



Structure-preserving interpolation of bilinear control systems

Peter Benner^{1,2} · Serkan Gugercin³ · Steffen W. R. Werner¹

Received: 14 July 2020 / Accepted: 23 March 2021 / Published online: 11 May 2021
© The Author(s) 2021

Abstract

In this paper, we extend the structure-preserving interpolatory model reduction framework, originally developed for linear systems, to structured bilinear control systems. Specifically, we give explicit construction formulae for the model reduction bases to satisfy different types of interpolation conditions. First, we establish the analysis for transfer function interpolation for single-input single-output structured bilinear systems. Then, we extend these results to the case of multi-input multi-output structured bilinear systems by matrix interpolation. The effectiveness of our structure-preserving approach is illustrated by means of various numerical examples.

Keywords Model reduction · Bilinear systems · Structure-preserving approximation · Structured interpolation

Mathematics Subject Classification (2010) 30E05 · 34K17 · 65D05 · 93C10 · 93A15 · 93C35

Communicated by: Jan Hesthaven

✉ Steffen W. R. Werner
werner@mpi-magdeburg.mpg.de

Peter Benner
benner@mpi-magdeburg.mpg.de

Serkan Gugercin
gugercin@vt.edu

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

² Faculty of Mathematics, Otto von Guericke University, Universitätsplatz 2, 39106 Magdeburg, Germany

³ Department of Mathematics and Computational Modeling and Data Analytics Division, Academy of Integrated Science, Virginia Tech, Blacksburg, VA 24061, USA

1 Introduction

The modeling of various real-world applications and processes results in dynamical control systems usually including nonlinearities. Since linear approximations are very often incapable of capturing all the features of nonlinear systems, they are an insufficient description for use in optimization and controller design. A special class of nonlinear systems are bilinear control systems, which contain the multiplication of control and state variables, i.e., they are linear in state and control separately, but not together [1]. In the last decades, the class of bilinear systems became an essential tool in systems theory. They naturally appear in the modeling process of many physical phenomena, e.g., in the modeling of population, economical, thermal, and mechanical dynamics [1, 2], of electrical circuits [3], of plasma devices [4, 5], or of medical processes [6]. Bilinear systems can also result from approximation of general nonlinear systems employing the Carleman linearization process [7, 8]. Moreover, bilinear systems are nowadays often used in the parameter control of partial differential equations (PDEs) [9, 10]. Looking back to the linear case, bilinear systems can be used as a generalizing framework in the modeling of linear stochastic [11] and parameter-varying systems [12–14], allowing the application of established system-theoretic tools such as model order reduction for those system classes.

In this paper, we focus on structured bilinear systems. Those structures arise from the underlying physical phenomena. For example, in case of bilinear time-delay systems, one has the bilinear control system defined by

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + A_d x(t - \tau) + \sum_{j=1}^m N_j x(t) u_j(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \quad (1)$$

with $E, A, A_d, N_j \in \mathbb{R}^{n \times n}$ for all $j = 1, \dots, m$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ and a constant delay $0 \leq \tau \in \mathbb{R}$. In (1), $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$ denote, respectively, the states (degrees of freedom), inputs (forcing terms), and the outputs (quantities of interest) of the underlying dynamical system. Due to the usual demand for increasing accuracy in applications, the number of differential equations n , describing the dynamics of systems as in (1) quickly increases, resulting in a high demand on computational resources such as time and memory. One remedy is model order reduction: a *reduced* system is created, consisting of a significantly smaller number of differential equations than needed to define the original one while still accurately approximating the input-to-output behavior. Then, one can use this lower-order approximation as a surrogate model for faster simulations or the design of controllers. The classical (unstructured) bilinear first-order systems are described by the state-space representation

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + \sum_{j=1}^m N_j x(t) u_j(t) + Bu(t), \\ y(t) &= Cx(t). \end{aligned} \quad (2)$$

There are different methodologies for model reduction of (2), e.g., the bilinear balanced truncation method [3, 11, 15], different types of moment matching approaches

for the underlying multi-variate transfer functions in the frequency domain [16–20], the interpolation of complete Volterra series [21–23], or even the construction of reduced-order bilinear systems from frequency data with the bilinear Loewner framework [24, 25].

While some structured system classes (see Section 2.3.2) can actually be rewritten into classical bilinear systems (2), the original structure is completely lost, which can lead to undesirable results in terms of accuracy, stability, or physical interpretation. Moreover, other structured bilinear systems, such as those with internal delays (1), cannot be represented in the form (2). Therefore, here we develop a structure-preserving model reduction approach for different system structures involving bilinear terms. Following [26], which studied structured linear dynamical systems, our goal is to generalize the structured interpolation approach to a general set of multivariate transfer functions associated with different structured bilinear control systems to preserve the system structure in the reduced-order model. The question we aim to answer is how we can construct an interpolatory reduced-order model of, e.g., (1), that has the same structure. Towards this goal, we develop a structure-preserving interpolation framework for this special class of nonlinear systems, namely the structured bilinear control systems, thus extending the theoretical analysis and computational framework developed by [26] for linear systems to bilinear control systems.

In Section 2, we review the theory for classical first-order bilinear systems and motivate the more general structure, we will consider, via two examples. Section 3 gives subspace construction formulae for interpolatory model reduction bases in the case of single-input single-output (SISO) systems and suggests reasonable choices for interpolation points, for which the effectiveness of the approach is illustrated by employing two numerical examples. The developed theory is then extended further in Section 4 to the multi-input multi-output (MIMO) case by matrix interpolation. Section 5 concludes the paper.

2 Structured bilinear systems

In this section, we present the basic properties of the structured bilinear systems considered in this paper. To clarify the presentation, we first revisit the unstructured (classical) bilinear control systems as given in (2) and then generalize these concepts to the structured case.

2.1 Revisiting the classical first-order bilinear systems

Given the unstructured bilinear system (2), define $N = [N_1 \dots N_m]$ and let I_{m^k} be the identity matrix of dimension m^k . Assuming for simplicity E to be invertible, the initial condition $x(0) = 0$, and some additional mild conditions, the output of (2) can be expressed in terms of a Volterra series [27], i.e.,

$$y(t) = \sum_{k=1}^{\infty} \int_0^t \int_0^{t_1} \dots \int_0^{t_{k-1}} g_k(t_1, \dots, t_k) \left(u \left(t - \sum_{i=1}^j t_i \right) \otimes \dots \otimes u(t - t_1) \right) dt_k \dots dt_1,$$

where g_k , for $k \geq 1$, is the k -th regular Volterra kernel given by

$$g_k(t_1, \dots, t_k) = C e^{E^{-1} A t_k} \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes E^{-1} N) (I_{m^j} \otimes e^{E^{-1} A t_{k-j}}) \right) \times (I_{m^{k-1}} \otimes E^{-1} B), \quad (3)$$

with $\otimes$ denoting the Kronecker product and $\times$ the continuation of the classical matrix multiplication to the next line. Using the multivariate Laplace transform [27], the regular Volterra kernels (3) yield a representation of (2) in the frequency domain by the so-called regular subsystem transfer functions

$$G_k(s_1, \dots, s_k) = C(s_k E - A)^{-1} \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes N) (I_{m^j} \otimes (s_{k-j} E - A)^{-1}) \right) \times (I_{m^{k-1}} \otimes B), \quad (4)$$

with $s_1, \dots, s_k \in \mathbb{C}$. This compact expression is actually the collection of the different combinations of the bilinear matrices, i.e., we can write (4) as

$$\begin{aligned} G_k(s_1, \dots, s_k) = & [C(s_k E - A)^{-1} N_1 \cdots N_1 (s_1 E - A)^{-1} B, \\ & C(s_k E - A)^{-1} N_1 \cdots N_2 (s_1 E - A)^{-1} B, \\ & \dots \\ & C(s_k E - A)^{-1} N_1 \cdots N_m (s_1 E - A)^{-1} B, \\ & \dots \\ & C(s_k E - A)^{-1} N_m \cdots N_m (s_1 E - A)^{-1} B]. \end{aligned} \quad (5)$$

For SISO systems, (4) simplifies to

$$G_k(s_1, \dots, s_k) = C(s_k E - A)^{-1} \left(\prod_{j=1}^{k-1} N(s_{k-j} E - A)^{-1} \right) B.$$

As stated in Section 1, for the unstructured bilinear system case (2), there are already different model reduction techniques. For the structured bilinear systems we consider here, we will concentrate on interpolatory methods.

Note that the assumption of E being invertible is only made for ease of presentation. The interpolation theory and interpolatory properties of the reduced-order model developed in the following sections hold for the general situation, yet the final construction of the reduced-order model might need some additional treatment as in the linear and unstructured bilinear cases; see, e.g., [28–30].

2.2 Moving from classical to structured bilinear systems

For the transition from unstructured to structured bilinear systems, we start by recalling the case of linear systems. The classical (unstructured) linear dynamical systems are described, in state-space, by

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

with $E, A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{p \times n}$. Assuming the initial condition $Ex(0) = 0$, the Laplace transform maps this problem to the frequency domain:

$$\begin{aligned} (sE - A)X(s) &= BU(s), \\ Y(s) &= CX(s), \end{aligned} \quad (6)$$

where $X(s)$, $U(s)$, and $Y(s)$ denote the Laplace transforms of the time-dependent functions $x(t)$, $u(t)$, and $y(t)$. Inspired by much richer structured systems than (6) appearing in the linear case such as those describing the dynamic response of a viscoelastic body, [26] introduced a more general system of equations in the frequency domain, given by

$$\begin{aligned} \mathcal{K}(s)X(s) &= \mathcal{B}(s)U(s), \\ Y(s) &= \mathcal{C}(s)X(s), \end{aligned} \quad (7)$$

with matrix-valued functions $\mathcal{K}: \mathbb{C} \rightarrow \mathbb{C}^{n \times n}$, $\mathcal{B}: \mathbb{C} \rightarrow \mathbb{C}^{n \times m}$ and $\mathcal{C}: \mathbb{C} \rightarrow \mathbb{C}^{p \times n}$. Note that (7) contains (6) as a special case. Assuming the problem to be regular, i.e., there exists an $s \in \mathbb{C}$ for which the matrix functions are defined and $\mathcal{K}(s)$ is full-rank, the problem (7) leads to the general formulation of structured transfer functions of linear systems

$$G_{\text{lin}}(s) = \mathcal{C}(s)\mathcal{K}(s)^{-1}\mathcal{B}(s), \quad (8)$$

describing the input-to-output behavior in the frequency domain.

Inspired by (8) and the structure of the examples in Sections 2.3.1 and 2.3.2, we consider here a more general, structured formulation of the regular subsystem transfer functions corresponding to structured bilinear systems, namely

$$\begin{aligned} G_k(s_1, \dots, s_k) &= \mathcal{C}(s_k)\mathcal{K}(s_k)^{-1} \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes \mathcal{N}(s_{k-j})) (I_{m^j} \otimes \mathcal{K}(s_{k-j})^{-1}) \right) \\ &\quad \times (I_{m^{k-1}} \otimes \mathcal{B}(s_1)), \end{aligned} \quad (9)$$

for $k \geq 1$ and where $\mathcal{N}(s) = [\mathcal{N}_1(s) \dots \mathcal{N}_m(s)]$ with the matrix functions $\mathcal{C}: \mathbb{C} \rightarrow \mathbb{C}^{p \times n}$, $\mathcal{K}: \mathbb{C} \rightarrow \mathbb{C}^{n \times n}$, $\mathcal{B}: \mathbb{C} \rightarrow \mathbb{C}^{n \times m}$, $\mathcal{N}_j: \mathbb{C} \rightarrow \mathbb{C}^{n \times n}$ for $j = 1, \dots, m$. This general formulation includes transfer functions of classical bilinear systems (4) since we can choose

$$\mathcal{C}(s) = C, \quad \mathcal{K}(s) = sE - A, \quad \mathcal{N}(s) = N, \quad \mathcal{B}(s) = B.$$

Section 2.3 gives two examples of structured system classes that can be formulated in this general setting.

