structure-preserving model order reduction methods have been developed in recent years. Many of these have been tailored to particular structures that occur, for example, in vibrational problems [3, 11–13], network systems [14, 15], or systems with Hamiltonian structure [16–18]. The framework in [1] allows the reduction of dynamical systems with arbitrary internal structures based on transfer function interpolation. The quality of reduced-order models obtained by interpolation strongly depends on the choice of interpolation points. Therefore, a variety of strategies has been developed to perform successive greedy searches for suitable interpolation points based on estimating the approximation error [19–25] or computing the exact error in, for example, the $\mathcal{L}_\infty$-norm [26, 27].

On the other hand, the *iterative rational Krylov algorithm* (`IRKA`) is a well-established interpolation method for unstructured systems (3) that iteratively updates the interpolation points [28]. At convergence, the interpolating reduced-order model satisfies the necessary $\mathcal{H}_2$-optimality conditions. Several extensions of `IRKA` for structured systems using similar ideas have been proposed. For second-order systems (4), the `SO-IRKA` method from [29] aims for an iterative process similar to `IRKA`. In [30], this has been considered the basis for a method to choose the resulting approximation order adaptively. The *Transfer Function IRKA* (`TF-IRKA`) [31] can be applied to arbitrarily structured systems and yields $\mathcal{H}_2$-optimal but unstructured reduced-order models. A structure-preserving variant of `TF-IRKA` has been proposed in [32].

A different take on structured model order reduction is data-driven methods. Since here only measurements of the transfer function (2) are used to compute realizations of dynamical systems, the original structure can be arbitrary. One of the most well-known approaches of this type is the Loewner framework, which constructs a reduced-order model that interpolates provided data samples [33]. The original formulation of the

Loewner framework only considers the construction of unstructured systems (3), but it has been extended to find structured realizations in [34]. Recently, structured extensions of the barycentric form for second-order systems (4) have been proposed that allow the extension of further data-driven frequency domain methods to the structure-preserving setting [35, 36].

In this work, we present a new approach to compute accurate reduced-order models that preserve the internal structure of the original system. Based on an `IRKA`-like iteration scheme, the new method computes in every step a new set of interpolation points (and tangential directions) which are then employed in the structure-preserving interpolation framework [1]. Instead of considering nonlinear eigenvalue problems corresponding to the resolvent terms of the structured systems, the Loewner framework allows us to solve linear eigenvalue problems in each step and to determine the approximation order adaptively and with respect to limited frequency regions of interest if desired.

The remainder of this manuscript is structured as follows: After introducing the mathematical preliminaries in Section 2, we revisit the structure-preserving transfer function `IRKA` and extend that method to the case of multiple-input/multiple-output systems in Section 3.1. Our new model reduction method is then described in Section 3.2. In Section 4, a number of numerical experiments are used to compare the new method to established model reduction techniques. The paper is concluded in Section 5.

2 Mathematical preliminaries

2.1 Structure-preserving interpolation via projection

We consider here interpolation-based model order reduction methods, which compute surrogate models approximating the dynamics of the high-fidelity system (1) while having much smaller dimensions $r \ll n$. Structure-preserving model order reduction methods construct approximations of (1) with the same internal structure

$$\widehat{\Sigma}\colon \begin{cases} \widehat{\mathcal{K}}(s)\widehat{\boldsymbol{X}}(s) = \widehat{\mathcal{B}}(s)\boldsymbol{U}(s), \quad \widehat{\boldsymbol{Y}}(s) = \widehat{\mathcal{C}}(s)\widehat{\boldsymbol{X}}(s), \end{cases} \tag{6}$$

where $\widehat{\mathcal{K}}\colon \mathbb{C} \to \mathbb{C}^{r\times r}$, $\widehat{\mathcal{B}}\colon \mathbb{C} \to \mathbb{C}^{r\times m}$, $\widehat{\mathcal{C}}\colon \mathbb{C} \to \mathbb{C}^{p\times r}$, and $\widehat{\boldsymbol{X}}\colon \mathbb{C} \to \mathbb{C}^{r}$, $\widehat{\boldsymbol{Y}}\colon \mathbb{C} \to \mathbb{C}^{p}$. Additionally, the compositions of the matrix-valued functions in (1) and (6) are the same: If the center term in (1) is given in frequency-affine form

$$\mathcal{K}(s) = \sum_{j=1}^{n_{\mathcal{K}}} g_j(s)\mathcal{K}_j, \tag{7}$$

with $g_j\colon \mathbb{C} \to \mathbb{C}$ and constant matrices $\mathcal{K}_j \in \mathbb{C}^{n\times n}$, for $j = 1, \ldots, n_{\mathcal{K}}$, then the center term of the reduced-order model must have the form

$$\widehat{\mathcal{K}}(s) = \sum_{j=1}^{n_{\mathcal{K}}} g_j(s)\widehat{\mathcal{K}}_j, \tag{8}$$

where $\widehat{\mathcal{K}}_j \in \mathbb{C}^{r\times r}$, for $j = 1, \ldots, n_{\mathcal{K}}$. Since the scalar functions in (7) and (8) are identical, the internal system structure is preserved and the system matrices $\mathcal{K}_j$ of the full-order system can be replaced by their reduced-order counterparts $\widehat{\mathcal{K}}_j$. The same relations must hold for the input and output terms $\mathcal{B}$ and $\mathcal{C}$. Consider as an example the second-order system with transfer function (4) from Section 1. A structure-preserving reduced-order model will be of the form

$$\widehat{\boldsymbol{H}}(s) = (\widehat{\boldsymbol{C}}_{\mathrm{p}} + s\widehat{\boldsymbol{C}}_{\mathrm{v}})(s^2\widehat{\boldsymbol{M}} + s\widehat{\boldsymbol{D}} + \widehat{\boldsymbol{K}})^{-1}\widehat{\boldsymbol{B}}.$$

To act as a suitable surrogate, the reduced-order model must approximate the input-to-output behavior of the original system at least for some $s \in \mathbb{C}$, which are important for the application in question. In other words, the outputs of (1) and (6) should match up to a specified tolerance τ in appropriate norms for a given input:

$$\|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}\| \leq \tau \cdot \|\boldsymbol{U}\|.$$

The relation above can be reformulated in terms of the transfer functions of the original and reduced-order model such that

$$\|\boldsymbol{H} - \widehat{\boldsymbol{H}}\| \leq \tau$$

holds.

Following [1], any matrix-valued function of the form (2) can be interpolated by a reduced-order transfer function $\widehat{\boldsymbol{H}}$, while preserving the internal system structure using the projection approach. Given two reduction spaces with basis matrices $\boldsymbol{V}, \boldsymbol{W} \in \mathbb{C}^{n\times r}$, the reduced-order model is computed by

$$\widehat{\mathcal{K}}(s) = \boldsymbol{W}^{\mathsf{H}}\mathcal{K}(s)\boldsymbol{V}, \quad \widehat{\mathcal{B}}(s) = \boldsymbol{W}^{\mathsf{H}}\mathcal{B}(s), \quad \widehat{\mathcal{C}}(s) = \mathcal{C}(s)\boldsymbol{V}. \tag{9}$$

While there are many potential choices for the basis matrices $\boldsymbol{V}$ and $\boldsymbol{W}$, we concentrate here on transfer function interpolation, i.e., the matrices $\boldsymbol{V}, \boldsymbol{W}$ are constructed such that the transfer function $\widehat{\boldsymbol{H}}$ corresponding to (9) interpolates the full-order transfer function (2) at chosen points. The following proposition gives a concise overview.

Proposition 1 (Structured interpolation [1, Thm. 1]) *Let $\boldsymbol{H}$ be the transfer function (2) of a linear system, described by (1), and $\widehat{\boldsymbol{H}}$ the reduced-order transfer function constructed via projection (9). Let the matrix functions $\mathcal{C}$, $\mathcal{K}^{-1}$, $\mathcal{B}$, and $\widehat{\mathcal{K}}^{-1}$ be analytic in the interpolation point $\sigma \in \mathbb{C}$. Then, the following statements hold.*

(a) If $\operatorname{span}\left(\mathcal{K}(\sigma)^{-1}\mathcal{B}(\sigma)\right) \subseteq \operatorname{span}(\boldsymbol{V})$ *holds, then* $\boldsymbol{H}(\sigma) = \widehat{\boldsymbol{H}}(\sigma)$.
(b) If $\operatorname{span}\left(\mathcal{K}(\sigma)^{-\mathsf{H}}\mathcal{C}(\sigma)^{\mathsf{H}}\right) \subseteq \operatorname{span}(\boldsymbol{W})$ *holds, then* $\boldsymbol{H}(\sigma) = \widehat{\boldsymbol{H}}(\sigma)$.
(c) If $\boldsymbol{V}$ and $\boldsymbol{W}$ are constructed as above, then additionally $\boldsymbol{H}'(\sigma) = \widehat{\boldsymbol{H}}'(\sigma)$ *holds.*

Overall, only linear systems of equations need to be solved for the construction of the basis matrices $\boldsymbol{V}$ and $\boldsymbol{W}$ in Proposition 1. However, the interpolation point σ has to be known beforehand, and its choice has a large influence on the approximation

quality of the resulting reduced-order model. Traditionally, points are chosen linearly or logarithmically equidistant on the frequency axis $\mathrm{i}\mathbb{R}$ in frequency ranges of interest to reduce the worst case approximation error of the transfer function given by the $\mathcal{L}_\infty$-norm. This typically leads to a reasonable approximation behavior over the considered frequency range but easily misses features of the system, which are not close enough to the interpolation points, or may result in unnecessarily large reduced-order models.

2.2 Unstructured interpolation via the Loewner framework

Independent of the structure of the original system, the Loewner framework can be used to construct unstructured systems from transfer function evaluations [33, 37]. As we will use this framework at several points throughout this manuscript, it is summarized below following the description in [37].