For the construction of *structured reduced-order bilinear models*, we will use the projection approach, i.e., we will construct two model reduction bases $W, V \in \mathbb{C}^{n \times r}$ such that the reduced-order bilinear system quantities will be computed by

$$\begin{aligned}\widehat{\mathcal{C}}(s) &= \mathcal{C}(s)V, & \widehat{\mathcal{K}}(s) &= W^H \mathcal{K}(s)V, \\ \widehat{\mathcal{B}}(s) &= W^H \mathcal{B}(s), & \widehat{\mathcal{N}}_j(s) &= W^H \mathcal{N}_j(s)V,\end{aligned}\quad (10)$$

for $j = 1, \dots, m$. Here, $W^H := \overline{W}^T$ denotes the conjugate transposed of the matrix W . The structured reduced-order bilinear control system $\widehat{G}$ is then given by the underlying reduced-order matrices from (10) and with the corresponding structured regular subsystem transfer functions

$$\begin{aligned}\widehat{G}_k(s_1, \dots, s_k) &= \widehat{\mathcal{C}}(s_k) \widehat{\mathcal{K}}(s_k)^{-1} \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes \widehat{\mathcal{N}}(s_{k-j})) (I_{m^j} \otimes \widehat{\mathcal{K}}(s_{k-j})^{-1}) \right) \\ &\quad \times (I_{m^{k-1}} \otimes \widehat{\mathcal{B}}(s_1)),\end{aligned}$$

for $k \geq 1$.

2.3 Examples for structured bilinear systems

We will consider, in particular, the following two system classes as examples for structured bilinear systems in this paper.

2.3.1 Bilinear time-delay systems

First, we revisit the introductory example of bilinear time-delay systems (1) from Section 1. It was shown in [25] that the regular subsystem transfer functions of (1) are given by

$$\begin{aligned}G_k(s_1, \dots, s_k) &= C(s_k E - A - e^{-s_k \tau} A_d)^{-1} \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes N) \right. \\ &\quad \left. \times (I_{m^j} \otimes (s_{k-j} E - A - e^{-s_{k-j} \tau} A_d)^{-1}) \right) (I_{m^{k-1}} \otimes B).\end{aligned}\quad (11)$$

We can directly see that (11) can be written in the setting of (9) using

$$\mathcal{C}(s) = C, \quad \mathcal{K}(s) = sE - A - e^{-s\tau} A_d, \quad \mathcal{N}(s) = N, \quad \text{and} \quad \mathcal{B}(s) = B.$$

Once the model reduction bases W and V are constructed, the resulting reduced-order model retains the delay structure of the original system as it is given by

$$\begin{aligned}\widehat{\mathcal{C}}(s) &= CV, & \widehat{\mathcal{K}}(s) &= s(W^H EV) - (W^H AV) - e^{-s\tau} (W^H A_d V), \\ \widehat{\mathcal{N}}(s) &= W^H N(I_m \otimes V), & \widehat{\mathcal{B}}(s) &= W^H B.\end{aligned}$$

In Sections 3 and 4, we will show how to construct the model reduction bases W and V such that the structured reduced-order bilinear control system provides interpolation of the full-order subsystems.

2.3.2 Bilinear second-order systems

Another example for structured bilinear control systems is the case of bilinear mechanical systems, i.e.,

$$M\ddot{q}(t) + D\dot{q}(t) + Kq(t) = \sum_{j=1}^m N_{p,j}q(t)u_j(t) + \sum_{j=1}^m N_{v,j}\dot{q}(t)u_j(t) + B_u u(t), \quad (12)$$

$$y(t) = C_p q(t) + C_v \dot{q}(t),$$

with $M, D, K, N_{p,j}, N_{v,j} \in \mathbb{R}^{n \times n}$ for all $j = 1, \dots, m$, $B_u \in \mathbb{R}^{n \times m}$ and $C_p, C_v \in \mathbb{R}^{p \times n}$. We note that (12) can be rewritten in the first-order (unstructured) form (2) by introducing the new state vector $x(t) = [q^T(t), \dot{q}^T(t)]^T$ such that we obtain

$$\underbrace{\begin{bmatrix} J & 0 \\ 0 & M \end{bmatrix}}_E \dot{x}(t) = \underbrace{\begin{bmatrix} 0 & J \\ -K & -D \end{bmatrix}}_A x(t) + \sum_{j=1}^m \underbrace{\begin{bmatrix} 0 & 0 \\ N_{p,j} & N_{v,j} \end{bmatrix}}_{N_j} x(t)u_j(t) + \underbrace{\begin{bmatrix} 0 \\ B_u \end{bmatrix}}_B u(t), \quad (13)$$

$$y(t) = \underbrace{\begin{bmatrix} C_p & C_v \end{bmatrix}}_C x(t),$$

for any invertible matrix $J \in \mathbb{R}^{n \times n}$. For this first-order companion realization (13), we know the frequency domain representation to be given by the multivariate regular transfer functions (4). If we now plug in the structured matrices from (13), we can make use of those special block structures. In general, we obtain

$$(sE - A)^{-1} = \begin{bmatrix} sJ & -J \\ K & sM + E \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} \frac{1}{s}J^{-1} - \frac{1}{s}(s^2M + sD + K)^{-1}KJ^{-1} & (s^2M + sD + K)^{-1} \\ -(s^2M + sD + K)^{-1}KJ^{-1} & s(s^2M + sD + K)^{-1} \end{bmatrix}$$

for the frequency-dependent center terms and, therefore,

$$N_j(sE - A)^{-1}B = \begin{bmatrix} 0 \\ (N_{p,j} + sN_{v,j})(s^2M + sD + K)^{-1}B_u \end{bmatrix}.$$

Using this, we obtain for the first part of the k -th subsystem transfer function

$$\left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes N)(I_{m^j} \otimes (s_{k-j}E - A)^{-1}) \right) (I_{m^{k-1}} \otimes B)$$

$$= \begin{bmatrix} 0 \\ \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes (N_p + s_{k-j}N_v))(I_{m^j} \otimes (s_{k-j}^2M + s_{k-j}D + K)^{-1}) \right) (I_{m^{k-1}} \otimes B_u) \end{bmatrix},$$

where we used the notion $N_p = [N_{p,1} \dots N_{p,m}]$ and $N_v = [N_{v,1} \dots N_{v,m}]$. Multiplication with the remaining terms yields the regular subsystem transfer functions of (12) to be written in the form

$$\begin{aligned} G_k(s_1, \dots, s_k) &= (C_p + s_k C_v)(s_k^2 M + s_k D + K)^{-1} \\ &\times \left(\prod_{j=1}^{k-1} (I_{m^{j-1}} \otimes (N_p + s_{k-j} N_v)) \right. \\ &\times \left. (I_{m^j} \otimes (s_{k-j}^2 M + s_{k-j} D + K)^{-1}) \right) (I_{m^{k-1}} \otimes B_u). \end{aligned} \quad (14)$$

Having the general formulation of regular transfer functions (9) in mind, we see that we can rewrite (14) in the structured bilinear form (9) by setting

$$\mathcal{C}(s) = C_p + s C_v, \quad \mathcal{K}(s) = s^2 M + s D + K, \quad \mathcal{N}(s) = N_p + s N_v, \quad \mathcal{B}(s) = B_u.$$

Now assume that we construct model reduction bases W and V , and compute the reduced-order model by projection as in (10). This leads to the reduced-order bilinear system

$$\begin{aligned} \widehat{\mathcal{C}}(s) &= C_p V + s(C_v V), \\ \widehat{\mathcal{K}}(s) &= s^2(W^H M V) + s(W^H D V) + (W^H K V), \\ \widehat{\mathcal{N}}(s) &= (W^H N_p(I_m \otimes V)) + s(W^H N_v(I_m \otimes V)), \\ \widehat{\mathcal{B}}(s) &= W^H B_u. \end{aligned} \quad (15)$$

Note that the reduced-order bilinear system in (15) has the same structure as the original one and can be viewed as a reduced second-order bilinear system, where the full-order matrices in (12) are simply replaced by the reduced analogues from (15).

3 Interpolation of single-input single-output systems

In this section, we assume the SISO system case, i.e., $m = p = 1$. Therefore, the bilinear part consists of, at most, one term $\mathcal{N} = \mathcal{N}_1$ and the matrix functionals $\mathcal{C}$ and $\mathcal{B}$ map frequency points only onto row and column vectors, respectively. In this setting, the regular subsystem transfer functions drastically simplify since (9) can now be written as

$$G_k(s_1, \dots, s_k) = \mathcal{C}(s_k) \mathcal{K}(s_k)^{-1} \left(\prod_{j=1}^{k-1} \mathcal{N}(s_{k-j}) \mathcal{K}(s_{k-j})^{-1} \right) \mathcal{B}(s_1), \quad (16)$$

for $k \geq 1$. In the remainder of this section, we develop the theory for structure-preserving interpolation (both the case of simple and high-order (Hermite) interpolation) and then present numerical examples to illustrate the analysis.

3.1 Structured transfer function interpolation

We want to construct the model reduction bases W and V , and the corresponding reduced structured-bilinear system via projection as in (10) such that its leading regular subsystem transfer functions interpolate those of the original one; i.e., $G_k(\sigma_1, \dots, \sigma_k) = \widehat{G}_k(\sigma_1, \dots, \sigma_k)$, where $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ are some selected interpolation points.

The following two theorems answer the question of how the model reduction bases V and W can be constructed independent of each other. In other words, the interpolation conditions are satisfied only via V or W , no matter how the respective other matrix is chosen. First, we consider the model reduction basis V .

Theorem 1 (Interpolation via V) *Let G be a bilinear SISO system, described by (16), and $\widehat{G}$ the reduced-order bilinear SISO system constructed by (10). Let $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ be interpolation points for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$ and $\mathcal{B}(s)$ are defined and $\widehat{\mathcal{K}}(s)$ is full-rank. Construct V using*

$$\begin{aligned} v_1 &= \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1), \\ v_j &= \mathcal{K}(\sigma_j)^{-1} \mathcal{N}(\sigma_{j-1}) v_{j-1}, & 2 \leq j \leq k, \\ \text{span}(V) &\supseteq \text{span}([v_1, \dots, v_k]). \end{aligned}$$

Let W be an arbitrary full-rank truncation matrix of appropriate dimension. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:

$$\begin{aligned} G_1(\sigma_1) &= \widehat{G}_1(\sigma_1), \\ G_2(\sigma_1, \sigma_2) &= \widehat{G}_2(\sigma_1, \sigma_2), \\ &\vdots \\ G_k(\sigma_1, \dots, \sigma_k) &= \widehat{G}_k(\sigma_1, \dots, \sigma_k). \end{aligned}$$