Given $2q$ transfer function measurements $\boldsymbol{H}_k := \boldsymbol{H}(s_k) \in \mathbb{C}^{p\times m}$ at some locations $s_k \in \mathbb{C}$, for $k = 1, \ldots, 2q$, the data is partitioned into two sets

$$\begin{cases} (\kappa_i, \boldsymbol{r}_i, \boldsymbol{w}_i), \text{ where } \kappa_i = s_i,\ \boldsymbol{w}_i = \boldsymbol{H}_i \boldsymbol{r}_i, \text{ for } i = 1, \ldots, q, \\ (\mu_j, \boldsymbol{\ell}_j, \boldsymbol{v}_j), \text{ where } \mu_j = s_{q+j},\ \boldsymbol{v}_j^{\mathsf{H}} = \boldsymbol{\ell}_j^{\mathsf{H}} \boldsymbol{H}_{q+j}, \text{ for } j = 1, \ldots, q, \end{cases}$$

with right and left tangential directions $\boldsymbol{r}_i \in \mathbb{C}^m$ and $\boldsymbol{\ell}_j \in \mathbb{C}^p$, for $i, j = 1, \ldots, q$. In practice, it has been shown to be beneficial for numerical reasons to partition the data in an alternating way with respect to the ordering of the absolute values of the sampling points. Under the assumption that the sets of sampling points are disjoint, $\{\kappa_i\}_{i=1}^q \cap \{\mu_j\}_{j=1}^q = \emptyset$, the partitioned data is arranged in the Loewner and shifted Loewner matrices

$$\mathbb{L} = \begin{bmatrix} \dfrac{\boldsymbol{v}_1^{\mathsf{H}} \boldsymbol{r}_1 - \boldsymbol{\ell}_1^{\mathsf{H}} \boldsymbol{w}_1}{\mu_1 - \kappa_1} & \cdots & \dfrac{\boldsymbol{v}_1^{\mathsf{H}} \boldsymbol{r}_q - \boldsymbol{\ell}_1^{\mathsf{H}} \boldsymbol{w}_q}{\mu_1 - \kappa_q} \\ \vdots & \ddots & \vdots \\ \dfrac{\boldsymbol{v}_q^{\mathsf{H}} \boldsymbol{r}_1 - \boldsymbol{\ell}_q^{\mathsf{H}} \boldsymbol{w}_1}{\mu_q - \kappa_1} & \cdots & \dfrac{\boldsymbol{v}_q^{\mathsf{H}} \boldsymbol{r}_q - \boldsymbol{\ell}_q^{\mathsf{H}} \boldsymbol{w}_q}{\mu_q - \kappa_q} \end{bmatrix}, \tag{10}$$

$$\mathbb{L}_\sigma = \begin{bmatrix} \dfrac{\mu_1 \boldsymbol{v}_1^{\mathsf{H}} \boldsymbol{r}_1 - \kappa_1 \boldsymbol{\ell}_1^{\mathsf{H}} \boldsymbol{w}_1}{\mu_1 - \kappa_1} & \cdots & \dfrac{\mu_1 \boldsymbol{v}_1^{\mathsf{H}} \boldsymbol{r}_q - \kappa_q \boldsymbol{\ell}_1^{\mathsf{H}} \boldsymbol{w}_q}{\mu_1 - \kappa_q} \\ \vdots & \ddots & \vdots \\ \dfrac{\mu_q \boldsymbol{v}_q^{\mathsf{H}} \boldsymbol{r}_1 - \kappa_1 \boldsymbol{\ell}_q^{\mathsf{H}} \boldsymbol{w}_1}{\mu_q - \kappa_1} & \cdots & \dfrac{\mu_q \boldsymbol{v}_q^{\mathsf{H}} \boldsymbol{r}_q - \kappa_q \boldsymbol{\ell}_q^{\mathsf{H}} \boldsymbol{w}_q}{\mu_q - \kappa_q} \end{bmatrix}. \tag{11}$$

If the matrix pencil $\mathbb{L}_\sigma - \lambda\mathbb{L}$ is regular, i.e., there exists a $\lambda \in \mathbb{C}$ such that $\det(\mathbb{L}_\sigma - \lambda\mathbb{L}) \neq 0$, assuming that the sampling points $\{\kappa_i\}_{i=1}^q$ and $\{\mu_j\}_{j=1}^q$ are not

eigenvalues of the matrix pencil $\mathbb{L}_\sigma - \lambda\mathbb{L}$, and given the matrices

$$\boldsymbol{W}_\mathbb{L} = \begin{bmatrix} \boldsymbol{w}_1 & \dots & \boldsymbol{w}_q \end{bmatrix} \quad \text{and} \quad \boldsymbol{V}_\mathbb{L} = \begin{bmatrix} \boldsymbol{v}_1^\mathsf{H} \\ \vdots \\ \boldsymbol{v}_q^\mathsf{H} \end{bmatrix}, \tag{12}$$

the transfer function of the form $\boldsymbol{H}_\mathbb{L}(s) = \boldsymbol{W}_\mathbb{L}(\mathbb{L}_\sigma - s\mathbb{L})^{-1}\boldsymbol{V}_\mathbb{L}$ tangentially interpolates the given data such that $\boldsymbol{H}_\mathbb{L}(\kappa_i)\boldsymbol{r}_i = \boldsymbol{w}_i$ and $\boldsymbol{\ell}_j^\mathsf{H}\boldsymbol{H}_\mathbb{L}(\mu_j) = \boldsymbol{v}_j^\mathsf{H}$ hold, for $i, j = 1, \dots, q$. The corresponding state-space realization of the underlying dynamical system is then given by

$$\boldsymbol{E}_\mathbb{L} := -\mathbb{L}, \quad \boldsymbol{A}_\mathbb{L} := -\mathbb{L}_\sigma, \quad \boldsymbol{B}_\mathbb{L} := \boldsymbol{V}_\mathbb{L}, \quad \boldsymbol{C}_\mathbb{L} := \boldsymbol{W}_\mathbb{L}, \tag{13}$$

using the matrices from (10), (11), and (12).

In the case that $n_\mathbb{L} = \operatorname{rank}(\zeta\mathbb{L} - \mathbb{L}_\sigma) = \operatorname{rank}\left(\begin{bmatrix}\mathbb{L} & \mathbb{L}_\sigma\end{bmatrix}\right) = \operatorname{rank}\left(\begin{bmatrix}\mathbb{L} \\ \mathbb{L}_\sigma\end{bmatrix}\right)$ holds for all $\zeta \in \{\kappa_i\}_{i=1}^q \cup \{\mu_i\}_{i=1}^q$, the number $n_\mathbb{L}$ is the minimal order of the model, to which the realization needs to be truncated to satisfy the regularity condition [33, 37]. In practice, it is reasonable to truncate any redundant data, which might have been collected into $(-\mathbb{L}, -\mathbb{L}_\sigma, \boldsymbol{V}_\mathbb{L}, \boldsymbol{W}_\mathbb{L})$. The required truncation matrices $\boldsymbol{V}$ and $\boldsymbol{W}$ can be chosen based on the left and right singular vectors obtained from singular value decompositions (SVDs) of $\begin{bmatrix}\mathbb{L} & \mathbb{L}_\sigma\end{bmatrix}$ and $\begin{bmatrix}\mathbb{L}^\mathsf{H} & \mathbb{L}_\sigma^\mathsf{H}\end{bmatrix}^\mathsf{H}$. Truncating the matrices of singular vectors at $n_\mathbb{L}$ columns and projecting the Loewner realization (13) yields a model interpolating the given data. Truncating after $r_\mathbb{L} < n_\mathbb{L}$ results in a model approximating the provided data.

In general, without further modifications, the models obtained from the Loewner framework may have complex matrices. However, many systems are described by real matrices in practical applications. Under the assumption that the original transfer function follows the *reflection principle*, i.e., $\overline{\boldsymbol{H}(s)} = \boldsymbol{H}(\overline{s})$ holds for all $s \in \mathbb{C}$ for which $\boldsymbol{H}$ is defined, sampling points as well as transfer function data and tangential directions can be chosen closed under conjugation, i.e., if κ is a sampling point so is $\overline{\kappa}$, and $\boldsymbol{H}(\kappa)$ and $\boldsymbol{H}(\overline{\kappa}) = \overline{\boldsymbol{H}(\kappa)}$ are the corresponding complex conjugate transfer function measurements. In this case, there exists a state-space transformation for (13) that yields real matrices. Assuming that all given data is complex, closed under conjugation, and ordered into complex pairs, then the transformation to obtain real matrices is given by

$$\boldsymbol{J} = \boldsymbol{I}_q \otimes \left(\frac{1}{\sqrt{2}}\begin{bmatrix}1 & -\mathrm{i} \\ 1 & \mathrm{i}\end{bmatrix}\right),$$

and the transformed system $(-\boldsymbol{J}^\mathsf{H}\mathbb{L}\boldsymbol{J},\ -\boldsymbol{J}^\mathsf{H}\mathbb{L}_\sigma\boldsymbol{J},\ \boldsymbol{J}^\mathsf{H}\boldsymbol{V}_\mathbb{L},\ \boldsymbol{W}_\mathbb{L}\boldsymbol{J})$ has real matrices and satisfies the same interpolation conditions as the original Loewner system (13); see [37].

2.3 Unstructured $\mathcal{H}_2$-optimal interpolation

A different approach for the construction of reduced-order models for structured systems is the *Transfer Function IRKA* (`TF-IRKA`) from [31]. Like the original `IRKA` [28], this method computes $\mathcal{H}_2$-optimal approximations but can also be applied to structured systems like (1), because only evaluations of the transfer function and its derivative are needed for the algorithm. However, `TF-IRKA` computes a reduced-order model of first-order form (3), i.e., the approach can be applied to structured systems but does not preserve the structure in the reduced-order model.

The procedure of `TF-IRKA` is as follows: Instead of computing an interpolating realization of the reduced-order model by projection, an interpolating first-order realization is obtained using the Loewner framework [33] in every iteration step. In contrast to the variant of the Loewner framework described in the previous section, the two sets of interpolation points are chosen to be identical. This leads to a modification of the formulas (10) and (11) involving the derivative of the sampled transfer function. Given a transfer function $\boldsymbol{H}(s)$, its derivative $\boldsymbol{H}'(s)$, interpolation points $\{\sigma_j\}_{j=1}^r$, and right and left tangential directions $\{\boldsymbol{b}_j\}_{j=1}^r$ and $\{\boldsymbol{c}_j\}_{j=1}^r$, with $\boldsymbol{b}_j \in \mathbb{C}^m$ and $\boldsymbol{c}_j \in \mathbb{C}^p$, this variant of the Loewner framework constructs a first-order model $\boldsymbol{H}_{\mathbb{L}}(s) = \boldsymbol{C}_{\mathbb{L}}\,(s\boldsymbol{E}_{\mathbb{L}} - \boldsymbol{A}_{\mathbb{L}})^{-1}\,\boldsymbol{B}_{\mathbb{L}}$ satisfying the following tangential Hermite interpolation conditions:

$$\boldsymbol{H}(\sigma_j)\boldsymbol{b}_j = \boldsymbol{H}_{\mathbb{L}}(\sigma_j)\boldsymbol{b}_j, \quad \boldsymbol{c}_j^{\mathsf{H}}\boldsymbol{H}(\sigma_j) = \boldsymbol{c}_j^{\mathsf{H}}\boldsymbol{H}_{\mathbb{L}}(\sigma_j), \quad \boldsymbol{c}_j^{\mathsf{H}}\boldsymbol{H}'(\sigma_j)\boldsymbol{b}_j = \boldsymbol{c}_j^{\mathsf{H}}\boldsymbol{H}'_{\mathbb{L}}(\sigma_j)\boldsymbol{b}_j,$$

for all $j = 1, \ldots, r$. The entries of the matrices in the Loewner realization are constructed via

$$(\boldsymbol{E}_{\mathbb{L}})_{i,j} := \begin{cases} -\dfrac{\boldsymbol{c}_i^{\mathsf{H}}\big(\boldsymbol{H}(\sigma_i) - \boldsymbol{H}(\sigma_j)\big)\boldsymbol{b}_j}{\sigma_i - \sigma_j} & \text{if } i \neq j, \\ -\boldsymbol{c}_i^{\mathsf{H}}\boldsymbol{H}'(\sigma_i)\boldsymbol{b}_i & \text{if } i = j, \end{cases} \tag{14}$$

$$(\boldsymbol{A}_{\mathbb{L}})_{i,j} := \begin{cases} -\dfrac{\boldsymbol{c}_i^{\mathsf{H}}\big(\sigma_i\boldsymbol{H}(\sigma_i) - \sigma_j\boldsymbol{H}(\sigma_j)\big)\boldsymbol{b}_j}{\sigma_i - \sigma_j} & \text{if } i \neq j, \\ -\boldsymbol{c}_i^{\mathsf{H}}\big(s\boldsymbol{H}(s)\big)'(\sigma_i)\boldsymbol{b}_i & \text{if } i = j, \end{cases} \tag{15}$$

$$\boldsymbol{B}_{\mathbb{L}} := \begin{bmatrix} \boldsymbol{c}_1^{\mathsf{H}}\boldsymbol{H}(\sigma_1) \\ \vdots \\ \boldsymbol{c}_r^{\mathsf{H}}\boldsymbol{H}(\sigma_r) \end{bmatrix} \quad \text{and} \quad \boldsymbol{C}_{\mathbb{L}} := \begin{bmatrix} \boldsymbol{H}(\sigma_1)\boldsymbol{b}_1 & \ldots & \boldsymbol{H}(\sigma_r)\boldsymbol{b}_r \end{bmatrix}. \tag{16}$$

Similar to the classical `IRKA` method, the eigenvectors and mirror images of the eigenvalues of $\boldsymbol{A}_{\mathbb{L}} - \lambda\boldsymbol{E}_{\mathbb{L}}$ with respect to the imaginary axis are used as interpolation points and tangential directions in the next iteration step of `TF-IRKA`. At convergence, the algorithm yields a reduced-order model with a first-order state-space realization satisfying the first-order interpolatory $\mathcal{H}_2$-optimality conditions [28, 31]. Note that in the case that the high-dimensional system also has first-order structure, `IRKA` and `TF-IRKA` are equivalent and converge to the same reduced-order model [31]. The main steps of `TF-IRKA` are summarized in Algorithm 1. Since the reduced-order

Algorithm 1 Transfer function IRKA (`TF-IRKA`).