Proof First, we note that the constructed vectors are given by

$$\begin{aligned} v_1 &= \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1), \\ v_2 &= \mathcal{K}(\sigma_2)^{-1} \mathcal{N}(\sigma_1) \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1), \\ &\vdots \\ v_k &= \mathcal{K}(\sigma_k)^{-1} \mathcal{N}(\sigma_{k-1}) \mathcal{K}(\sigma_{k-1})^{-1} \cdots \mathcal{K}(\sigma_1) \mathcal{B}(\sigma_1), \end{aligned}$$

and that by construction all those vectors are contained in $\text{span}(V)$. Therefore, for the first subsystem transfer function, we obtain

$$\begin{aligned} \widehat{G}_1(\sigma_1) &= \widehat{\mathcal{C}}(\sigma_1) \widehat{\mathcal{K}}(\sigma_1)^{-1} \widehat{\mathcal{B}}(\sigma_1) \\ &= \mathcal{C}(\sigma_1) V (W^H \mathcal{K}(\sigma_1) V)^{-1} W^H \mathcal{B}(\sigma_1) \\ &= \mathcal{C}(\sigma_1) \underbrace{V (W^H \mathcal{K}(\sigma_1) V)^{-1} W^H \mathcal{K}(\sigma_1)^{-1}}_{=: P_{v_1}} \mathcal{B}(\sigma_1) \\ &= \mathcal{C}(\sigma_1) \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1) \\ &= G_1(\sigma_1), \end{aligned}$$

where we used the fact that P_{v_1} is an oblique projector onto $\text{span}(V)$, i.e., $z = P_{v_1}z$ holds for all $z \in \text{span}(V)$, and that $\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1) = v_1 \in \text{span}(V)$. Considering the second subsystem transfer function, we get

$$\begin{aligned}\widehat{G}_2(\sigma_1, \sigma_2) &= \widehat{\mathcal{C}}(\sigma_2)\widehat{\mathcal{K}}(\sigma_2)^{-1}\widehat{\mathcal{N}}(\sigma_1)\widehat{\mathcal{K}}(\sigma_1)^{-1}\widehat{\mathcal{B}}(\sigma_1) \\ &= \mathcal{C}(\sigma_2)V(W^H\mathcal{K}(\sigma_2)V)^{-1}W^H\mathcal{N}(\sigma_1)V(W^H\mathcal{K}(\sigma_1)V)^{-1}W^H\mathcal{B}(\sigma_1) \\ &= \mathcal{C}(\sigma_2)V(W^H\mathcal{K}(\sigma_2)V)^{-1}W^H\mathcal{N}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1) \\ &= \mathcal{C}(\sigma_2)\underbrace{V(W^H\mathcal{K}(\sigma_2)V)^{-1}W^H\mathcal{K}(\sigma_2)}_{=: P_{v_2}}\mathcal{K}(\sigma_2)^{-1}\mathcal{N}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1) \\ &= \mathcal{C}(\sigma_2)\mathcal{K}(\sigma_2)^{-1}\mathcal{N}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1) \\ &= G_2(\sigma_1, \sigma_2),\end{aligned}$$

using the same arguments as for the first transfer function and, additionally, the construction of v_2 and the oblique projector P_{v_2} . Continuing with this argument, the desired result follows by induction over the transfer function index k . $\square$

The proof of Theorem 1 shows that the recursive construction of the truncation matrix is necessary for the interpolation of higher-order transfer functions. Also, it should be noted that W was an arbitrary full-rank truncation matrix of suitable dimensions but with no additional constraints for the interpolation of (16). Theorem 2 is the counterpart to Theorem 1 by only giving constraints for the left model reduction basis W , while V is now allowed to be arbitrary.

Theorem 2 (Interpolation via W) *Let G , $\widehat{G}$, and the interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ be as in Theorem 1. Construct W using*

$$\begin{aligned}w_1 &= \mathcal{K}(\sigma_k)^{-H}\mathcal{C}(\sigma_k)^H, \\ w_j &= \mathcal{K}(\sigma_{k-j+1})^{-H}\mathcal{N}(\sigma_{k-j+1})^H w_{j-1}, \quad 2 \leq j \leq k, \\ \text{span}(W) &\supseteq \text{span}([w_1, \dots, w_k]).\end{aligned}$$

Let V be an arbitrary full-rank truncation matrix of appropriate dimension. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:

$$\begin{aligned}G_1(\sigma_k) &= \widehat{G}_1(\sigma_k), \\ G_2(\sigma_{k-1}, \sigma_k) &= \widehat{G}_2(\sigma_{k-1}, \sigma_k), \\ &\vdots \\ G_k(\sigma_1, \dots, \sigma_k) &= \widehat{G}_k(\sigma_1, \dots, \sigma_k).\end{aligned}$$

Proof The proof of this theorem follows analogous to the proof of Theorem 1. We only need to note that the left projection space $\text{span}(W)$ involves the $\mathcal{C}(s)$ matrix, which takes always the last argument of G_k into account. Therefore, the order of the interpolation points is reversed and the recursion formula follows the transfer function order going from left to right. The rest follows as in the proof of Theorem 1 by taking the transposed conjugate of the matrix functions for the construction. $\square$

The main difference between Theorem 1 and Theorem 2 is the order in which the interpolation points have to be used. Switching between the two projection schemes leads to a reverse ordering of the interpolation points for the intermediate transfer functions. The last theorem of this section states now the combination of Theorem 1 and Theorem 2 by two-sided projection.

Theorem 3 (Interpolation by two-sided projection) *Let G and $\widehat{G}$ be as in Theorem 1 and let V be constructed as in Theorem 1 for a given set of interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ and W as in Theorem 2 for another set of interpolation points $\varsigma_1, \dots, \varsigma_\theta \in \mathbb{C}$, for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$ and $\mathcal{B}(s)$ are defined and $\widehat{\mathcal{K}}(s)$ is full-rank. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:*

$$\begin{aligned} G_1(\sigma_1) &= \widehat{G}_1(\sigma_1), \\ G_2(\sigma_1, \sigma_2) &= \widehat{G}_2(\sigma_1, \sigma_2), \\ &\vdots \\ G_k(\sigma_1, \dots, \sigma_k) &= \widehat{G}_k(\sigma_1, \dots, \sigma_k), \quad \text{and} \\ G_1(\varsigma_\theta) &= \widehat{G}_1(\varsigma_\theta), \\ G_2(\varsigma_{\theta-1}, \varsigma_\theta) &= \widehat{G}_2(\varsigma_{\theta-1}, \varsigma_\theta), \\ &\vdots \\ G_\theta(\varsigma_1, \dots, \varsigma_\theta) &= \widehat{G}_\theta(\varsigma_1, \dots, \varsigma_\theta), \end{aligned} \quad (17)$$

and, additionally,

$$G_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta) = \widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta), \quad (18)$$

for $1 \leq q \leq k$ and $1 \leq \eta \leq \theta$.

Proof Since the interpolation conditions in (17) follow directly from Theorem 1 and Theorem 2, we only need to prove (18), the mixed interpolation conditions. For q and η as described in the theorem, we obtain

$$\begin{aligned} &\widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta) \\ &= \widehat{\mathcal{C}}(\varsigma_\theta) \widehat{\mathcal{K}}(\varsigma_\theta)^{-1} \left(\prod_{j=1}^{\eta-1} \widehat{\mathcal{N}}(\varsigma_{\theta-j}) \widehat{\mathcal{K}}(\varsigma_{\theta-j})^{-1} \right) \left(\prod_{i=0}^{q-1} \widehat{\mathcal{N}}(\sigma_{q-i}) \widehat{\mathcal{K}}(\sigma_{q-i})^{-1} \right) \widehat{\mathcal{B}}(\sigma_1) \\ &= \widehat{\mathcal{C}}(\varsigma_\theta) \widehat{\mathcal{K}}(\varsigma_\theta)^{-1} \left(\prod_{j=1}^{\eta-1} \widehat{\mathcal{N}}(\varsigma_{\theta-j}) \widehat{\mathcal{K}}(\varsigma_{\theta-j})^{-1} \right) \underbrace{W^H \left(\prod_{i=0}^{q-1} \mathcal{N}(\sigma_{q-i}) \mathcal{K}(\sigma_{q-i})^{-1} \right) \mathcal{B}(\sigma_1)}_{\in \text{span}(V)} \\ &= \underbrace{\mathcal{C}(\varsigma_\theta) \mathcal{K}(\varsigma_\theta)^{-1} \left(\prod_{j=1}^{\eta-1} \mathcal{N}(\varsigma_{\theta-j}) \mathcal{K}(\varsigma_{\theta-j})^{-1} \right)}_{=: h, \quad h^H \in \text{span}(W)} \left(\prod_{i=0}^{q-1} \mathcal{N}(\sigma_{q-i}) \mathcal{K}(\sigma_{q-i})^{-1} \right) \mathcal{B}(\sigma_1) \\ &= G_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta), \end{aligned}$$

where we used the construction of $\text{span}(V)$ in the third and of $\text{span}(W)$ in the fourth lines as denoted, and following the strategy in the proof of Theorem 1. $\square$

It is an important observation that we can interpolate higher subsystem transfer functions by only evaluating lower ones for the construction of the model reduction bases. Following Theorem 3, we can in fact interpolate up to the $(k + \theta)$ -th subsystem transfer function. Also, we recognize that the two-sided projection-based interpolation is able to match $k + \theta + k \cdot \theta$ interpolation conditions at the same time. Those results are similar to the unstructured systems case [20]. The special case of identical sets of interpolation points is discussed in the following section regarding Hermite interpolation.

3.2 Hermite interpolation

As in the linear case, we can use the projection framework to interpolate not only the transfer functions but also their derivatives. In the setting of the multivariate transfer function appearing in bilinear systems, this amounts to partial derivatives with respect to the different frequency arguments. For ease of notation, we introduce an abbreviation for partial derivatives

$$\partial_{s_1^{j_1} \dots s_k^{j_k}} f(z_1, \dots, z_k) := \frac{\partial^{j_1 + \dots + j_k} f}{\partial s_1^{j_1} \dots \partial s_k^{j_k}}(z_1, \dots, z_k),$$

denoting the differentiation of an analytic function $f: \mathbb{C}^k \rightarrow \mathbb{C}^\ell$ with respect to the variables $s_1, \dots, s_k$ and evaluated at $z_1, \dots, z_k \in \mathbb{C}$. Moreover, the Jacobian of f is denoted by

$$\nabla f = [\partial_{s_1} f \ \dots \ \partial_{s_k} f],$$

as the concatenation of all partial derivatives.

The following theorem states a Hermite interpolation result via V only.

Theorem 4 (Hermite interpolation via V) *Let G be a bilinear SISO system, described by (16), and $\hat{G}$ the reduced-order bilinear SISO system constructed by (10). Let $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ be the interpolation points for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$ and $\mathcal{B}(s)$ are analytic and $\hat{\mathcal{K}}(s)$ is full-rank. Construct V using*

$$\begin{aligned} v_{1,j_1} &= \partial_{s^{j_1}} (\mathcal{K}^{-1} \mathcal{B})(\sigma_1), & j_1 &= 0, \dots, \ell_1, \\ v_{2,j_2} &= \partial_{s^{j_2}} \mathcal{K}^{-1}(\sigma_2) \partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1), & j_2 &= 0, \dots, \ell_2, \\ &\vdots \\ v_{k,j_k} &= \partial_{s^{j_k}} \mathcal{K}^{-1}(\sigma_k) \left(\prod_{j=1}^{k-2} \partial_{s^{\ell_{k-j}}} (\mathcal{N} \mathcal{K}^{-1})(\sigma_{k-j}) \right) \\ &\quad \times \partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1), & j_k &= 0, \dots, \ell_k, \\ \text{span}(V) &\supseteq \text{span}([v_{1,0}, \dots, v_{k,\ell_k}]), \end{aligned}$$

and let W be an arbitrary full-rank truncation matrix of appropriate dimension. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:

$$\begin{aligned}\partial_{s_1^{j_1}} G_1(\sigma_1) &= \partial_{s_1^{j_1}} \widehat{G}_1(\sigma_1), & j_1 &= 0, \dots, \ell_1, \\ \partial_{s_1^{\ell_1} s_2^{j_2}} G_2(\sigma_1, \sigma_2) &= \partial_{s_1^{\ell_1} s_2^{j_2}} \widehat{G}_2(\sigma_1, \sigma_2), & j_2 &= 0, \dots, \ell_2, \\ &\vdots \\ \partial_{s_1^{\ell_1} \dots s_{k-1}^{\ell_{k-1}} s_k^{j_k}} G_k(\sigma_1, \dots, \sigma_k) &= \partial_{s_1^{\ell_1} \dots s_{k-1}^{\ell_{k-1}} s_k^{j_k}} \widehat{G}_k(\sigma_1, \dots, \sigma_k), & j_k &= 0, \dots, \ell_k.\end{aligned}$$