Input: Transfer function $\boldsymbol{H}(s)$, initial interpolation points $\{\sigma_j\}_{j=1}^r$, and tangential directions $\{\boldsymbol{b}_j\}_{j=1}^r$ and $\{\boldsymbol{c}_j\}_{j=1}^r$.

Output: Reduced first-order model $\widehat{\Sigma}$: $(\boldsymbol{E}_\mathbb{L}, \boldsymbol{A}_\mathbb{L}, \boldsymbol{B}_\mathbb{L}, \boldsymbol{C}_\mathbb{L})$.

while *no convergence* **do**

Construct $\boldsymbol{E}_\mathbb{L}$, $\boldsymbol{A}_\mathbb{L}$, $\boldsymbol{B}_\mathbb{L}$, and $\boldsymbol{C}_\mathbb{L}$ as in (14), (15), and (16) using the interpolation points $\{\sigma_j\}_{j=1}^r$ and tangential directions $\{\boldsymbol{b}_j\}_{j=1}^r$ and $\{\boldsymbol{c}_j\}_{j=1}^r$.

Compute the generalized eigenvalues and eigenvectors $\left\{\left(\lambda_j, \boldsymbol{x}_j, \boldsymbol{y}_j\right)\right\}_{j=1}^r$ from

$$\boldsymbol{A}_\mathbb{L}\boldsymbol{x}_j = \lambda_j \boldsymbol{E}_\mathbb{L}\boldsymbol{x}_j \quad \text{and} \quad \boldsymbol{y}_j^\mathsf{H}\boldsymbol{A}_\mathbb{L} = \lambda_j \boldsymbol{y}_j^\mathsf{H}\boldsymbol{E}_\mathbb{L}.$$

Update the interpolation points and tangential directions via

$$\sigma_j \leftarrow -\lambda_j, \quad \boldsymbol{b}_j^\mathsf{H} \leftarrow \boldsymbol{y}_j^\mathsf{H}\boldsymbol{B}_\mathbb{L} \quad \text{and} \quad \boldsymbol{c}_j \leftarrow \boldsymbol{C}_\mathbb{L}\boldsymbol{x}_j,$$

for $j = 1, \ldots, r$.

end

model is directly obtained from the underlying Loewner framework, the realness of the original model can be preserved using the technique described in Section 2.2.

3 Structure-preserving near-optimal interpolation

In the following, we consider two iteration schemes similar to `IRKA` for finding near-optimal interpolation points for structure-preserving model reduction in the case of general systems (1) with transfer functions of the form (2). Before we derive our new approach in Section 3.2, we generalize the $\mathcal{H}_2$-norm based method from [32] to the case of multiple-input/multiple-output (MIMO) systems. As it follows similar concepts, we use this method as the main benchmark for the performance of our new approach in the numerical experiments.

3.1 Structure-preserving transfer function IRKA

The problem of constructing heuristically near-optimal interpolants for general structured systems has been considered before in [32]. Therein, the authors present an $\mathcal{H}_2$-norm inspired strategy based on `TF-IRKA` in combination with the structured interpolation framework from Proposition 1 to compute structure-preserving reduced-order models. `SPTF-IRKA`, as sketched in Algorithm 2, can in general be seen as a two-step approach: First, a structured reduced-order model is computed via projection using Proposition 1; then, the transfer function of this structured reduced-order model is approximated by `TF-IRKA`, which yields an $\mathcal{H}_2$-optimal first-order realization, from which the mirror images of the poles of its transfer function are then used to update the interpolation points for the next iteration. The resulting reduced-order

Algorithm 2 Structure-preserving transfer function IRKA (`SPTF-IRKA`).

Input: Dynamical system Σ: $(\mathcal{K}, \mathcal{B}, \mathcal{C})$, initial interpolation points $\{\sigma_j\}_{j=1}^r$, and tangential directions $\{b_j\}_{j=1}^r$ and $\{c_j\}_{j=1}^r$.

Output: Reduced-order system $\widehat{\Sigma}$: $(\widehat{\mathcal{K}}, \widehat{\mathcal{B}}, \widehat{\mathcal{C}})$.

1 **while** *no convergence* **do**

2 Compute the orthogonal basis matrices $\boldsymbol{V}$, $\boldsymbol{W}$ via

$$\boldsymbol{V} \leftarrow \mathtt{orth}\left(\left[\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1)\boldsymbol{b}_1 \ \ldots \ \mathcal{K}(\sigma_r)^{-1}\mathcal{B}(\sigma_r)\boldsymbol{b}_r\right]\right),$$
$$\boldsymbol{W} \leftarrow \mathtt{orth}\left(\left[\mathcal{K}(\sigma_1)^{-\mathsf{H}}\mathcal{C}(\sigma_1)^{\mathsf{H}}\boldsymbol{c}_1 \ \ldots \ \mathcal{K}(\sigma_r)^{-\mathsf{H}}\mathcal{C}(\sigma_r)^{\mathsf{H}}\boldsymbol{c}_r\right]\right).$$

3 Project the system matrices such that

$$\widehat{\mathcal{K}}(s) \leftarrow \boldsymbol{W}^{\mathsf{H}}\mathcal{K}(s)\boldsymbol{V}, \quad \widehat{\mathcal{B}}(s) \leftarrow \boldsymbol{W}^{\mathsf{H}}\mathcal{B}(s), \quad \widehat{\mathcal{C}}(s) \leftarrow \mathcal{C}(s)\boldsymbol{V}.$$

4 Compute an order-r approximation

$$\mathcal{S}_r(s) = \boldsymbol{C}_{\mathbb{L}}\left(s\boldsymbol{E}_{\mathbb{L}} - \boldsymbol{A}_{\mathbb{L}}\right)^{-1}\boldsymbol{B}_{\mathbb{L}},$$

by applying `TF-IRKA` (Algorithm 1) to $\widehat{\boldsymbol{H}}(s) = \widehat{\mathcal{C}}(s)\widehat{\mathcal{K}}(s)^{-1}\widehat{\mathcal{B}}(s)$.

5 Compute the generalized eigenvalues and eigenvectors $\left\{\left(\lambda_j, \boldsymbol{x}_j, \boldsymbol{y}_j\right)\right\}_{j=1}^r$ from

$$\boldsymbol{A}_{\mathbb{L}}\boldsymbol{x}_j = \lambda_j\boldsymbol{E}_{\mathbb{L}}\boldsymbol{x}_j \quad \text{and} \quad \boldsymbol{y}_j^{\mathsf{H}}\boldsymbol{A}_{\mathbb{L}} = \lambda_j\boldsymbol{y}_j^{\mathsf{H}}\boldsymbol{E}_{\mathbb{L}}.$$

6 Update the interpolation points and tangential directions via

$$\sigma_j \leftarrow -\lambda_j, \quad \boldsymbol{b}_j^{\mathsf{H}} \leftarrow \boldsymbol{y}_j^{\mathsf{H}}\boldsymbol{B}_{\mathbb{L}} \quad \text{and} \quad \boldsymbol{c}_j \leftarrow \boldsymbol{C}_{\mathbb{L}}\boldsymbol{x}_j,$$

for $j = 1, \ldots, r$.

7 **end**

model is structure-preserving due to the employed projection framework; the unstructured realization obtained from `TF-IRKA` is only used to update the interpolation points.

Originally, `SPTF-IRKA` has been formulated for single-input/single-output (SISO) systems in [32]. The extension to the MIMO case in Algorithm 2 follows directly from the observation that `TF-IRKA` yields tangential interpolation conditions for MIMO systems. Consequently, basis matrices ensuring tangential interpolation are constructed in Line 2 of Algorithm 2. Similar to the interpolation points, the tangential directions are updated in every step of the iteration in Line 6 of Algorithm 2 by computing additionally to the eigenvalues also the corresponding left and right eigenvectors of the matrix pencil of the $\mathcal{H}_2$-optimal approximation in Line 5. Note that also the tangential version of `TF-IRKA` is used in Line 4 as given in Algorithm 1. Realness of reduced-order models computed with `SPTF-IRKA` can be preserved similarly to the procedure used in the original `IRKA` and stated in [28, Cor. 2.2]: Given a set of interpolation points with tangential directions, which is closed under complex conjugation, the basis matrices $\boldsymbol{V}$, $\boldsymbol{W}$ can be chosen to be real-valued. Using these

matrices in a projection (9) preserves the realness of the original system matrices, while enforcing the tangential interpolation conditions.

In terms of computational effort for the choice of $\mathcal{H}_2$-optimal interpolation points, the two-step approach in Algorithm 2 can be seen as beneficial. Applying `TF-IRKA` (Algorithm 1) directly to the full-order system requires the solution of r linear systems of equations of order n in each step of the iteration. In `SPTF-IRKA` however, `TF-IRKA` is only applied to transfer functions for which linear systems of dimension r have to be solved. In this situation, the outer loop of `SPTF-IRKA` (Algorithm 2) can also be seen as a pre-reduction step that reduces the computational costs of `TF-IRKA`. Similar ideas to reduce the computational costs of iterative model order reduction methods have been used, for example, in [30, 38, 39].

`SPTF-IRKA` uses unstructured $\mathcal{H}_2$-optimality as heuristic for choosing suitable interpolation points in the next iteration step. However, it cannot provide $\mathcal{H}_2$-optimality for the structured case. Another important difference between `TF-IRKA` and `SPTF-IRKA` lies in the requirements of the methods on the availability of the original system. `TF-IRKA` is a true black-box approach, where only access to transfer function evaluations are required. In contrary, `SPTF-IRKA` requires access to the system matrices to construct the basis matrices $\boldsymbol{V}$, $\boldsymbol{W}$ as well as for the projection step.

3.2 Structure-preserving adaptive iterative Krylov algorithm

In addition to accuracy problems already observed in the original publication [32], a flexible application of `SPTF-IRKA` and `TF-IRKA` is limited by the fact that the final reduced order r has to be fixed before the algorithm is started. The choice of a reasonable r is highly problem-dependent and an a priori choice can often only be based on heuristics or in-depth knowledge about the system dynamics. In the cases where a maximum r is not given by implementational restrictions, it needs to be determined by several independent runs of `TF-IRKA` or `SPTF-IRKA` followed by system evaluations to estimate the approximation errors. Another limitation of many `IRKA`-like methods is that the user has no influence on the distribution of the interpolation points. For some applications, surrogates that approximate the high-fidelity model in a specific frequency range only are more interesting than global approximations. While frequency-limited variants of `IRKA` exist [40], these methods rely on the first-order realization of the full- as well as the reduced-order model and are computationally costly for large-scale systems.

Here, we present a new approach for the structure-preserving realization of reduced-order models building on similar concepts as `SPTF-IRKA`, but also addressing the issues raised above. We call this new method the ***Str**ucture-preserving **A**daptive **I**terative **K**rylov **A**lgorithm* (`StrAIKA`). The approach is summarized in Algorithm 3.

3.2.1 Computational procedure

Comparing Algorithms 2 and 3, the main computational procedures look similar. Structure-preserving reduced-order models are computed via Proposition 1, which

Algorithm 3 Structure-preserving adaptive iterative Krylov algorithm (`StrAIKA`).

Input: Dynamical system $\Sigma\colon (\mathcal{K}, \mathcal{B}, \mathcal{C})$, initial interpolation points $\{\sigma_j\}_{j=1}^{r}$, tangential directions $\{\boldsymbol{b}_j\}_{j=1}^{r}$ and $\{\boldsymbol{c}_j\}_{j=1}^{r}$, frequency range Ω, Loewner sampling points $\{\theta_j\}_{j=1}^{2q}$ such that $|\mathrm{Im}\,(\theta_j)| \in \Omega$ for $j = 1, \ldots 2q$, maximum reduced order $r_{\max}$.

Output: Reduced-order system $\widehat{\Sigma}\colon (\widehat{\mathcal{K}}, \widehat{\mathcal{B}}, \widehat{\mathcal{C}})$ of order $r \le r_{\max}$.

while *no convergence* **do**

Compute the orthogonal basis matrices $\boldsymbol{V}$, $\boldsymbol{W}$ via

$$\boldsymbol{V} \leftarrow \mathtt{orth}\left(\left[\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1)\boldsymbol{b}_1 \ \ldots \ \mathcal{K}(\sigma_r)^{-1}\mathcal{B}(\sigma_r)\boldsymbol{b}_r\right]\right),$$
$$\boldsymbol{W} \leftarrow \mathtt{orth}\left(\left[\mathcal{K}(\sigma_1)^{-\mathsf{H}}\mathcal{C}(\sigma_1)^{\mathsf{H}}\boldsymbol{c}_1 \ \ldots \ \mathcal{K}(\sigma_r)^{-\mathsf{H}}\mathcal{C}(\sigma_r)^{\mathsf{H}}\boldsymbol{c}_r\right]\right).$$

Project the system matrices such that

$$\widehat{\mathcal{K}}(s) \leftarrow \boldsymbol{W}^{\mathsf{H}}\mathcal{K}(s)\boldsymbol{V}, \quad \widehat{\mathcal{B}}(s) \leftarrow \boldsymbol{W}^{\mathsf{H}}\mathcal{B}(s), \quad \widehat{\mathcal{C}}(s) \leftarrow \mathcal{C}(s)\boldsymbol{V}.$$

Compute the Loewner interpolant (13), $\Sigma_{\mathbb{L}}\colon (\boldsymbol{E}_{\mathbb{L}}, \boldsymbol{A}_{\mathbb{L}}, \boldsymbol{B}_{\mathbb{L}}, \boldsymbol{C}_{\mathbb{L}})$, via (10), (11) and (12) with samples of $\widehat{\boldsymbol{H}}(s)$ in the points $\{\theta_j\}_{j=1}^{2q}$.

Compute the generalized eigenvalues and eigenvectors $\left\{\left(\lambda_j, \boldsymbol{x}_j, \boldsymbol{y}_j\right)\right\}_{j=1}^{k}$ from

$$\boldsymbol{A}_{\mathbb{L}}\boldsymbol{x}_j = \lambda_j \boldsymbol{E}_{\mathbb{L}}\boldsymbol{x}_j \quad \text{and} \quad \boldsymbol{y}_j^{\mathsf{H}}\boldsymbol{A}_{\mathbb{L}} = \lambda_j \boldsymbol{y}_j^{\mathsf{H}}\boldsymbol{E}_{\mathbb{L}}.$$

Choose eigentriples according to frequency region of interest

$$\boldsymbol{\Lambda}_{\Theta} \leftarrow \left\{\left(\lambda_j, \boldsymbol{x}_j, \boldsymbol{y}_j\right) \,\middle|\, |\mathrm{Im}\,(\lambda_j)| \in \Omega, \ \text{for } j = 1, \ldots, k\right\}.$$

if $|\boldsymbol{\Lambda}_{\Theta}| > r_{\max}$ **then**

Compute dominance (17) for all poles in $\boldsymbol{\Lambda}_{\Theta}$ with respect to $\Sigma_{\mathbb{L}}$.

Keep only the $r = r_{\max}$ most dominant poles in $\boldsymbol{\Lambda}_{\Theta}$.

else

Set $r \leftarrow |\boldsymbol{\Lambda}_{\Theta}|$.

end

Update the interpolation points and tangential directions via

$$\sigma_j \leftarrow -\lambda_j, \quad \boldsymbol{b}_j^{\mathsf{H}} \leftarrow \boldsymbol{y}_j^{\mathsf{H}}\boldsymbol{B}_{\mathbb{L}} \quad \text{and} \quad \boldsymbol{c}_j \leftarrow \boldsymbol{C}_{\mathbb{L}}\boldsymbol{x}_j,$$

such that $(\lambda_j, \boldsymbol{x}_j, \boldsymbol{y}_j) \in \boldsymbol{\Lambda}_{\Theta}$, for $j = 1, \ldots, r$.

end

are then used to construct Loewner surrogates that are used to update the interpolation points and tangential directions for the next iteration step. The main difference between `SPTF-IRKA` and `StrAIKA` lies in the construction of the Loewner interpolants during the iteration. While `SPTF-IRKA` employs a complete run of `TF-IRKA` to construct an order-r unstructured approximation of the structured reduced-order model $\widehat{\Sigma}$, in

`StrAIKA`, the transfer function $\widehat{\boldsymbol{H}}$ is sampled in the frequency range of interest Ω to reveal all essential system dynamics. Similar to the methods discussed in [30, 38], the intermediate models $\widehat{\Sigma}$ and $\Sigma_{\mathbb{L}}$ are used to leverage the computational costs of different tasks.

In Line 4 of Algorithm 3, we use the variant of the Loewner framework described in Section 2.2 that only relies on the evaluation of the low-dimensional transfer function $\widehat{\boldsymbol{H}}$ rather than its derivatives as needed in `TF-IRKA`. However, this can be arbitrarily replaced by other approaches for the identification of unstructured first-order systems (3) from frequency domain data. This includes other variants of the Loewner framework such as the one described in Section 2.3, its block version [33], and variations in these for choosing the dominant dynamics [41], but also completely different methods can be employed such as vector fitting [42, 43], RKFIT [44] or the AAA algorithm [45]. The additional computational cost of evaluating $\widehat{\boldsymbol{H}}$, which is of order $r \leq r_{\max}$, is negligible compared to updating the basis matrices $\boldsymbol{V}, \boldsymbol{W}$, which requires decompositions of large-scale matrices of dimension n, if cases with $r_{\max} \ll n$ are considered. The advantages of considering Line 4 detached from the desired reduced order are that, first, concepts such as oversampling and localized sampling can be used to influence the accuracy of the approximation $\Sigma_{\mathbb{L}}$ in the frequency range Ω of interest, and second, that the amount of poles in the frequency range of interest Ω is a strong indicator for the reduced order needed to well approximate the original transfer function in this region.

In the case of irrational transfer functions, the Loewner framework in Line 4 of Algorithm 3 will recover parts of the potentially infinite number of poles of the full-order transfer function. Depending on the size of the Loewner approximation, the frequency range of interest may contain significantly more poles than effectively needed for the reduced order of the structured approximation. In the following section, we propose to use a dominance measure, in addition to the frequency range of interest, to assess the relevance of the poles retrieved from the Loewner approximation. This allows to select only a subset of the potential interpolation points for the next computational step. Additionally, one needs to note that if multiple interpolation points are selected that do not contribute to the accuracy of the reduced-order model due to the system structure, the orthogonalization step in Line 2 of Algorithm 3 will result in basis matrices with less columns than selected poles, which leads to an adjustment of the reduced order accordingly.

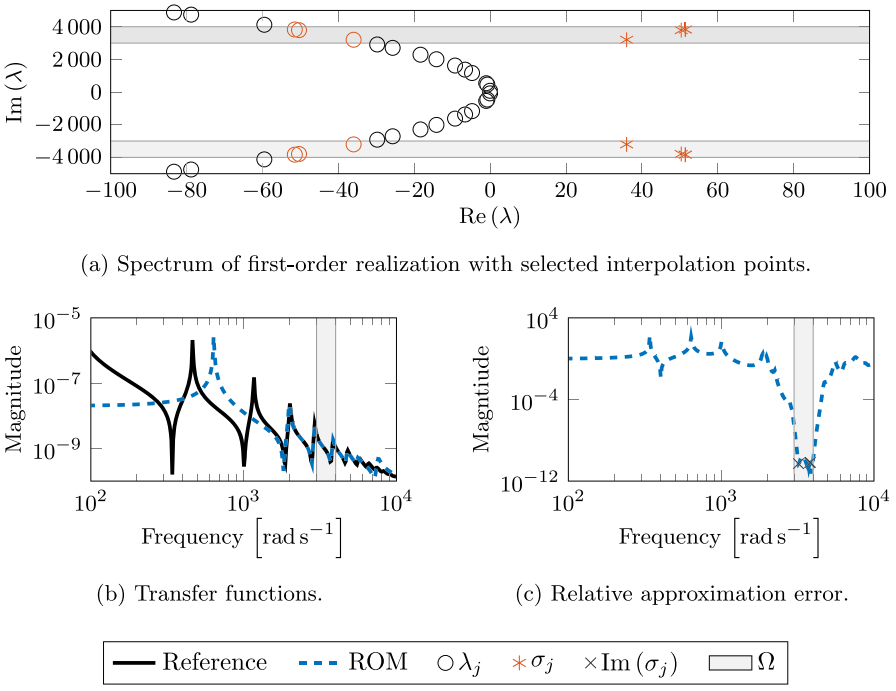
Realness of the original system matrices can be preserved in the reduced-order moel throughout the iteration using similar ideas as for the previously discussed methods. Under the assumption that the initial interpolation points and tangential directions are closed under complex conjugation, and the original model has a reflective transfer function, real matrices $\boldsymbol{V}, \boldsymbol{W}$ can be computed in Line 2 of Algorithm 3 by splitting basis contributions corresponding to complex conjugate interpolation points and by concatenating

$$\boldsymbol{V}_{\mathbb{R}} = \begin{bmatrix} \operatorname{Re}(\boldsymbol{V}) & \operatorname{Im}(\boldsymbol{V}) \end{bmatrix} \quad \text{and} \quad \boldsymbol{W}_{\mathbb{R}} = \begin{bmatrix} \operatorname{Re}(\boldsymbol{W}) & \operatorname{Im}(\boldsymbol{W}) \end{bmatrix}.$$

Thereby, the matrices of the reduced-order model computed in Line 3 are also real. Using the realification of the Loewner framework as described at the end of Section 2.2 leads to sets of eigenvalues and eigenvectors in Line 5 closed under conjugation.