Proof First, we note that the case $k = 1$ was already proven in [26] and $\ell_1 = \dots = \ell_k = 0$ corresponds to Theorem 1. For $k = 2$, we start with $j_2 = 0$ to investigate the partial derivative with respect to s_1 involving the bilinear term. Using the product rule, the partial derivative can be written as

$$\partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1) = \sum_{i=0}^{\ell_1} c_i \partial_{s^i} \mathcal{N}(\sigma_1) \partial_{s^{\ell_1-i}} (\mathcal{K}^{-1} \mathcal{B})(\sigma_1),$$

for some appropriate constants $c_i \in \mathbb{R}$, $i = 0, \dots, \ell_1$. Now, we can show

$$\begin{aligned}\partial_{s_1^{\ell_1}} \widehat{G}_2(\sigma_1, \sigma_2) &= \widehat{\mathcal{C}}(\sigma_2) \widehat{\mathcal{K}}(\sigma_2)^{-1} \partial_{s_1^{\ell_1}} (\widehat{\mathcal{N}} \widehat{\mathcal{K}}^{-1} \widehat{\mathcal{B}})(\sigma_1) \\ &= \widehat{\mathcal{C}}(\sigma_2) \widehat{\mathcal{K}}(\sigma_2)^{-1} \left(\sum_{i=0}^{\ell_1} c_i \partial_{s^i} \widehat{\mathcal{N}}(\sigma_1) \partial_{s^{\ell_1-i}} (\widehat{\mathcal{K}}^{-1} \widehat{\mathcal{B}})(\sigma_1) \right) \\ &= \widehat{\mathcal{C}}(\sigma_2) \widehat{\mathcal{K}}(\sigma_2)^{-1} W^H \left(\sum_{i=0}^{\ell_1} c_i \partial_{s^i} \mathcal{N}(\sigma_1) V \partial_{s^{\ell_1-i}} (\widehat{\mathcal{K}}^{-1} \widehat{\mathcal{B}})(\sigma_1) \right) \\ &= \widehat{\mathcal{C}}(\sigma_2) \widehat{\mathcal{K}}(\sigma_2)^{-1} W^H \left(\sum_{i=0}^{\ell_1} c_i \partial_{s^i} \mathcal{N}(\sigma_1) \partial_{s^{\ell_1-i}} (\mathcal{K}^{-1} \mathcal{B})(\sigma_1) \right) \\ &= \widehat{\mathcal{C}}(\sigma_2) \widehat{\mathcal{K}}(\sigma_2)^{-1} W^H \partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1) \\ &= \mathcal{C}(\sigma_2) \underbrace{V (W^H \mathcal{K}(\sigma_2) V)^{-1} W^H \mathcal{K}(\sigma_2)}_{=: P_{v_{2,0}}} \mathcal{K}(\sigma_2)^{-1} \partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1) \\ &= \mathcal{C}(\sigma_2) \mathcal{K}(\sigma_2)^{-1} \partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1) \\ &= \partial_{s_1^{\ell_1}} G_2(\sigma_1, \sigma_2),\end{aligned}$$

where we first used the construction of v_{1,j_1} and then that of $v_{2,0}$ with the projector $P_{v_{2,0}}$ onto $\text{span}(V)$. By induction over j_2 , the results for the case $k = 2$ follow from [26]; and by induction over k and j_k , using the same arguments, the rest of the theorem follows. $\square$

We note the difference between Theorem 1 and Theorem 4 in terms of the subspace construction. While for the previous interpolation results, we are able to recursively construct the next part of the model reduction subspace by using the previous one, this is not possible in Theorem 4 due to the frequency dependence of the

bilinear term $\mathcal{N}(s)$. Also, it follows that for the interpolation of the ℓ -th derivative, $\ell = \ell_1 + \dots + \ell_k$, of the k -th subsystem transfer function G_k in the interpolation points $\sigma_1, \dots, \sigma_k$, the minimal dimension of the projection space $\text{span}(V)$ is given by $\ell + k$.

As before, we can consider the counterpart to Theorem 4. In addition to reversing the order of interpolation points, the order of the derivatives needs to be reverted as well for the Hermite interpolation.

Theorem 5 (Hermite interpolation via W) *Let $G, \widehat{G}$ the original and reduced-order models, respectively, and the interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ be as in Theorem 4. Construct W using*

$$\begin{aligned} w_{1,j_k} &= \partial_{s^{j_k}} (\mathcal{K}^{-H} \mathcal{C}^H)(\sigma_k), & j_k &= 0, \dots, \ell_k, \\ w_{2,j_{k-1}} &= \partial_{s^{j_{k-1}}} (\mathcal{K}^{-H} \mathcal{N}^H)(\sigma_{k-1}) \partial_{s^{\ell_k}} (\mathcal{K}^{-H} \mathcal{C}^H)(\sigma_k), & j_{k-1} &= 0, \dots, \ell_{k-1}, \\ &\vdots \\ w_{k,j_1} &= \partial_{s^{j_1}} (\mathcal{K}^{-H} \mathcal{N}^H)(\sigma_1) \left(\prod_{j=2}^{k-1} \partial_{s^{\ell_j}} (\mathcal{K}^{-H} \mathcal{N}^H)(\sigma_j) \right) \partial_{s^{\ell_k}} \\ &\quad \times (\mathcal{K}^{-H} \mathcal{C}^H)(\sigma_k), & j_1 &= 0, \dots, \ell_1, \\ \text{span}(W) &\supseteq \text{span}([w_{1,0}, \dots, w_{k,\ell_k}]), \end{aligned}$$

and let V be an arbitrary full-rank truncation matrix of appropriate dimension. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:

$$\begin{aligned} \partial_{s_1^{j_k}} G_1(\sigma_k) &= \partial_{s_1^{j_k}} \widehat{G}_1(\sigma_k), & j_k &= 0, \dots, \ell_k, \\ \partial_{s_1^{j_{k-1}} s_2^{\ell_k}} G_2(\sigma_{k-1}, \sigma_k) &= \partial_{s_1^{j_{k-1}} s_2^{\ell_k}} \widehat{G}_2(\sigma_{k-1}, \sigma_k), & j_{k-1} &= 0, \dots, \ell_{k-1}, \\ &\vdots \\ \partial_{s_1^{j_1} s_2^{\ell_2} \dots s_k^{\ell_k}} G_k(\sigma_1, \dots, \sigma_k) &= \partial_{s_1^{j_1} s_2^{\ell_2} \dots s_k^{\ell_k}} \widehat{G}_k(\sigma_1, \dots, \sigma_k), & j_1 &= 0, \dots, \ell_1. \end{aligned}$$

Proof Observing that the order of the derivatives changed in the same way as the interpolation points, the proof works analogously to the proof of Theorem 4 while building on the ideas from the proof of Theorem 2. $\square$

An interesting fact in the structured linear case, as stated in [26], is the implicit matching of Hermite interpolation conditions without sampling the derivatives of the transfer function. Next, we extend this construction to the structured bilinear case. This result becomes a special case of Theorem 3 by using identical sets of interpolation points for V and W .

Theorem 6 (Implicit Hermite interpolation by two-sided projection) *Let G and $\widehat{G}$ be as in Theorem 4. Also let V and W be constructed as in Theorems 1 and 2, respectively, for the same set of interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$, for which the matrix*

functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$ and $\mathcal{B}(s)$ are analytic and $\widehat{\mathcal{K}}(s)$ is full-rank. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:

$$\begin{aligned} G_1(\sigma_1) &= \widehat{G}_1(\sigma_1), \quad \dots, \quad G_{k-1}(\sigma_1, \dots, \sigma_{k-1}) = \widehat{G}_{k-1}(\sigma_1, \dots, \sigma_{k-1}), \\ G_1(\sigma_k) &= \widehat{G}_1(\sigma_k), \quad \dots, \quad G_{k-1}(\sigma_2, \dots, \sigma_k) = \widehat{G}_{k-1}(\sigma_2, \dots, \sigma_k), \end{aligned}$$

and, additionally,

$$\begin{aligned} G_k(\sigma_1, \dots, \sigma_k) &= \widehat{G}_k(\sigma_1, \dots, \sigma_k), \\ \nabla G_k(\sigma_1, \dots, \sigma_k) &= \nabla \widehat{G}_k(\sigma_1, \dots, \sigma_k), \\ G_{q+\eta}(\sigma_1, \dots, \sigma_q, \sigma_{k-\eta+1}, \dots, \sigma_k) &= \widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \sigma_{k-\eta+1}, \dots, \sigma_k), \end{aligned}$$

hold for $1 \leq q, \eta \leq k$.

Proof While most of the results follow directly from Theorem 3 by using identical sets of interpolation points for V and W , the Hermite interpolation of the complete Jacobian ∇G_k of the k -th order transfer function is new. Since $k = 1$ (the linear subsystem) is covered by [26], we assume $k > 1$. Therefore, and by the structure of the multivariate transfer functions G_k , three different cases can occur depending on the differentiation variable, i.e., we have

$$\begin{aligned} \partial_{s_1} : \quad \partial_s(\mathcal{N}\mathcal{K}^{-1}\mathcal{B}) &= (\partial_s\mathcal{N})\mathcal{K}^{-1}\mathcal{B} + \mathcal{N}(\partial_s(\mathcal{K}^{-1}\mathcal{B})), \\ \partial_{s_j} : \quad \partial_s(\mathcal{N}\mathcal{K}^{-1}) &= (\partial_s\mathcal{N})\mathcal{K}^{-1} + \mathcal{N}(\partial_s\mathcal{K}^{-1}), \quad \text{for } 1 < j < k, \\ \partial_{s_k} : \quad \partial_s(\mathcal{C}\mathcal{K}^{-1}) &= (\partial_s\mathcal{C})\mathcal{K}^{-1} + \mathcal{C}(\partial_s\mathcal{K}^{-1}), \end{aligned}$$

as possible derivative terms. Since those three cases work analogously to each other, we restrict ourselves, for the sake of compactness, to the first one. First, we extend the expression of the partial derivative further into

$$\partial_s(\mathcal{N}\mathcal{K}^{-1}\mathcal{B}) = (\partial_s\mathcal{N})\mathcal{K}^{-1}\mathcal{B} + \mathcal{N}(-\mathcal{K}^{-1}(\partial_s\mathcal{K})\mathcal{K}^{-1}\mathcal{B} + \mathcal{K}^{-1}(\partial_s\mathcal{B})).$$

Therefore, for the complete partial derivative, we obtain

$$\begin{aligned} &\partial_{s_1}\widehat{G}_k(\sigma_1, \dots, \sigma_k) \\ &= \widehat{\mathcal{C}}(\sigma_k)\widehat{\mathcal{K}}(\sigma_k)^{-1} \left(\prod_{j=1}^{k-2} \widehat{\mathcal{N}}(\sigma_{k-j})\widehat{\mathcal{K}}(\sigma_{k-j})^{-1} \right) \partial_s(\widehat{\mathcal{N}}\widehat{\mathcal{K}}^{-1}\widehat{\mathcal{B}})(\sigma_1) \\ &= \widehat{\mathcal{C}}(\sigma_k)\widehat{\mathcal{K}}(\sigma_k)^{-1} \left(\prod_{j=1}^{k-2} \widehat{\mathcal{N}}(\sigma_{k-j})\widehat{\mathcal{K}}(\sigma_{k-j})^{-1} \right) \\ &\quad \times \left[(\partial_s\widehat{\mathcal{N}})\widehat{\mathcal{K}}^{-1}\widehat{\mathcal{B}} - \widehat{\mathcal{N}}\widehat{\mathcal{K}}^{-1}(\partial_s\widehat{\mathcal{K}})\widehat{\mathcal{K}}^{-1}\widehat{\mathcal{B}} + \widehat{\mathcal{N}}\widehat{\mathcal{K}}^{-1}(\partial_s\widehat{\mathcal{B}}) \right](\sigma_1) \\ &= \widehat{\mathcal{C}}(\sigma_k)\widehat{\mathcal{K}}(\sigma_k)^{-1} \left(\prod_{j=1}^{k-2} \widehat{\mathcal{N}}(\sigma_{k-j})\widehat{\mathcal{K}}(\sigma_{k-j})^{-1} \right) \partial_s\widehat{\mathcal{N}}(\sigma_1)\widehat{\mathcal{K}}(\sigma_1)^{-1}\widehat{\mathcal{B}}(\sigma_1) \end{aligned}$$

$$\begin{aligned}
& -\widehat{\mathcal{C}}(\sigma_k)\widehat{\mathcal{K}}(\sigma_k)^{-1}\left(\prod_{j=1}^{k-2}\widehat{\mathcal{N}}(\sigma_{k-j})\widehat{\mathcal{K}}(\sigma_{k-j})^{-1}\right)\widehat{\mathcal{N}}(\sigma_1)\widehat{\mathcal{K}}(\sigma_1)^{-1} \\
& \quad \times \partial_s \widehat{\mathcal{K}}(\sigma_1)\widehat{\mathcal{K}}(\sigma_1)^{-1}\widehat{\mathcal{B}}(\sigma_1) \\
& + \widehat{\mathcal{C}}(\sigma_k)\widehat{\mathcal{K}}(\sigma_k)^{-1}\left(\prod_{j=1}^{k-2}\widehat{\mathcal{N}}(\sigma_{k-j})\widehat{\mathcal{K}}(\sigma_{k-j})^{-1}\right)\widehat{\mathcal{N}}(\sigma_1)\widehat{\mathcal{K}}(\sigma_1)^{-1}\partial_s \widehat{\mathcal{B}}(\sigma_1) \\
& = \underbrace{\mathcal{C}(\sigma_k)\mathcal{K}(\sigma_k)^{-1}\left(\prod_{j=1}^{k-2}\mathcal{N}(\sigma_{k-j})\mathcal{K}(\sigma_{k-j})^{-1}\right)}_{=: h_1, h_1^H \in \text{span}(W)} \underbrace{\partial_s \mathcal{N}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1)}_{\in \text{span}(V)} \\
& \quad - \underbrace{\mathcal{C}(\sigma_k)\mathcal{K}(\sigma_k)^{-1}\left(\prod_{j=1}^{k-2}\mathcal{N}(\sigma_{k-j})\mathcal{K}(\sigma_{k-j})^{-1}\right)\mathcal{N}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}}_{=: h_2, h_2^H \in \text{span}(W)} \\
& \quad \times \underbrace{\partial_s \mathcal{K}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}\mathcal{B}(\sigma_1)}_{\in \text{span}(V)} \\
& \quad + \underbrace{\mathcal{C}(\sigma_k)\mathcal{K}(\sigma_k)^{-1}\left(\prod_{j=1}^{k-2}\mathcal{N}(\sigma_{k-j})\mathcal{K}(\sigma_{k-j})^{-1}\right)\mathcal{N}(\sigma_1)\mathcal{K}(\sigma_1)^{-1}\partial_s \mathcal{B}(\sigma_1)}_{=: h_2, h_2^H \in \text{span}(W)} \\
& = \partial_{s_1} G_k(\sigma_1, \dots, \sigma_k),
\end{aligned}$$

where we used, as denoted by the underbraces, the construction of either $\text{span}(W)$ or $\text{span}(V)$, and the fact that the model reduction bases V and W are constant matrices. As stated before, the results for the other partial derivatives follow analogously, which proves interpolation of the full Jacobian in the end. $\square$

As in the previous section, by using two-sided projection, we can match interpolation conditions for a larger number of interpolation points and higher subsystem transfer functions. Following the results of Theorem 3 we can expect, using derivatives for the two-sided projection, to match at least $(k + \ell) + (\theta + \nu) + (k + \ell) \cdot (\theta + \nu)$ transfer function values, where k, ℓ relate to $\text{span}(V)$ and θ, ν to $\text{span}(W)$, and where $\ell = \ell_1 + \dots + \ell_k$ and $\nu = \nu_1 + \dots + \nu_\theta$ denote the orders of the partial derivatives and k, θ the subsystem order of the transfer functions to interpolate.

Theorem 7 (Hermite interpolation by two-sided projection) *Let G and $\widehat{G}$ be as in Theorem 4 and let V be constructed as in Theorem 4 for a given set of interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ and orders of partial derivatives $\ell_1, \dots, \ell_k$, and W as in Theorem 5 for another set of interpolation points $\varsigma_1, \dots, \varsigma_\theta \in \mathbb{C}$ and orders of*

partial derivatives $v_1, \dots, v_\theta$, for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$ and $\mathcal{B}(s)$ are analytic and $\widehat{\mathcal{K}}(s)$ has full-rank. Then the subsystem transfer functions of $\widehat{G}$ interpolate those of G in the following way:

$$\begin{aligned} \partial_{s_1^{j_1}} G_1(\sigma_1) &= \partial_{s_1^{j_1}} \widehat{G}_1(\sigma_1), & j_1 &= 0, \dots, \ell_1, \\ &\vdots \\ \partial_{s_1^{\ell_1} \dots s_{k-1}^{\ell_{k-1}} s_k^{j_k}} G_k(\sigma_1, \dots, \sigma_k) &= \partial_{s_1^{\ell_1} \dots s_{k-1}^{\ell_{k-1}} s_k^{j_k}} \widehat{G}_k(\sigma_1, \dots, \sigma_k), & j_k &= 0, \dots, \ell_k, \\ \partial_{s_1^{i_\theta}} G_1(\varsigma_\theta) &= \partial_{s_1^{i_\theta}} \widehat{G}_1(\varsigma_\theta), & i_\theta &= 0, \dots, v_\theta, \\ &\vdots \\ \partial_{s_1^{i_1} s_2^{v_2} \dots s_\theta^{v_\theta}} G_\theta(\varsigma_1, \dots, \varsigma_\theta) &= \partial_{s_1^{i_1} s_2^{v_2} \dots s_\theta^{v_\theta}} \widehat{G}_\theta(\varsigma_1, \dots, \varsigma_\theta), & i_1 &= 0, \dots, v_1, \end{aligned}$$

and, additionally,

$$\begin{aligned} &\partial_{s_1^{\ell_1} \dots s_{q-1}^{\ell_{q-1}} s_q^{j_q} s_{q+1}^{i_{\theta-\eta+1}} s_{q+2}^{v_{\theta-\eta+2}} \dots s_{q+\eta}^{v_\theta}} G_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta) \\ &= \partial_{s_1^{\ell_1} \dots s_{q-1}^{\ell_{q-1}} s_q^{j_q} s_{q+1}^{i_{\theta-\eta+1}} s_{q+2}^{v_{\theta-\eta+2}} \dots s_{q+\eta}^{v_\theta}} \widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta) \end{aligned}$$

holds for $j_q = 0, \dots, \ell_q$; $i_{\theta-\eta+1} = 0, \dots, v_{\theta-\eta+1}$; $1 \leq q \leq k$ and $1 \leq \eta \leq \theta$.

Proof As for Theorem 3, the first parts of the result just summarize the theorems stating the one-sided projection approaches (Theorems 4 and 5), i.e., we only need to prove the additional interpolation constraints with the mixed partial derivatives. It holds

$$\begin{aligned} &\partial_{s_1^{\ell_1} \dots s_{q-1}^{\ell_{q-1}} s_q^{j_q} s_{q+1}^{i_{\theta-\eta+1}} s_{q+2}^{v_{\theta-\eta+2}} \dots s_{q+\eta}^{v_\theta}} G_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta) \\ &= \partial_{s^{v_\theta}} (\widehat{\mathcal{C}} \widehat{\mathcal{K}}^{-1})(\varsigma_\theta) \dots \partial_{s^{v_{\theta-\eta+2}}} (\widehat{\mathcal{N}} \widehat{\mathcal{K}}^{-1})(\varsigma_{\theta-\eta+2}) \partial_{s^{i_{\theta-\eta+1}}} (\widehat{\mathcal{N}} \widehat{\mathcal{K}}^{-1})(\varsigma_{\theta-\eta+1}) \\ &\quad \times \partial_{s^{j_q}} (\widehat{\mathcal{N}} \widehat{\mathcal{K}}^{-1})(\sigma_q) \partial_{s^{\ell_{q-1}}} (\widehat{\mathcal{N}} \widehat{\mathcal{K}}^{-1})(\sigma_{q-1}) \dots \partial_{s^{\ell_1}} (\widehat{\mathcal{N}} \widehat{\mathcal{K}}^{-1} \widehat{\mathcal{B}})(\sigma_1) \\ &= \underbrace{\partial_{s^{v_\theta}} (\mathcal{C} \mathcal{K}^{-1})(\varsigma_\theta) \dots \partial_{s^{v_{\theta-\eta+2}}} (\mathcal{N} \mathcal{K}^{-1})(\varsigma_{\theta-\eta+2}) \partial_{s^{i_{\theta-\eta+1}}} (\mathcal{N} \mathcal{K}^{-1})(\varsigma_{\theta-\eta+1})}_{=: h, h \in \text{span}(W)} \\ &\quad \times \underbrace{\partial_{s^{j_q}} (\mathcal{N} \mathcal{K}^{-1})(\sigma_q) \partial_{s^{\ell_{q-1}}} (\mathcal{N} \mathcal{K}^{-1})(\sigma_{q-1}) \dots \partial_{s^{\ell_1}} (\mathcal{N} \mathcal{K}^{-1} \mathcal{B})(\sigma_1)}_{\in \text{span}(V)} \\ &= \partial_{s_1^{\ell_1} \dots s_{q-1}^{\ell_{q-1}} s_q^{j_q} s_{q+1}^{i_{\theta-\eta+1}} s_{q+2}^{v_{\theta-\eta+2}} \dots s_{q+\eta}^{v_\theta}} \widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta), \end{aligned}$$

for $j_q = 0, \dots, \ell_q$; $i_{\theta-\eta+1} = 0, \dots, v_{\theta-\eta+1}$; $1 \leq q \leq k$ and $1 \leq \eta \leq \theta$. $\square$

For an easier understanding of Theorem 7, we consider here a small theoretical example, where we only interpolate the linear part choosing $k = \theta = 1$, the interpolation points σ, ς , and for the partial derivatives $\ell = \ell_1 = 2$ and $v = v_1 = 1$. Then

using the first part of Theorem 7, we enforce interpolation of the following terms by means of $\text{span}(V)$:

$$G_1(\sigma), \quad \partial_{s_1} G_1(\sigma), \quad \partial_{s_1^2} G_1(\sigma),$$

And similarly via $\text{span}(W)$, we enforce interpolation of

$$G_1(\varsigma), \quad \partial_{s_1} G_1(\varsigma).$$

By using two-sided projection, we can now additionally match higher subsystem transfer functions and their partial derivatives, namely

$$\begin{aligned} G_2(\sigma, \varsigma), \quad \partial_{s_1} G_2(\sigma, \varsigma), \quad \partial_{s_2} G_2(\sigma, \varsigma), \\ \partial_{s_1^2} G_2(\sigma, \varsigma), \quad \partial_{s_1 s_2} G_2(\sigma, \varsigma), \quad \partial_{s_1^2 s_2} G_2(\sigma, \varsigma). \end{aligned}$$

As already realized in Theorem 6, two-sided projection with the same sets of interpolation points leads to additional interpolation of derivatives. This also works in combination with Theorem 7. The following corollary states a particular special case.

Corollary 1 Assume G and $\widehat{G}$ are constructed as in Theorem 7 for identical sets of interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ and matching orders of the partial derivatives, i.e., $\ell_1 = \nu_1, \dots, \ell_k = \nu_k$. Then additionally to the interpolation results of Theorem 7 it holds

$$\nabla \left(\partial_{s_1^{\ell_1} \dots s_k^{\ell_k}} G_k \right) (\sigma_1, \dots, \sigma_k) = \nabla \left(\partial_{s_1^{\ell_1} \dots s_k^{\ell_k}} \widehat{G}_k \right) (\sigma_1, \dots, \sigma_k).$$

Proof The proof follows directly from Theorem 6 by setting the last partial derivative as the final interpolation condition of the left and right projection spaces. $\square$

3.3 Heuristics for selecting interpolation points

A crucial step in the success of interpolatory methods is clearly the selection of good/optimal interpolation points. Choosing optimal interpolation points is an open problem even in the case of structure-preserving interpolation for linear systems. For the numerical experiments in Section 3.4, based on some successful heuristics from the linear case, we suggest and compare three strategies for choosing the interpolation points in the proposed framework of structure-preserving bilinear transfer function interpolation. Below we summarize these three approaches.