3.2.2 Interpolation point selection

Reduced-order models computed with the structure-preserving framework presented in Section 2.1 approximate the full-order model well in the vicinity of chosen interpolation points. This observation can be used to compute reduced-order models, which approximate the original model in a specific frequency region only. Additionally, `IRKA`-like methods aim for interpolation at the mirror images of transfer function poles with respect to the imaginary axis. Figure 1 illustrates the combination of these two ideas. The part of the spectrum of the matrix pencil of the model close to the imaginary axis is shown in Fig. 1a and all eigenvalues λ_j with $3000 < |\mathrm{Im}\,(\lambda_j)| < 4000$ are marked, which corresponds to the frequency region of interest $\Omega = [3000, 4000]\,\mathrm{rad\,s^{-1}}$. The mirror images of these eigenvalues are considered interpolation points in `StrAIKA` for the structure-preserving interpolation framework. The transfer function of the interpolating structured reduced-order model is shown in Fig. 1b with the pointwise relative approximation error in Fig. 1c. It can be seen that the reduced-order model is an accurate approximation of the original system



(a) Spectrum of first-order realization with selected interpolation points.

(b) Transfer functions.

(c) Relative approximation error.

Fig. 1 Approximation of a model in a specified frequency region Ω. Only mirror images with respect to the imaginary axis of poles in the specified frequency region are considered leading to an accurate local approximation of the transfer function

in the vicinity of the interpolation points σ_j. Choosing only interpolation points in the frequency region Ω, which is important for the application of the reduced-order model, can therefore be a strategy to decrease the required size of the reduced-order model.

Note that there are no limitations for the choice of Ω. For the global approach, it can be chosen to be the complete positive real axis $\mathbb{R}_{\geq 0}$, but in other applications, only subintervals might be of interest such that

$$\Omega = \bigcup_{j=1}^{k} [\omega_{1,j}, \omega_{2,j}],$$

where $\omega_{1,j}, \omega_{2,j} \in \mathbb{R}_{\geq 0}$ and $\omega_{1,j} \leq \omega_{2,j}$. In Line 6 of Algorithm 3, the absolute value of the imaginary part of the eigenvalues is considered, which implies a certain symmetry in the importance of positive and negative frequency regions; see also Fig. 1a. For certain applications, it may be advantageous to select eigenvalues using other criteria, for example, their distance to the imaginary axis.

In principle, the reduced-order model might grow too large if all interpolation points inside a defined region are considered. Especially in cases where the global dynamics are approximated, the order r can grow fast and even approach n. For such cases, `StrAIKA` chooses eigenvalues up to a defined maximum $r_{\max}$ as locations for interpolation points. In this case, the $r_{\max}$ interpolation points, which will result in a suitably good approximation of the original model, have to be selected from all potential interpolation points. To this end, the *dominance* of all poles given by the eigenvalues in Ω is computed and, for the $r_{\max}$ most dominant poles, the interpolation points for the next iteration are set as their mirror images. The systems constructed in Line 4 of Algorithm 3 are in first-order unstructured form (3). For such systems, the dominance of a pole λ_j with corresponding right and left eigenvectors $\boldsymbol{x}_j$ and $\boldsymbol{y}_j$ is defined as

$$d_j = \frac{\left\| (\boldsymbol{C}_{\mathbb{L}} \boldsymbol{x}_j)(\boldsymbol{y}_j^{\mathsf{H}} \boldsymbol{B}_{\mathbb{L}}) \right\|_2}{|\mathrm{Re}\,(\lambda_j)|}; \tag{17}$$

where $\boldsymbol{C}_{\mathbb{L}}$ and $\boldsymbol{B}_{\mathbb{L}}$ are the output and input matrices constructed in Line 4 of Algorithm 3, respectively, and the eigenvectors are normalized such that $\boldsymbol{y}_j^{\mathsf{H}} \boldsymbol{E}_{\mathbb{L}} \boldsymbol{x}_j = 1$. A pole λ_j is called dominant, if $d_j > d_k$ for all $j \neq k$; see [46].

To ensure a high approximation quality in the frequency range of interest Ω, it is often beneficial to include also the first potential interpolation points located outside both ends of Ω. While having only a small impact on the size of the reduced-order model, this can greatly increase the accuracy of the reduced-order model, especially, if the full-order model's transfer function has poles near the boundaries of Ω.

Note that similar to `SPTF-IRKA`, the `StrAIKA` method relies on the $\mathcal{H}_2$-optimality of the classical `IRKA` as its main heuristic for choosing and updating interpolation points. This is supplemented by the dominance measure in (17). As for `SPTF-IRKA`, optimality cannot be guaranteed for structured systems but occurs for unstructured ones as `StrAIKA` resembles the classical `IRKA` method in this special case.

Remark 1 In some cases, the constructed Loewner approximation may not have any transfer function poles in the frequency range of interest that could be used for the selection as new interpolation points. This can occur if in the frequency range of interest, the full-order transfer function does not contain any peaks or sinks such that its behavior is solely determined by poles lying outside this range. As mentioned previously, a potential strategy in this situation is to choose the two poles that lie closest to the two boundaries of the frequency range of interest as these are the predominant factors of the observed transfer function behavior. A different strategy is making use of the fact that the lack of poles in the frequency range of interest indicates a very smooth behavior of the transfer function in this region such that it is enough to choose the two end points of the frequency range on the imaginary axis as interpolation points to provide an accurate approximation.

For the discussion above, we have implicitly assumed that the matrix pencil of the Loewner approximation has finite, distinct eigenvalues that do not lie on the imaginary axis as this is a typical situation in model reduction for dynamical systems. First, note that in general, the descriptor matrix $\boldsymbol{E}_{\mathbb{L}}$ in the Loewner approximation does not need to be invertible as the proposed method only selects finite poles as potential interpolation points. In the case of a singular $\boldsymbol{E}_{\mathbb{L}}$, the approximation contains algebraic constraints, which indicates that such constraints have also been present in the original full-order system. While in most cases, interpolation at finite points is sufficient since the algebraic constraints can be eliminated, this might not be desired by the user. We refer the reader to [47] for a detailed discussion about interpolation-based model reduction for systems with differential-algebraic equations.

In the case of identical poles, there might exist an underlying higher-order Jordan block such that the matrix pencil is not diagonalizable. In this case, it is necessary in the classical `IRKA` method to include additional derivative information at the mirror images of these poles. This can also be included in the `StrAIKA` method by computing the derivatives of the three transfer function factors $\mathcal{C}$, $\mathcal{K}$, and $\mathcal{B}$. However, this case may numerically only occur in the Loewner framework when the full-order model had a rational transfer function with such higher-order Jordan blocks and is exactly recovered in the Loewner approximation. Similar to the original `IRKA` method, the only assumption about the placement of the transfer function poles of the Loewner approximation is that they should not lie on the imaginary axis such that the interpolation points that are chosen as mirror images of the poles are different from the poles. As in `IRKA`, we do not differentiate between stable and anti-stable poles as it has been observed that these will occur and persist naturally for unstable models [48].

4 Numerical experiments

We now demonstrate the performance of `StrAIKA` in comparison with the established `IRKA`-like methods `TF-IRKA` and `SPTF-IRKA`. Where applicable, the classical `IRKA` is also included in the comparison. The numerical experiments have been performed on a laptop equipped with an AMD Ryzen™ 7 PRO 5850U and 12 GB RAM running on Linux Mint 21 as an operating system. All algorithms have been

implemented and run with MATLAB® version 9.11.0.1837725 (R2021b Update 2). The results for `IRKA` have been computed with M-M.E.S.S. version 2.2 [49]; for `TF-IRKA` and `SPTF-IRKA`, we used the implementation from [3] in combination with an implementation of the Loewner method by the authors of [37]. The source code, data, and computed results of the numerical experiments are available at [50]. Note that the numerical results may vary depending on the hardware and software specifications of the computational environment.

In all experiments, we use a maximum number of iterations: $n_{\text{iter,max}} = 50$ for `StrAIKA` and the outer iterations of `SPTF-IRKA`, and $n_{\text{iter,max}} = 100$ for `TF-IRKA` and the inner iterations of `SPTF-IRKA`. The algorithms terminate, if the relative difference between the interpolation points in two consecutive iterations falls under the threshold of $\epsilon = 1 \cdot 10^{-3}$. To compare the accuracy of the methods, we plot the pointwise relative approximation errors, given by

$$\text{relerr}(\omega) := \frac{\|\boldsymbol{H}(\mathrm{i}\omega) - \widehat{\boldsymbol{H}}(\mathrm{i}\omega)\|_2}{\|\boldsymbol{H}(\mathrm{i}\omega)\|_2}$$

in specified frequency intervals of interest $\omega \in [\omega_{\min}, \omega_{\max}]$. We also approximate the local, relative errors in the closed frequency range Ω under the $\mathcal{L}_\infty$-norm via

$$\text{relerr}_{\mathcal{L}_\infty,\Omega} = \frac{\max\limits_{\omega\in\Omega} \|\boldsymbol{H}(\mathrm{i}\omega) - \widehat{\boldsymbol{H}}(\mathrm{i}\omega)\|_2}{\max\limits_{\omega\in\Omega} \|\boldsymbol{H}(\mathrm{i}\omega)\|_2} \approx \frac{\|\boldsymbol{H} - \widehat{\boldsymbol{H}}\|_{\mathcal{L}_\infty,\Omega}}{\|\boldsymbol{H}\|_{\mathcal{L}_\infty,\Omega}},$$

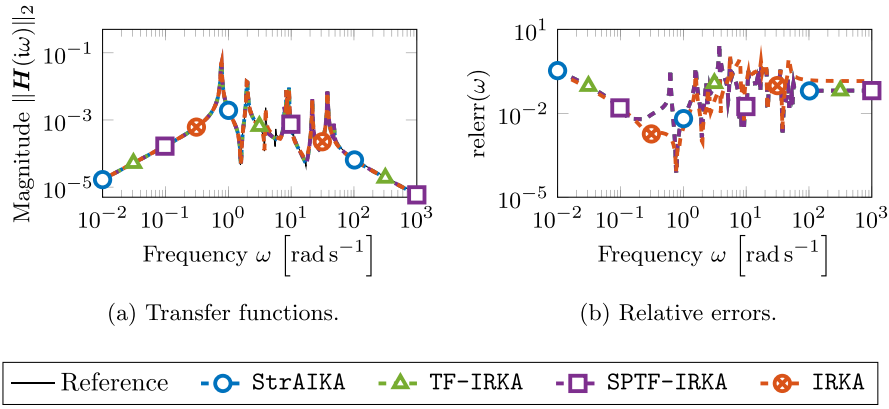
using equidistant discretizations of Ω. In addition to the accuracy of the reduced-order models, we also compare the number of solutions of full-order n linear systems performed by each method and the overall resulting computation time. Note that these values are influenced by the chosen implementation and hardware and do not necessarily reflect properties of the method itself.

4.1 Unstructured first-order system example

In the first example, we consider a system modeling the structural response of the Russian Service Module of the International Space Station (ISS) [51]. The system is given via first-order differential equations (3) and has $n = 270$ states, $m = 3$ inputs, and $p = 3$ outputs. The transfer function is given by

$$\boldsymbol{H}(s) = \boldsymbol{C}\,(s\boldsymbol{E} - \boldsymbol{A})^{-1}\,\boldsymbol{B}.$$

The model is evaluated for frequencies in the range $\Omega = \left[1 \cdot 10^{-2}, 1 \cdot 10^{3}\right] \text{rad}\,\text{s}^{-1}$. Because of the first-order structure of the full-order model, `IRKA` can be applied in this case. Additionally, `TF-IRKA`, `SPTF-IRKA`, and `StrAIKA` are employed to compute real-valued reduced-order models of size $r = 20$ each. Sigma plots of the transfer functions and relative approximation errors for all models are given in Fig. 2. Table 1 summarizes the performance of the algorithms.



(a) Transfer functions. (b) Relative errors.

Fig. 2 First-order system example: All applied methods provide a similar approximation behavior since no special structure needs to be preserved in the example. Insignificant differences are revealed by the pointwise relative errors

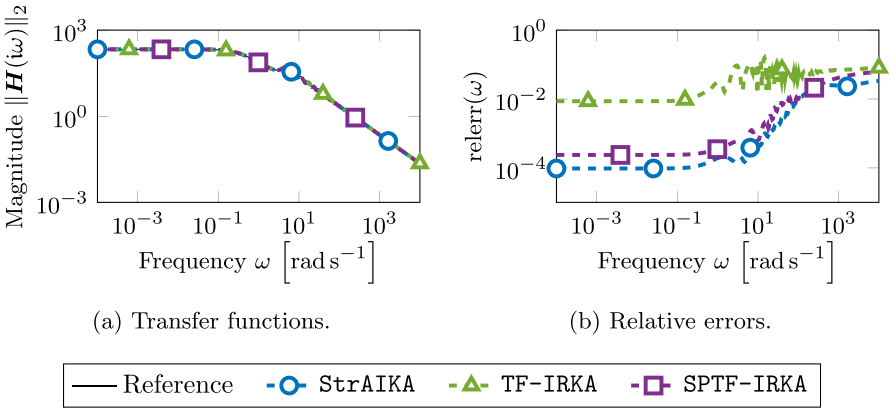
All employed methods succeed in computing surrogates, which approximate the original system up to the same degree of accuracy; `StrAIKA`, `TF-IRKA`, and `SPTF-IRKA` even converge to the same interpolation points. Only `IRKA` obtains a different local optimum, which results in an insignificantly larger relative approximation error (cf. Table 1). This meets expectations, as the first-order structure of the original system can be represented well by the realization `TF-IRKA` yields. `TF-IRKA` converges after only ten iterations, while the other methods require more. `IRKA` performs the fewest decompositions of the full-order matrices and has the shortest runtime; however, the significance of the runtime is limited for this small example. `StrAIKA` requires the most iterations, however not the most matrix decompositions. The noticeably longer runtime compared to the other methods is related to computing the intermediate Loewner interpolant in each iteration. In most cases, the computational cost of this step can be neglected. In this example, however, n is relatively small compared to q, so the Loewner step has a measurable impact on the runtime of `StrAIKA`.

4.2 Time-delayed heated rod

Here, we consider a model of a heated rod with distributed control and homogeneous Dirichlet boundary conditions, which is cooled by delayed feedback. This system

Table 1 First-order system example: comparison of the relative, local $\mathcal{L}_\infty$-error, the required number of iterations n_{iter}, the number of solutions of full-order n linear systems n_{ls}, and the computation time t_{c}

Algorithm	$\text{relerr}_{\mathcal{L}_\infty,\Omega}$	n_{iter}	n_{ls}	t_{c} (s)
`StrAIKA`	$3.67 \cdot 10^{-2}$	20	212	1.94
`TF-IRKA`	$3.67 \cdot 10^{-2}$	10	660	0.26
`SPTF-IRKA`	$3.67 \cdot 10^{-2}$	12	132	0.34
`IRKA`	$4.22 \cdot 10^{-2}$	13	130	0.22



(a) Transfer functions. (b) Relative errors.

Fig. 3 Time-delay system example: The unstructured approximation computed by `TF-IRKA` is around two orders of magnitude less accurate than the structure-preserving reduced-order models computed by `StrAIKA` and `SPTF-IRKA` for smaller to medium frequencies. Overall, `StrAIKA` provides the most accurate approximation

has also been analyzed in [52]. A discretization of the underlying partial differential equation leads to the transfer function

$$\boldsymbol{H}(s) = \boldsymbol{C}\left(s\boldsymbol{E} - \boldsymbol{A}_0 - \mathrm{e}^{-\tau s}\,\boldsymbol{A}_\mathrm{d}\right)^{-1}\boldsymbol{B},$$

with $n = 1\,000\,000$ states, $m = 5$ inputs, and $p = 4$ outputs. The delay is $\tau = 1$ in this example. The input matrix $\boldsymbol{B}$ has a block structure such that the rod is heated uniformly at different sections by the inputs. The outputs are the average temperatures on these sections. For this example, the frequency range $\Omega = \left[1 \cdot 10^{-4}, 1 \cdot 10^{4}\right]\mathrm{rad\,s^{-1}}$ is considered. For the experiments, we fix the reduced order to $r = 10$ and compute reduced-order models with `StrAIKA`, `TF-IRKA`, and `SPTF-IRKA`. The system cannot be transformed into an equivalent system with first-order structure; therefore, `IRKA` cannot be applied in this case. The initial interpolation points are distributed logarithmically equidistant in $\mathrm{i}\left[1 \cdot 10^{-4}, 1 \cdot 10^{2}\right]$ and its complex conjugate. Figure 3 plots the maximum singular values of the transfer functions and the error systems. Further results are given in Table 2.

In this example, `StrAIKA` and `SPTF-IRKA` compute reduced-order models with comparable accuracy. However, `StrAIKA` provides the smallest worst-case error of all methods as shown in Table 2. As expected, the first-order realization computed by `TF-IRKA` cannot capture the dynamics of the delay system well and, therefore,

Table 2 Time-delay system example: comparison of the relative, local $\mathcal{L}_\infty$-error, the required number of iterations n_iter, the number of solutions of full-order n linear systems n_ls, and the computation time t_c

Algorithm	$\mathrm{relerr}_{\mathcal{L}_\infty,\Omega}$	n_iter	n_ls	t_c (s)
`StrAIKA`	$1.01 \cdot 10^{-4}$	37	190	245.07
`TF-IRKA`	$2.06 \cdot 10^{-2}$	73	2220	365.24
`SPTF-IRKA`	$2.38 \cdot 10^{-4}$	3	29	35.96

provides an approximation that is around two orders of magnitude less accurate than the models computed by StrAIKA and SPTF-IRKA. Because of its rapid convergence, the runtime of SPTF-IRKA is considerably lower than for the other two algorithms. But also the runtime of StrAIKA is significantly shorter than that of TF-IRKA. Note that most of the system dynamics happen in the considered frequency range. Therefore, the intermediate Loewner realizations inside the StrAIKA iterations reach early an order of $r = 150$ in this example. StrAIKA chooses the 10 most dominant poles from all available poles in the frequency region of interest to keep the final reduced-order model at the desired size. However, the computational effort of StrAIKA is dominated by solving the linear systems for the construction of the projection matrices. In this example, the subsequent choice of dominant poles hinders a fast convergence that was observed when applying SPTF-IRKA.

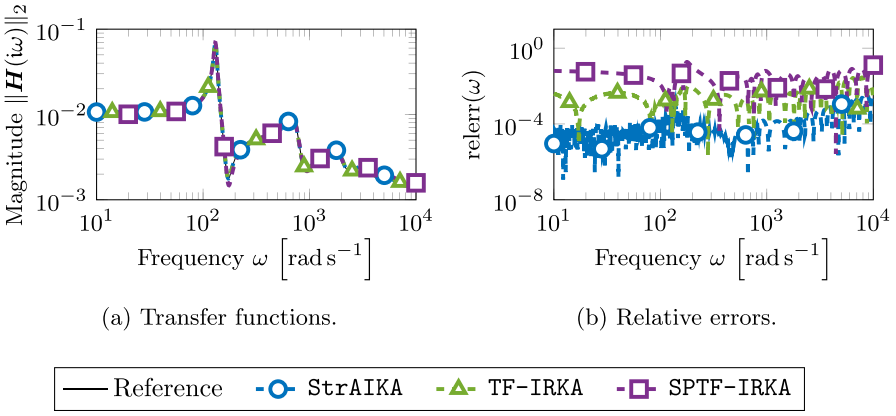
4.3 Viscoelastic beam

This example models a flexible beam with viscoelastic core. The beam of length $l = 0.21\,\mathrm{m}$ has a symmetric sandwich structure consisting of two layers of cold rolled steel surrounding a viscoelastic ethylene-propylene-diene core [7]; the beam is clamped at one side. After discretization, the transfer function of the system is given by

$$\boldsymbol{H}(s) = \boldsymbol{C}\left(s^2\boldsymbol{M} + \boldsymbol{K} + \frac{G_0 + G_\infty (s\tau)^\alpha}{1 + (s\tau)^\alpha}\boldsymbol{G}\right)^{-1}\boldsymbol{B}.$$

The model has $n = 3\,360$ states, $m = 1$ input, and $p = 1$ output. The beam is excited by a single load at its free end, and the displacement is measured at the same location, resulting in output and input mappings $\boldsymbol{C} = 100 \cdot \boldsymbol{B}^\mathsf{T} = \begin{bmatrix}0 \cdots 0\ 1\end{bmatrix}$. The matrices $\boldsymbol{M}$, $\boldsymbol{K}$, and $\boldsymbol{G}$ are available from [53], and the parameters are chosen as in [53] to be $G_0 = 350.4$, $G_\infty = 3.062$, $\tau = 8.230$, and $\alpha = 0.675$. In this example, we limit the frequency range of interest to $\Omega = \left[10, 1 \cdot 10^4\right]\,\mathrm{rad\,s^{-1}}$. Note that the system's transfer function has poles, which lie outside of this range. No maximum reduced order r is set in this case, so StrAIKA determines it in an adaptive way. The initial interpolation points are a single complex conjugate pair, where the absolute value of its imaginary part is located near the mean of the boundaries of the frequency range of interest Ω. The automatically determined order is used for the experiments with TF-IRKA and SPTF-IRKA, where the $\lceil r/2 \rceil$ initial expansion points and their complex conjugates are distributed logarithmically equidistant in Ω. The sigma plots for the frequency responses of the reference and the reduced-order models as well as the corresponding errors are given in Fig. 4. The performance of the methods is shown in Table 3.

StrAIKA converges after ten iterations to a model of size $r = 15$, i.e., the reduced-order models computed by TF-IRKA and SPTF-IRKA have order $r = 16$. All three algorithms produce reasonably accurate models regarding the reference, while the model computed by StrAIKA is around three orders of magnitude more accurate for most frequencies (cf. Figure 4b). Also, the worst-case approximation error of StrAIKA is around one order of magnitude better than SPTF-IRKA; see Table 3. This can be explained by the fact that StrAIKA places all interpolation points in Ω, while some of the interpolation points obtained by the other two algorithms are



(a) Transfer functions. (b) Relative errors.