First is the classical choice of taking logarithmically equidistant points on the imaginary axis in a frequency range of interest. The major advantage of this idea is its simplicity, as the points do not need any further methods to be computed but are directly given via the frequency interval and the chosen reduced order, e.g., in MATLAB, they can be simply generated via the `logspace` function. Often, this approach catches a good overall behavior of the system but misses specific features that are not close enough to the interpolation points. In principle, it is possible to choose a new set of interpolation points for every subsystem transfer function. Via combining different sets, this leads to an exponentially growing number of interpolation conditions

to be met, and thus exponentially growing subspace dimensions. Instead, to have a better control on the resulting reduced order, we suggest to use the same interpolation points on each transfer function level, i.e., for given points $\sigma^{(1)}, \dots, \sigma^{(\ell)} \in \mathbb{C}$ with interpolation up the k -th subsystem transfer function, the interpolation point sequences are set to be

$$\begin{aligned} \sigma^{(1)} &= \sigma_1^{(1)} = \dots = \sigma_k^{(1)}, \\ &\vdots \\ \sigma^{(\ell)} &= \sigma_1^{(\ell)} = \dots = \sigma_k^{(\ell)}, \end{aligned} \quad (19)$$

such that the following transfer function values are interpolated:

$$\begin{aligned} G_1(\sigma^{(1)}), \quad G_2(\sigma^{(1)}, \sigma^{(1)}), \quad \dots \quad G_k(\sigma^{(1)}, \dots, \sigma^{(1)}), \\ \vdots \\ G_1(\sigma^{(\ell)}), \quad G_2(\sigma^{(\ell)}, \sigma^{(\ell)}), \quad \dots \quad G_3(\sigma^{(\ell)}, \dots, \sigma^{(\ell)}); \end{aligned}$$

see, e.g., [16–19].

The second strategy we will employ relates to the $\mathcal{H}_2$ -norm of dynamical systems. The Iterative Rational Krylov Algorithm (IRKA) can be used in the case of linear (unstructured) first-order systems to compute a locally optimal approximation in the $\mathcal{H}_2$ -norm via interpolation [31]. For unstructured quadratic-bilinear systems, using IRKA based on the linear dynamics (first subsystem) to generate optimal interpolation points and then using the resulting interpolation points as in (19) for the subsystem interpolation was already applied successfully in [32]. In our setting, this would amount to applying IRKA first to a structured linear dynamical system. However, there is no known general structure-preserving variant of IRKA compliant with the setting in [26], except for some extensions to special cases [33–38]. But in [39], the $\mathcal{H}_2$ -optimal interpolation theory for unstructured systems was extended to general linear transfer functions via Transfer Function IRKA (TF-IRKA) by allowing the reduced model to be unstructured. The basic idea is, in every iteration step, to evaluate the given transfer function and its first derivative at the current interpolation points and then to use the Loewner framework [40] to construct an interpolating unstructured linear system. As extension to our structured bilinear systems framework, we will employ TF-IRKA to compute optimal interpolation points for the structured linear subsystem and then use the resulting points for the subsystem interpolation as in (19).

As the third and final strategy, we will consider a greedy selection of interpolation points based on the $\mathcal{H}_\infty$ -norm. Here, we will make use of the ideas developed in [41, 42] to use the large-scale sparse $\mathcal{H}_\infty$ -norm computation methods from [43–45]. The main idea is to choose the next interpolation point, during the iteration, as the frequency where the $\mathcal{H}_\infty$ -norm of the error system is attained, i.e., the frequency for which the approximation error attains the maximum. Here, we further extend this approach to multivariate transfer functions of structured bilinear systems based on the following observation: To interpolate a higher-level transfer function, the vectors from previous levels need to be included in the projection space. Therefore, assume

we have just computed the vectors up to level $k - 1$ with v_{k-1} for the right truncation matrix V corresponding to the interpolation points $\sigma_1, \dots, \sigma_{k-1}$. To interpolate now the k -th subsystem transfer function, we need to employ those previous vectors and observe that the transfer function satisfies

$$G_k(\sigma_1, \dots, \sigma_{k-1}, s_k) = \mathcal{C}(s_k)\mathcal{K}(s_k)\mathcal{N}(\sigma_{k-1})v_{k-1}.$$

This is again a univariate transfer function in the k -th frequency argument as the previous interpolation points are fixed, where the input function is just the constant vector $\mathcal{N}(\sigma_{k-1})v_{k-1}$. Therefore, we can do a greedy search for the current subsystem transfer function, interpolate in the resulting point, update the error system, and repeat this procedure with the next higher subsystem transfer function. This is precisely what we will do. In the literature, there are other approaches for greedy selections, e.g., by using error estimators [46, 47]. However, employing these ideas will first require developing appropriate estimators for structured bilinear systems. This remains an open problem to be investigated in future work.

3.4 Numerical examples

We illustrate the SISO analysis and the strategies for the interpolation point selection using two numerical examples, having the structured bilinearities as in Sections 2.3.1 and 2.3.2. We compare our resulting structure-preserving interpolation framework to other approaches from the literature that have been used to approximate structured bilinear systems *without preserving the structure*, as in, e.g., [3, 25].

We will compute reduced-order models using the three interpolation point selection strategies from Section 3.3, where we will denote the equidistant points as *equi.*, the IRKA-based points by *IRKA*, and the greedy selection by $\mathcal{H}_\infty$. We compare the approximation error both in time and frequency domains. In time domain, we consider a point-wise relative output error for a given input signal, namely

$$\text{err}_t(t) := \frac{|y(t) - \hat{y}(t)|}{|y(t)|},$$

for $t \in [0, t_f]$, and in frequency domain, we use the point-wise relative error of the first and second subsystem transfer functions, i.e.,

$$\begin{aligned} \text{err}_{f,1}(\omega_1) &:= \frac{|G_1(\omega_1 i) - \widehat{G}_1(\omega_1 i)|}{|G_1(\omega_1 i)|} && \text{and} \\ \text{err}_{f,2}(\omega_1, \omega_2) &:= \frac{|G_2(\omega_1 i, \omega_2 i) - \widehat{G}_2(\omega_1 i, \omega_2 i)|}{|G_2(\omega_1 i, \omega_2 i)|}, \end{aligned}$$

for the frequencies $\omega_1, \omega_2 \in [\omega_{\min}, \omega_{\max}]$.

The experiments reported here have been executed on a machine with 2 Intel(R) Xeon(R) Silver 4110 CPU processors running at 2.10 GHz and equipped with 192 GB total main memory. The computer runs on CentOS Linux release 7.5.1804 (Core) using MATLAB 9.7.0.1190202 (R2019b).

3.4.1 Time-delayed heated rod

First, we consider an example for a bilinear time-delay system as in Section 1. This example, taken from [25], models a semi-discretized heated rod with distributed control and homogeneous Dirichlet boundary conditions, which is cooled by a delayed feedback and is described by the partial differential equation

$$\partial_t v(\zeta, t) = \partial_{\zeta^2} v(\zeta, t) - 2 \sin(\zeta) v(\zeta, t) + 2 \sin(\zeta) v(\zeta, t - 1) + u(t),$$

with $(\zeta, t) \in (0, \pi) \times (0, t_f)$ and boundary conditions $v(0, t) = v(\pi, t) = 0$ for $t \in [0, t_f]$. After a spatial discretization using centered finite differences, we obtain a bilinear time-delay system of the form

$$\begin{aligned} \dot{x}(t) &= Ax(t) + A_d x(t - 1) + Nx(t)u(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned}$$

with $A, A_d, N \in \mathbb{R}^{n \times n}$, $B, C^T \in \mathbb{R}^n$. We chose $n = 5000$ for our experiments.

To compare with our structure-preserving approximation (StrInt), in this example, we use the approach from [25] to construct an unstructured bilinear system (2) without time delay using the bilinear Loewner framework, denoted by BiLoewner. For the structured interpolation, we have computed 4 interpolation points for the interpolation of the first two subsystem transfer functions with each strategy. For the equidistant points, we used $\pm \logspace(-3, 3, 2)i$. The resulting interpolation points for the first subsystem transfer function using the three different strategies can be seen in Fig. 1. Note that since the interpolation points appear as conjugate pairs in the subspace construction, in Fig. 1, we only depict the top half of the complex plane. While the points generated by equi. and $\mathcal{H}_\infty$ appear very similar, the IRKA-based points are distinguishably different and even involve 2 real interpolation points. The interpolation points $\pm 10^{-3}i$ are obtained by equi. as well as $\mathcal{H}_\infty$ due to the focus on the frequency range of interest to $[10^{-3}, 10^3]$ rad/sec. The structured interpolations were then obtained via the two-sided projection approach from Theorem 3. The resulting reduced-order bilinear time-delay systems are of order $r = 8$. For the bilinear Loewner method, we chose the interpolation points $\pm \logspace(-3, 3, 80)i$ and used the rank truncation idea to obtain a classical (unstructured) bilinear system, also of order 8.



Fig. 1 Computed interpolation points for structured interpolation of the first subsystem transfer function for the time-delay system (points appear in conjugates pairs, only top half of the complex plane is depicted)

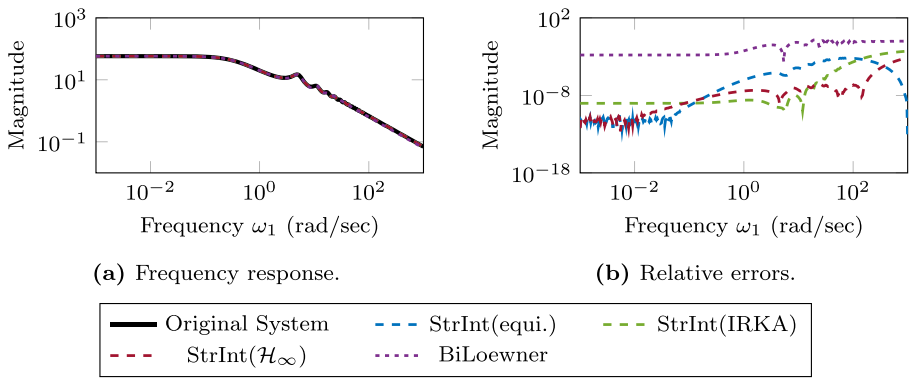


Fig. 2 Frequency domain results of the first subsystem transfer functions for the time-delay system

First, we consider our computed approximations in frequency domain. Those results are plotted in Figs. 2 and 3 for the first and second subsystem transfer functions. It is clearly visible that the structure-preserving approximation of any

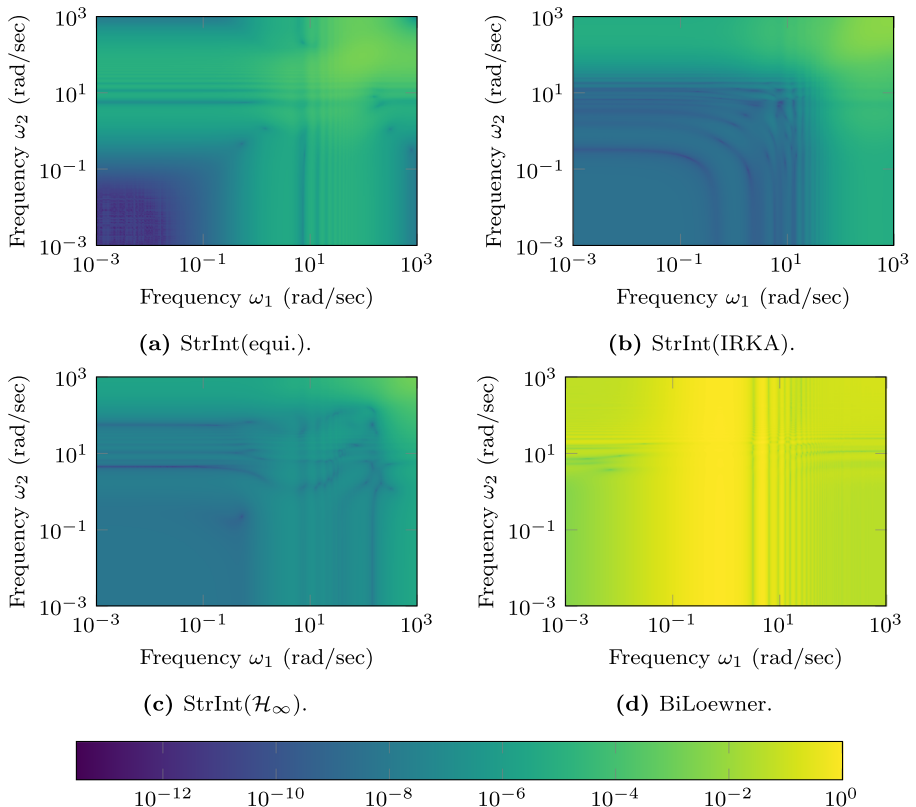


Fig. 3 Relative errors of the second subsystem transfer functions for the time-delay system