Fig. 4 Viscoelastic beam example: All methods succeed in computing reasonably accurate reduced-order models. The relative approximation error of the reduced-order model obtained by `StrAIKA` is around three orders of magnitude smaller compared to the other methods for most frequencies

located in the frequency region above $1 \cdot 10^4$ rad s^{-1}, leading to a higher error in Ω. `SPTF-IRKA` did not converge after 50 iterations, resulting in a high computation time. In terms of computational effort, `StrAIKA` is clearly advantageous in this example.

4.4 Radio frequency gun

As the last example, we consider a radio frequency gun as described in [54]. Discretizing the system leads to the transfer function

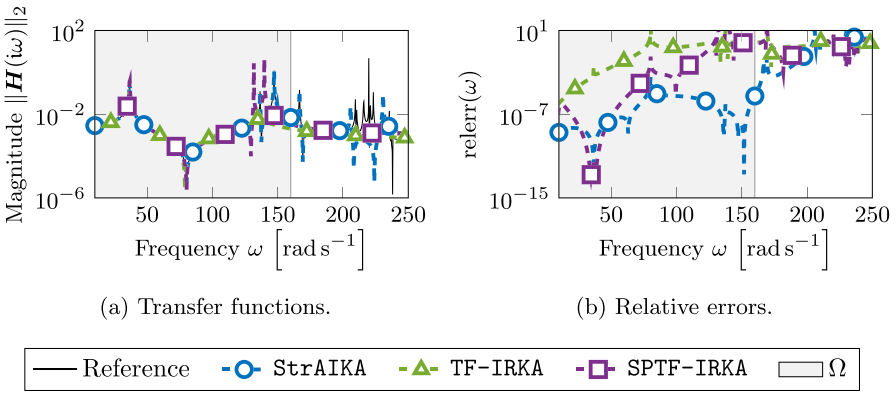
$$\boldsymbol{H}(s) = \boldsymbol{C}\left(s^2\boldsymbol{M} + \mathrm{i}\left(s^2 - \sigma_1^2\right)^{\frac{1}{2}}\boldsymbol{W}_1 + \mathrm{i}\left(s^2 - \sigma_2^2\right)^{\frac{1}{2}}\boldsymbol{W}_2 + \boldsymbol{K}\right)^{-1}\boldsymbol{B},$$

where $\sigma_1 = 0$ and $\sigma_2 = 108.8774$. The full-order model has $n = 9\,956$ states and the matrices $\boldsymbol{M}$, $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, and $\boldsymbol{K}$ are taken from [53]. We consider $m = 1$ input and $p = 1$ output. The input vector is populated with values drawn from the standard normal distribution and the system output is measured at the first degree of freedom, corresponding to $\boldsymbol{C} = \begin{bmatrix}1\ 0\ \ldots\ 0\end{bmatrix}$. These vectors are also part of the code package supplementing this article [50]. One structural feature of this model is that its system matrices are real while its transfer function is not reflective, i.e., $\overline{H(s)} = H(\overline{s})$ does

Table 3 Viscoelastic beam example: comparison of the relative, local $\mathcal{L}_\infty$-error, the required number of iterations n_{iter}, the number of solutions of full-order n linear systems n_{ls}, and the computation time t_{c}

Algorithm	relerr$_{\mathcal{L}_\infty,\Omega}$	n_{iter}	n_{ls}	t_{c} (s)	
`StrAIKA`	$9.14 \cdot 10^{-4}$	10	74	0.71	
`TF-IRKA`	$1.04 \cdot 10^{-3}$	18	912	0.94	
`SPTF-IRKA`	$9.60 \cdot 10^{-3}$	50	457	6.83	$*$

The $*$ marks experiments, where the maximum number of iterations has been reached without convergence



(a) Transfer functions. (b) Relative errors.

Fig. 5 Radio frequency gun example: The reduced-order model computed by `StrAIKA` shows a relatively uniform accuracy in Ω, while the surrogate obtained form `SPTF-IRKA` is only accurate for less than 120 rad s^{-1}. The first-order system computed by `TF-IRKA` is not able to approximate the dynamics of the original system at all

not hold for all $s \in \mathbb{C}$. Since our interest in approximation accuracy lies here only in the positive imaginary axis, we use the realification approach from Section 3.2.1 to guarantee the projection bases to be real. The reduced-order models are computed to approximate the original system in $\Omega = [1, 160]\,\mathrm{rad\,s^{-1}}$ and the necessary order r for the surrogate is automatically determined by `StrAIKA`. One initial complex conjugate pair of interpolation points is placed in the middle of Ω. The initial $\lceil r/2 \rceil$ pairs of interpolation points for `TF-IRKA` and `SPTF-IRKA` are distributed linearly equidistant in Ω. The sigma plots for the frequency responses of the reference and the reduced-order models as well as the corresponding relative approximation errors are shown in Fig. 5. The local, relative $\mathcal{L}_\infty$-errors and computational costs are presented in Table 4.

`StrAIKA` automatically determines a reduced order of $r = 15$ and computes a reduced-order model, which is more accurate in terms of the local maximum error compared to the reduced-order models computed by the other two algorithms. Placing the interpolation points only inside Ω leads to a higher accuracy in this region, while the approximation deteriorates for higher frequencies. `StrAIKA` converges after 26 iterations and the worst-case approximation error is several orders of magnitude smaller than for the other methods; see Table 4. `SPTF-IRKA` reaches convergence after only 4 iterations but the resulting reduced-order model fails to approximate the full-order

Table 4 Radio frequency gun example: comparison of the relative, local $\mathcal{L}_\infty$-error, the required number of iterations n_{iter}, the number of solutions of full-order n linear systems n_{ls}, and the computation time t_{c}

Algorithm	relerr$_{\mathcal{L}_\infty,\Omega}$	n_{iter}	n_{ls}	t_{c} (s)	
`StrAIKA`	$2.71 \cdot 10^{-8}$	26	216	63.35	
`TF-IRKA`	1.00	100	4848	1481.25	$*$
`SPTF-IRKA`	1.00	4	54	14.49	

The $*$ marks experiments, where the maximum number of iterations has been reached without convergence

model for frequencies larger than $120\,\mathrm{rad\,s^{-1}}$. The unstructured method TF-IRKA does not converge in the maximum number of iterations and fails to compute an accurate reduced-order model. The comparison to TF-IRKA again shows the importance of structure-preserving model order reduction strategies. Note that although StrAIKA starts with a single complex conjugate pair of interpolation points, the resulting reduced-order model is able to capture all important system dynamics in the required frequency range by determining a reasonable size for the reduced-order model.

5 Conclusions

In this paper, we proposed StrAIKA, a new algorithm that computes structure-preserving reduced-order models of systems with arbitrary transfer function structure in an iterative way. Similar to other IRKA-like methods, StrAIKA uses mirror images of the poles of intermediate reduced-order model transfer functions as interpolation points for the next iteration. In each iteration, the Loewner framework is used to compute first-order realizations of transfer function data collected from the current reduced-order model, whose matrix pencil eigenvalues are the basis for the interpolation points in the next iteration. StrAIKA automatically determines a size for the reduced-order model by considering the eigenvalues located in a given frequency range of interest. This ensures a reasonable approximation of the system dynamics at least in these regions. StrAIKA is completely agnostic to the actual transfer function structure of the original model and does not require any transfer function derivatives.

We demonstrated the versatility and effectiveness of StrAIKA in four numerical examples with different internal structures. The benchmark systems model structural vibration, heat transfer with internal delay, viscoelasticity, and radio wave propagation. While StrAIKA cannot guarantee optimality similar to other projection-based approaches for structured systems, it showed comparable or even significantly better accuracy with respect to established IRKA-like methods. In the case of a structured transfer function and if only a limited frequency range is of interest for the application, StrAIKA easily outperforms methods that optimize the approximation error under the $\mathcal{H}_2$-norm in terms of accuracy and often also in terms of the required computational effort.

An open question that has not been covered in this paper is the preservation of additional system properties like stability. However, the preservation of such system properties has only been solved for particular system structures such as first-order systems or second-order systems with a mechanical matrix structure. For the case of generally structured systems as considered in this paper, no solution to this problem for projection-based model reduction is known yet. Another idea for future investigations is the reduction of the computational costs of StrAIKA by considering an additional layer of approximation as it has been done for IRKA-like methods in [30, 38].

Acknowledgements Parts of this work were carried out while Werner was with the Courant Institute of Mathematical Sciences, New York University, USA.

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Beattie, C.A., Gugercin, S.: Interpolatory projection methods for structure-preserving model reduction. Syst. Control Lett. **58**(3), 225–232 (2009). https://doi.org/10.1016/j.sysconle.2008.10.016
2. Abraham, R., Marsden, J.E.: Foundations of mechanics, 2nd edn. Addison-Wesley Publishing Company, Inc., Redwood City (1987). https://resolver.caltech.edu/CaltechBOOK:1987.001
3. Werner, S.W.R.: Structure-preserving model reduction for mechanical systems. Dissertation, Otto-von-Guericke-Universität, Magdeburg, Germany (2021). https://doi.org/10.25673/38617
4. Wu, K.: Power converters with digital filter feedback control. Academic Press, London (2016). https://doi.org/10.1016/C2015-0-01103-0
5. Gao, Q., Karimi, H.R.: Stability, control and application of time-delay systems. Butterworth-Heinemann, Oxford (2019). https://doi.org/10.1016/C2017-0-02175-4
6. Aumann, Q., Deckers, E., Jonckheere, S., Desmet, W., Müller, G.: Automatic model order reduction for systems with frequency-dependent material properties. Comput. Methods Appl. Mech. Eng. **397**, 115076 (2022). https://doi.org/10.1016/j.cma.2022.115076
7. Van Beeumen, R., Meerbergen, K., Michiels, W.: A rational Krylov method based on Hermite interpolation for nonlinear eigenvalue problems. SIAM J. Sci. Comput. **35**(1), 327–350 (2013). https://doi.org/10.1137/120877556
8. Cohen, G., Hauck, A., Kaltenbacher, M., Otsuru, T.: Different types of finite elements. In: Marburg, S., Nolte, B. (eds.) Computational Acoustics of Noise Propagation in Fluids – Finite and Boundary Element Methods, pp. 57–88. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-77448-8_3
9. Antoulas, A.C.: Approximation of large-scale dynamical systems. Adv. Des. Control, vol. 6. SIAM, Philadelphia, PA (2005). https://doi.org/10.1137/1.9780898718713
10. Deckers, E., Desmet, W., Meerbergen, K., Naets, F.: Case studies of model order reduction for acoustics and vibrations. In: Benner, P., Schilders, W., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Silveira, L.M. (eds.) Model Order Reduction. Volume 3: Applications, pp. 76–110. De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110499001-003
11. Hetmaniuk, U., Tezaur, R., Farhat, C.: Review and assessment of interpolatory model order reduction methods for frequency response structural dynamics and acoustics problems. Int. J. Numer. Methods Eng. **90**(13), 1636–1662 (2012). https://doi.org/10.1002/nme.4271
12. Aumann, Q., Werner, S.W.R.: Structured model order reduction for vibro-acoustic problems using interpolation and balancing methods. J. Sound Vib. **543**, 117363 (2023). https://doi.org/10.1016/j.jsv.2022.117363
13. Beddig, R.S., Benner, P., Dorschky, I., Reis, T., Schwerdtner, P., Voigt, M., Werner, S.W.R.: Structure-preserving model reduction for dissipative mechanical systems. In: Eberhard, P. (ed.) Calm, Smooth and Smart. Lect. Notes Appl. Comput. Mech., vol. 102, pp. 209–230. Springer, Cham (2024). https://doi.org/10.1007/978-3-031-36143-2_11
14. Cheng, X., Kawano, Y., Scherpen, J.M.A.: Reduction of second-order network systems with structure preservation. IEEE Trans. Autom. Control. **62**(10), 5026–5038 (2017). https://doi.org/10.1109/TAC.2017.2679479