Table 1 Maximum relative approximation errors for the time-delay system

	StrInt(equi.)	StrInt(IRKA)	StrInt($\mathcal{H}_\infty$)	BiLoewner
$\max(\text{err}_{f,1})$	6.5768e-04	5.1859e-03	7.1636e-04	1.4683e-01
$\max(\text{err}_{f,2})$	1.7612e-03	4.8620e-03	1.3815e-03	1.0530e+00
$\max(\text{err}_t)$	8.1166e-03	2.1001e-05	1.2843e-05	2.5705e+00

interpolation point selection strategy performs significantly better than the unstructured bilinear Loewner framework. This is due to the time-delay structure being preserved in the reduced-order models. Concerning the selection of interpolation points, no clear winner can be deduced from the plots. Looking at the maximum relative approximation errors in Table 1, for the first subsystem transfer function, the equidistant and greedy $\mathcal{H}_\infty$ methods perform best, while for the second-level transfer function, the three methods behave similarly.

For a complete comparison of the reduced-order models, we simulated the systems in time domain with the input signal

$$u(t) = \frac{\cos(10t)}{20} + \frac{\cos(5t)}{20},$$

from [25], in the time range $[0, 10]$ s. Figure 4 shows that (a) the output trajectories of the original system, the structure-preserving interpolations and the bilinear system without time-delay are indistinguishable in the eye ball norm but (b) the relative error reveals that all structured interpolations perform several orders of magnitude better than BiLoewner while having the same state-space dimension. A look into Table 1 reveals that the interpolation methods perform at least three orders of magnitude better than Loewner and that this time, IRKA and $\mathcal{H}_\infty$ points perform best while gaining two orders of magnitude in accuracy compared to the equidistant points.

3.4.2 Damped mass-spring system

Next, we consider a damped mass-spring system. The linear parts of the dynamics are modeled as in [48], describing a chain of masses connected by springs and

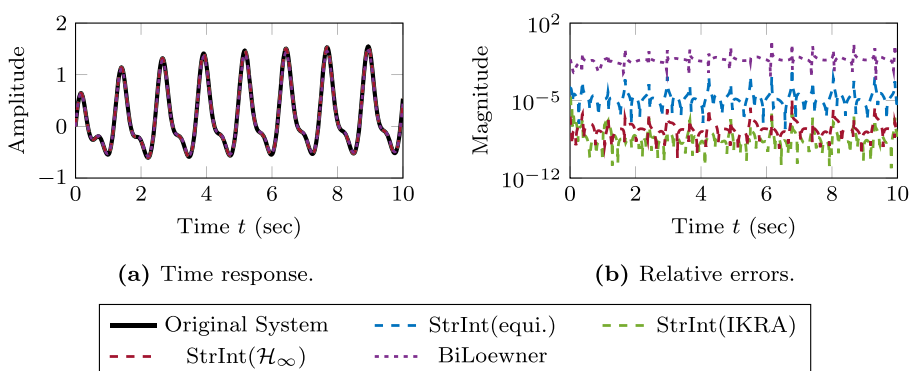


Fig. 4 Time simulation results for the time-delay system

dampers, where each mass is additionally connected to a separate spring and damper. In order to focus on the mechanical structure only, we removed the holonomic constraint from [48]. For the bilinear part, the springs are modeled to be dependent on the applied external force, such that a displacement to the right increases the stiffness due to compression of the springs and to the left decreases it due to the appearing strain. This results in a structured bilinear control system of the form

$$\begin{aligned} M\ddot{q}(t) + D\dot{q}(t) + Kq(t) &= N_p q(t)u(t) + B_u u(t), \\ y(t) &= C_p q(t), \end{aligned} \quad (20)$$

with $M, D, K, N_p \in \mathbb{R}^{n \times n}$ and $B_u, C_p^T \in \mathbb{R}^n$. The input matrix is chosen to apply the external force only to the first mass, i.e., $B = e_1$, and the output gives the displacement of the second mass, i.e., $C = e_2^T$, where e_i denotes the i -th column of the identity matrix I_n . The bilinear term is a scaled version of the stiffness matrix

$$N_p = -SKS,$$

where S is a diagonal matrix with diagonal entries `linspace(0.2, 0, n)`. For our experiment, we chose the original system to consist of $n = 1\,000$ masses.

We construct three types of reduced-order models: (i) our structure-preserving bilinear interpolation, denoted by StrInt, (ii) two unstructured classical bilinear approximations by converting (20) to first-order form (13) followed by interpolatory model reduction of this first-order system, denoted by FOInt, which has the same order as StrInt, and FOInt2, which will be twice as large as StrInt. Note that FOInt and FOInt2 yield reduced-order models of the form (2), which do not retain the underlying physical structure. Also, it needs to be remarked that the computational effort for the construction of FOInt is higher than for the structure-preserving approach due to solving underlying linear systems of double size, even in a structure exploiting implementation; see, e.g., [49]. Since the original system is a mechanical model, we use only a one-sided projection to preserve the mechanical properties in the reduced-order model, i.e., we apply Theorem 1 and set $W = V$. As before, we interpolate the first two subsystem transfer functions via all three strategies for choosing interpolation points from Section 3.3, where the equidistant points for StrInt and FOInt were $\pm \text{logspace}(-2, 2, 3)i$ and for FOInt2 they were $\pm \text{logspace}(-2, 2, 6)i$. The interpolation points for the structure-preserving approach are shown in Fig. 5, where their complex conjugates are omitted for simplicity. The $\mathcal{H}_\infty$ points are clustered around 10^{-1} rad/sec and the IRKA points have basically the same imaginary part and only change in their real parts. The reduced-order models denoted by StrInt and FOInt are by construction of order $r = 12$, and the ones for FOInt2 are of order $r = 24$.

We consider first the results in the frequency domain, where the maximum relative approximation errors for the first subsystem transfer function can be found in Table 2 and for the second subsystem transfer function in Table 3. In general, the unstructured interpolation, FOInt, performs worst for all interpolation point choices but especially bad for IRKA and $\mathcal{H}_\infty$ points. The unstructured interpolation of double size, FOInt2, is compatible with the structured interpolation. The IRKA interpolation points are the exception, for which the structured interpolation performs three orders of magnitude better than FOInt2. Also, comparing only the structured results, one can see that

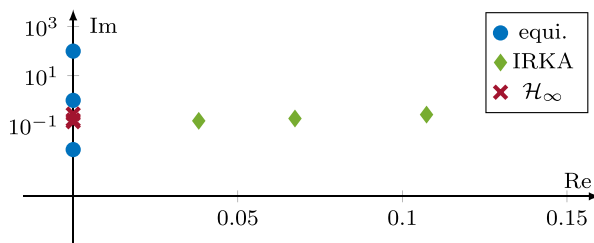


Fig. 5 Computed interpolation points for structured interpolation of the first subsystem transfer function for the damped mass-spring system (points appear in conjugates pairs, only top half of the complex plane is depicted)

the IRKA points perform overall best, while the $\mathcal{H}_\infty$ points perform worst. For an easier overview, we decided to plot only the frequency domain results of the three approaches using the IRKA points in Figs. 6 and 7.

As for the previous example, for a proper comparison of the complete bilinear systems, we simulated the original and reduced-order models in the time interval $[0, 100]$ s using the input signal

$$u(t) = \sin(200t) + 200,$$

which can be seen as a step signal with a sinusoidal disturbance. The worst case relative approximation errors can be found in Table 4. First, note that for some choices of interpolation points, the unstructured reduced-order model outputs grow unbounded in the time simulation due to instability. This is the case for FOInt(equi.), FOInt($\mathcal{H}_\infty$), and FOInt2($\mathcal{H}_\infty$). On the other hand, all structure-preserving reduced-order model outputs are bounded for the chosen input signal, i.e., behave as stable models. Best performing are, ordered by their worst case errors, StrInt(IRKA), StrInt($\mathcal{H}_\infty$), and FOInt2(IRKA), whereas in the frequency domain, the structure-preserving method with IRKA points is the best. But also StrInt(equi.) is compatible as it is only one order of magnitude worse than the best reduced-order model. For this choice of points, only FOInt2 behaves stable and is another order of magnitude worse than the structure-preserving variant. Also note that the worst structure-preserving reduced-order model still performs one order of magnitude better than the only stable FOInt. As for the frequency domain results, we decided to plot only the time domain results using the IRKA interpolation points in Fig. 8.

Table 2 Maximum relative approximation errors of the first subsystem transfer functions $\max(\text{err}_{f,1})$ for the damped mass-spring system

	StrInt	FOInt	FOInt2
equi.	1.3852e-05	4.3641e-03	1.2199e-05
IRKA	2.3105e-06	2.7126e+01	1.4166e-03
$\mathcal{H}_\infty$	5.7449e-03	7.2879e+02	2.6656e-02

Table 3 Maximum relative approximation errors of the second subsystem transfer functions $\max(\text{err}_{f,2})$ for the damped mass-spring system

	StrInt	FOInt	FOInt2
equi.	1.6742e-04	2.8235e-02	1.3956e-04
IRKA	3.4370e-06	8.6974e+01	1.3253e-03
$\mathcal{H}_\infty$	4.3940e-03	9.0163e+02	4.4294e-02

4 Interpolation of multi-input multi-output systems

In this section, we will generalize the results from SISO structured bilinear systems to MIMO ones as in (9) and give a numerical example to illustrate the theory.