15. Egger, H., Kugler, T., Liljegren-Sailer, B., Marheineke, N., Mehrmann, V.: On structure-preserving model reduction for damped wave propagation in transport networks. SIAM J. Sci. Comput. **40**(1), 331–365 (2018). https://doi.org/10.1137/17M1125303
16. Bendokat, T., Zimmermann, R.: Geometric optimization for structure-preserving model reduction of Hamiltonian systems. IFAC-Pap. **55**(20), 457–462 (2022). https://doi.org/10.1016/j.ifacol.2022.09.137. 10th Vienna International Conference on Mathematical Modelling MATHMOD 2022
17. Gugercin, S., Polyuga, R.V., Beattie, C., Schaft, A.: Structure-preserving tangential interpolation for model reduction of port-Hamiltonian systems. Automatica J. IFAC. **48**(9), 1963–1974 (2012). https://doi.org/10.1016/j.automatica.2012.05.052
18. Hesthaven, J.S., Pagliantini, C., Ripamonti, N.: Rank-adaptive structure-preserving model order reduction of Hamiltonian systems. ESAIM: Math. Model. Numer. Anal. **56**(2), 617–650 (2022). https://doi.org/10.1051/m2an/2022013
19. Chellappa, S., Feng, L., Benner, P.: An adaptive sampling approach for the reduced basis method. In: Beattie, C., Benner, P., Embree, M., Gugercin, S., Lefteriu, S. (eds.) Realization and Model Reduction of Dynamical Systems, pp. 137–155. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-95157-3_8
20. Bonin, T., Faßbender, H., Soppa, A., Zaeh, M.: A fully adaptive rational global Arnoldi method for the model-order reduction of second-order MIMO systems with proportional damping. Math. Comput. Simul. **122**, 1–19 (2016). https://doi.org/10.1016/j.matcom.2015.08.017
21. Feng, L., Korvink, J.G., Benner, P.: A fully adaptive scheme for model order reduction based on moment matching. IEEE Trans. Compon. Packag. Manuf. Technol. **5**(12), 1872–1884 (2015). https://doi.org/10.1109/TCPMT.2015.2491341
22. Rumpler, R., Göransson, P., Deü, J.-F.: A finite element approach combining a reduced-order system, Padé approximants, and an adaptive frequency windowing for fast multi-frequency solution of poro-acoustic problems. Int. J. Numer. Methods Eng. **97**(10), 759–784 (2014). https://doi.org/10.1002/nme.4609
23. Panzer, H.K.F., Wolf, T., Lohmann, B.: H_2 and H_∞ error bounds for model order reduction of second order systems by Krylov subspace methods. In: 2013 European Control Conference (ECC), pp. 4484–4489 (2013). https://doi.org/10.23919/ECC.2013.6669657
24. Aumann, Q., Müller, G.: Robust error assessment for reduced order vibro-acoustic problems. J. Sound Vib. **545**, 117427 (2023). https://doi.org/10.1016/j.jsv.2022.117427
25. Feng, L., Lombardi, L., Benner, P., Romano, D., Antonini, G.: Model order reduction for delayed PEEC models with guaranteed accuracy and observed stability. IEEE Trans. Circuits Syst. I: Regul. Pap. **69**(10), 4177–4190 (2022). https://doi.org/10.1109/TCSI.2022.3189389
26. Aliyev, N., Benner, P., Mengi, E., Voigt, M.: A subspace framework for $\mathcal{H}_\infty$-norm minimization. SIAM J. Matrix Anal. Appl. **41**(2), 928–956 (2020). https://doi.org/10.1137/19M125892X
27. Schwerdtner, P., Voigt, M.: Adaptive sampling for structure-preserving model order reduction of port-Hamiltonian systems. IFAC-Pap. **54**(19), 143–148 (2021). https://doi.org/10.1016/j.ifacol.2021.11.069. 7th IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control LHMNC 2021
28. Gugercin, S., Antoulas, A.C., Beattie, C.: $\mathcal{H}_2$ model reduction for large-scale linear dynamical systems. SIAM J. Matrix Anal. Appl. **30**(2), 609–638 (2008). https://doi.org/10.1137/060666123
29. Wyatt, S.: Issues in interpolatory model reduction: inexact solves, second-order systems and DAEs. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA (2012). http://hdl.handle.net/10919/27668
30. Aumann, Q., Müller, G.: An adaptive method for reducing second-order dynamical systems. IFAC-Pap. **55**(20), 337–342 (2022). https://doi.org/10.1016/j.ifacol.2022.09.118. 10th Vienna International Conference on Mathematical Modelling MATHMOD 2022
31. Beattie, C.A., Gugercin, S.: Realization-independent $\mathcal{H}_2$-approximation. In: 51st IEEE Conference on Decision and Control (CDC), pp. 4953–4958 (2012). https://doi.org/10.1109/CDC.2012.6426344
32. Sinani, K., Gugercin, S., Beattie, C.: A structure-preserving model reduction algorithm for dynamical systems with nonlinear frequency dependence. IFAC-Pap. **49**(9), 56–61 (2016). https://doi.org/10.1016/j.ifacol.2016.07.492. 6th IFAC Symposium on System Structure and Control SSSC 2016
33. Mayo, A.J., Antoulas, A.C.: A framework for the solution of the generalized realization problem. Linear Algebra Appl. **425**(2–3), 634–662 (2007). https://doi.org/10.1016/j.laa.2007.03.008. Special issue in honor of P. A. Fuhrmann, Edited by A. C. Antoulas, U. Helmke, J. Rosenthal, V. Vinnikov, and E. Zerz

34. Schulze, P., Unger, B., Beattie, C., Gugercin, S.: Data-driven structured realization. Linear Algebra Appl. **537**, 250–286 (2018). https://doi.org/10.1016/j.laa.2017.09.030
35. Gosea, I.V., Gugercin, S., Werner, S.W.R.: Structured barycentric forms for interpolation-based data-driven reduced modeling of second-order systems. e-print 2303.12576, arXiv (2023). https://doi.org/10.48550/arXiv.2303.12576. Numerical Analysis (math.NA)
36. Werner, S.W.R., Gosea, I.V., Gugercin, S.: Structured vector fitting framework for mechanical systems. IFAC-Pap. **55**(20), 163–168 (2022). https://doi.org/10.1016/j.ifacol.2022.09.089. 10th Vienna International Conference on Mathematical Modelling MATHMOD 2022
37. Antoulas, A.C., Lefteriu, S., Ionita, A.C.: A tutorial introduction to the Loewner framework for model reduction. In: Benner, P., Ohlberger, M., Cohen, A., Willcox, K. (eds.) Model Reduction and Approximation: Theory and Algorithms. Computational Science & Engineering, pp. 335–376. SIAM, Philadelphia, PA (2017). https://doi.org/10.1137/1.9781611974829.ch8
38. Castagnotto, A., Lohmann, B.: A new framework for $\mathcal{H}_2$-optimal model reduction. Math. Comput. Model. Dyn. Syst. **24**(3), 236–257 (2018). https://doi.org/10.1080/13873954.2018.1464030
39. Benner, P., Grundel, S., Hornung, N.: Parametric model order reduction with a small $\mathcal{H}_2$-error using radial basis functions. Adv. Comput. Math. **41**(5), 1231–1253 (2015). https://doi.org/10.1007/s10444-015-9410-7
40. Vuillemin, P., Poussot-Vassal, C., Alazard, D.: $\mathcal{H}_2$ optimal and frequency limited approximation methods for large-scale LTI dynamical systems. IFAC Proc. Vol. **46**(2), 719–724 (2013). https://doi.org/10.3182/20130204-3-FR-2033.00061. 5th IFAC Symposium on System Structure and Control
41. Karachalios, D.S., Gosea, I.V., Antoulas, A.C.: The Loewner framework for system identification and reduction. In: Benner, P., Schilders, W., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Silveira, L.M. (eds.) Model Order Reduction. Volume 1: System- and Data-Driven Methods and Algorithms, pp. 181–228. De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110498967-006
42. Gustavsen, B., Semlyen, A.: Rational approximation of frequency domain responses by vector fitting. IEEE Trans. Power Del. **14**(3), 1052–1061 (1999). https://doi.org/10.1109/61.772353
43. Drmač, Z., Gugercin, S., Beattie, C.: Quadrature-based vector fitting for discretized $\mathcal{H}_2$ approximation. SIAM J. Sci. Comput. **37**(2), 625–652 (2015). https://doi.org/10.1137/140961511
44. Berljafa, M., Güttel, S.: The RKFIT algorithm for nonlinear rational approximation. SIAM J. Sci. Comput. **39**(5), 2049–2071 (2017). https://doi.org/10.1137/15M1025426
45. Nakatsukasa, Y., Sète, O., Trefethen, L.N.: The AAA algorithm for rational approximation. SIAM J. Sci. Comput. **40**(3), 1494–1522 (2018). https://doi.org/10.1137/16M1106122
46. Martins, N., Lima, L.T.G., Pinto, H.J.C.P.: Computing dominant poles of power system transfer functions. IEEE Trans. Power Syst. **11**(1), 162–170 (1996). https://doi.org/10.1109/59.486093
47. Gugercin, S., Stykel, T., Wyatt, S.: Model reduction of descriptor systems by interpolatory projection methods. SIAM J. Sci. Comput. **35**(5), 1010–1033 (2013). https://doi.org/10.1137/130906635
48. Borggaard, J.T., Gugercin, S.: Model reduction for DAEs with an application to flow control. In: King, R. (ed.) Active Flow and Combustion Control 2014. Notes on Numerical Fluid Mechanics and Multidisciplinary Design, vol. 127, pp. 381–396. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-11967-0_23
49. Saak, J., Köhler, M., Benner, P.: M-M.E.S.S. – The matrix equations sparse solvers library (version 2.2). see also: https://www.mpi-magdeburg.mpg.de/projects/mess (2022). https://doi.org/10.5281/zenodo.5938237
50. Aumann, Q., Werner, S.W.R.: Code, data and results for the numerical experiments in "Adaptive choice of near-optimal expansion points for interpolation-based structure-preserving model reduction" (version 1.2) (2024). https://doi.org/10.5281/zenodo.10945494
51. Gugercin, S., Antoulas, A.C., Bedrossian, M.: Approximation of the international space station 1R and 12A models. In: Proceedings of the 40th IEEE Conference on Decision and Control, pp. 1515–1516 (2001). https://doi.org/10.1109/CDC.2001.981109
52. Benner, P., Gugercin, S., Werner, S.W.R.: A unifying framework for tangential interpolation of structured bilinear control systems. Numer. Math. **155**(3–4), 445–483 (2023). https://doi.org/10.1007/s00211-023-01380-w
53. Higham, N.J., Negri Porzio, G.M., Tisseur, F.: An updated set of nonlinear eigenvalue problems. e-print 2019.5, MIMS (2019). http://eprints.maths.manchester.ac.uk/id/eprint/2699
54. Liao, B.-S.: Subspace projection methods for model order reduction and nonlinear eigenvalue computation. Dissertation, University of California, Davis, California, USA (2007)