4.1 Matrix interpolation

In principle, all the results from Section 3 can directly be extended to the MIMO system case (9). However, one needs to realize that in this case the subsystem transfer functions to be interpolated are matrix-valued. The main difference from the SISO case lies in the collection of the bilinear terms into $\mathcal{N}(s) = [\mathcal{N}_1(s) \dots \mathcal{N}_m(s)]$ and the corresponding Kronecker products that produce the different combinations of the linear and bilinear parts in the k -th subsystem transfer function, e.g., in (5). Additionally, we will use the following notation:

$$\tilde{\mathcal{N}}(s) := \begin{bmatrix} \mathcal{N}_1(s) \\ \vdots \\ \mathcal{N}_m(s) \end{bmatrix}$$

as an alternative way of concatenating the bilinear terms. In this paper, we will only focus on matrix interpolation, i.e., we will interpolate the full matrix-valued structured subsystem transfer functions. There is a concept of tangential interpolation [20, 50] to handle matrix-valued functions in which interpolation is enforced only in

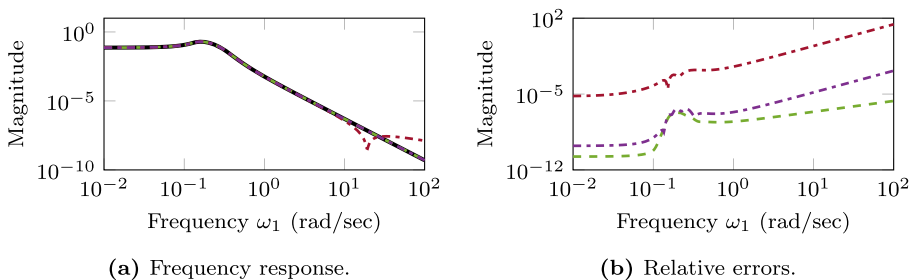


Fig. 6 Frequency domain results of the first subsystem transfer functions for the damped mass-spring system

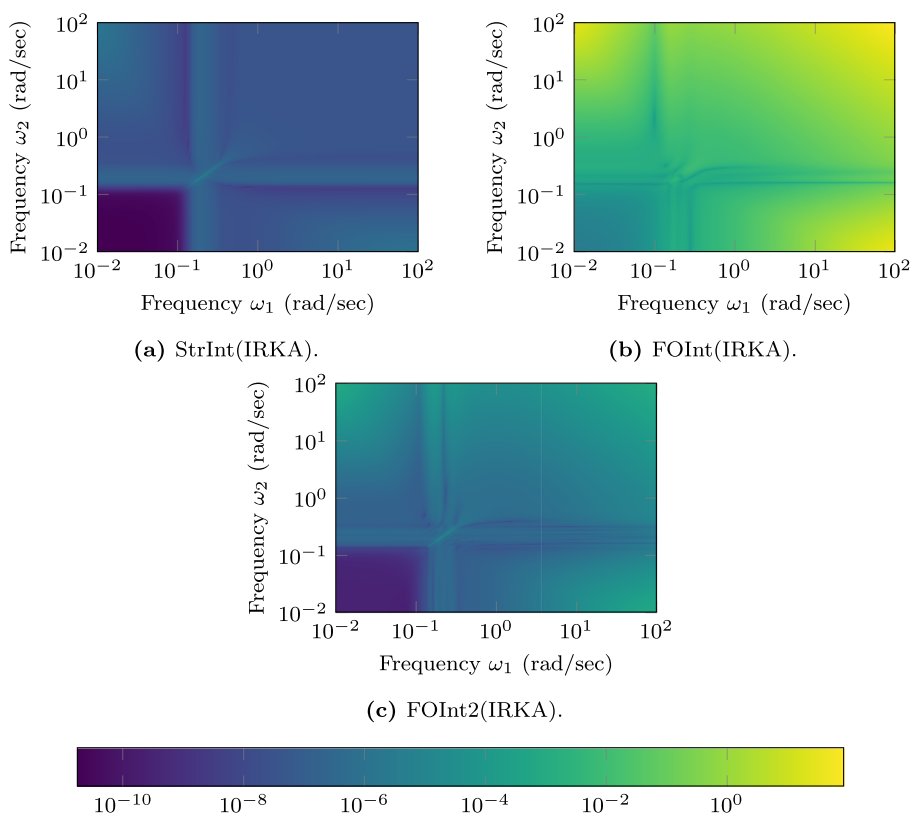


Fig. 7 Relative errors of the second subsystem transfer functions for the damped mass-spring system

selected directions. We will consider that framework in a separate work since the definition of tangential interpolation is not unified yet for bilinear systems [51, 52], let alone the structured ones we consider here.

The following theorem extends the results from Theorems 1, 2, and 3 to MIMO structured bilinear systems.

Theorem 8 (Matrix interpolation) *Let G be a bilinear system, as described by (9), and $\hat{G}$ the reduced-order bilinear system, constructed by (10). Given sets of*

Table 4 Maximum relative approximation errors of the time simulation $\max(\text{err}_t)$ for the damped mass-spring system

	StrInt	FOInt	FOInt2
equi.	1.4916e-03	2.6223e+06	6.7979e-02
IRKA	1.6791e-04	5.4102e-02	5.6114e-04
$\mathcal{H}_\infty$	3.9867e-04	8.2374e+04	1.0453e+13

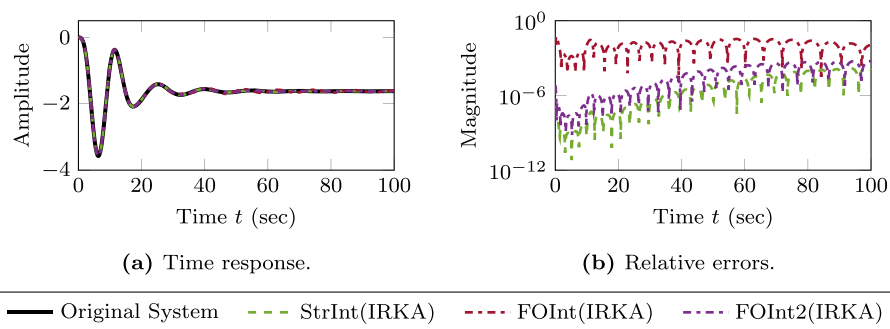


Fig. 8 Time simulation results for the damped mass-spring system

interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ and $\varsigma_1, \dots, \varsigma_\theta \in \mathbb{C}$, for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$, $\mathcal{B}(s)$ are defined and $\widehat{\mathcal{K}}(s)$ is full-rank, the following statements hold:

(a) If V is constructed as

$$\begin{aligned} V_1 &= \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1), \\ V_j &= \mathcal{K}(\sigma_j)^{-1} \mathcal{N}(\sigma_{j-1}) (I_m \otimes V_{j-1}), \quad 2 \leq j \leq k, \\ \text{span}(V) &\supseteq \text{span}([V_1, \dots, V_k]), \end{aligned}$$

then the following interpolation conditions hold true:

$$\begin{aligned} G_1(\sigma_1) &= \widehat{G}_1(\sigma_1), \\ G_2(\sigma_1, \sigma_2) &= \widehat{G}_2(\sigma_1, \sigma_2), \\ &\vdots \\ G_k(\sigma_1, \dots, \sigma_k) &= \widehat{G}_k(\sigma_1, \dots, \sigma_k). \end{aligned}$$

(b) If W is constructed as

$$\begin{aligned} W_1 &= \mathcal{K}(\varsigma_\theta)^{-H} \mathcal{C}(\varsigma_\theta)^H, \\ W_i &= \mathcal{K}(\varsigma_{\theta-i+1})^{-H} \widetilde{\mathcal{N}}(\varsigma_{k-i+1})^H (I_m \otimes W_{i-1}), \quad 2 \leq i \leq \theta, \\ \text{span}(W) &\supseteq \text{span}([W_1, \dots, W_\theta]), \end{aligned}$$

then the following interpolation conditions hold true:

$$\begin{aligned} G_1(\varsigma_\theta) &= \widehat{G}_1(\varsigma_\theta), \\ G_2(\varsigma_{\theta-1}, \varsigma_\theta) &= \widehat{G}_2(\varsigma_{\theta-1}, \varsigma_\theta), \\ &\vdots \\ G_\theta(\varsigma_1, \dots, \varsigma_\theta) &= \widehat{G}_\theta(\varsigma_1, \dots, \varsigma_\theta). \end{aligned}$$

- (c) Let V be constructed as in part (a) and W as in (b), then, additionally to the results in (a) and (b), the interpolation conditions

$$G_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta) = \widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \varsigma_{\theta-\eta+1}, \dots, \varsigma_\theta),$$

hold for $1 \leq q \leq k$ and $1 \leq \eta \leq \theta$.

Proof Starting with part (a), we remember that the transfer functions can be rewritten by multiplying out the Kronecker products as

$$\begin{aligned} G_k(\sigma_1, \dots, \sigma_k) &= [\mathcal{C}(\sigma_k) \mathcal{K}(\sigma_k)^{-1} \mathcal{N}_1(\sigma_{k-1}) \cdots \mathcal{N}_1(\sigma_1) \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1)^{-1}, \\ &\quad \mathcal{C}(\sigma_k) \mathcal{K}(\sigma_k)^{-1} \mathcal{N}_1(\sigma_{k-1}) \cdots \mathcal{N}_2(\sigma_1) \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1)^{-1}, \\ &\quad \dots \\ &\quad \mathcal{C}(\sigma_k) \mathcal{K}(\sigma_k)^{-1} \mathcal{N}_m(\sigma_{k-1}) \cdots \mathcal{N}_m(\sigma_1) \mathcal{K}(\sigma_1)^{-1} \mathcal{B}(\sigma_1)^{-1}]. \end{aligned}$$

From the construction of V , it follows that applying Theorem 1 for the transfer functions in each single entry gives the result. Part (b) directly follows from part (a) by replacing the matrix functions by their Hermitian conjugate versions except for $\mathcal{N}(s) = [\mathcal{N}_1 \dots \mathcal{N}_m]$, where the single entries have to be transposed conjugated. Therefore, the differently stacked $\widetilde{\mathcal{N}}(s)$ is used here to give $\widetilde{\mathcal{N}}(s)^H = [\mathcal{N}_1(s)^H \dots \mathcal{N}_m(s)^H]$. Finally, part (c) follows directly from part (a), (b) and Theorem 3 for the single transfer function entries. $\square$

For Hermite interpolation as in Theorems 4, 5, and 7, a similar extension to the MIMO case follows.

Theorem 9 (Hermite matrix interpolation) *Let G be a bilinear system, described by (9), and $\widehat{G}$ the reduced-order bilinear system, constructed by (10). Given sets of interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$ and $\varsigma_1, \dots, \varsigma_\theta \in \mathbb{C}$, for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$, $\mathcal{B}(s)$ are analytic and $\widehat{\mathcal{K}}(s)$ is full-rank, the following statements hold:*

- (a) If V is constructed as

$$\begin{aligned} V_{1,j_1} &= \partial_{s^{j_1}} (\mathcal{K}^{-1} \mathcal{B})(\sigma_1), & j_1 &= 0, \dots, \ell_1, \\ V_{2,j_2} &= \partial_{s^{j_2}} \mathcal{K}^{-1}(\sigma_2) \partial_{s^{\ell_1}} (\mathcal{N}(I_m \otimes \mathcal{K}^{-1} \mathcal{B}))(\sigma_1), & j_2 &= 0, \dots, \ell_2, \\ &\vdots \\ V_{k,j_k} &= \partial_{s^{j_k}} \mathcal{K}^{-1}(\sigma_k) \\ &\quad \times \left(\prod_{j=1}^{k-2} \partial_{s^{\ell_{k-j}}} ((I_{m^{j-1}} \otimes \mathcal{N})(I_{m^j} \otimes \mathcal{K}))(\sigma_{k-j}) \right) \\ &\quad \times \partial_{s^{\ell_1}} ((I_{m^{k-2}} \otimes \mathcal{N})(I_{m^{k-1}} \otimes \mathcal{K})(I_{m^{k-1}} \otimes \mathcal{B}))(\sigma_1), & j_k &= 0, \dots, \ell_k, \\ \text{span}(V) &\supseteq \text{span}([V_{1,0}, \dots, V_{k,\ell_k}]), \end{aligned}$$

then the following interpolation conditions hold true:

$$\begin{aligned} \partial_{s_1^{j_1}} G_1(\sigma_1) &= \partial_{s_1^{j_1}} \widehat{G}_1(\sigma_1), & j_1 &= 0, \dots, \ell_1, \\ &\vdots \\ \partial_{s_1^{\ell_1} \dots s_{k-1}^{\ell_{k-1}} s_k^{j_k}} G_k(\sigma_1, \dots, \sigma_k) &= \partial_{s_1^{\ell_1} \dots s_{k-1}^{\ell_{k-1}} s_k^{j_k}} \widehat{G}_k(\sigma_1, \dots, \sigma_k), & j_k &= 0, \dots, \ell_k. \end{aligned}$$

(b) If W is constructed as

$$\begin{aligned} W_{1,i_\theta} &= \partial_{s^{i_\theta}} (\mathcal{K}^{-H} \mathcal{C}^H)(\zeta_\theta), & i_\theta &= 0, \dots, v_\theta, \\ W_{2,i_{\theta-1}} &= \partial_{s^{i_{\theta-1}}} (\mathcal{K}^{-H} \widetilde{\mathcal{N}}^H)(\zeta_{\theta-1}) \\ &\quad \times (I_m \otimes \partial_{s^{v_\theta}} (\mathcal{K}^{-H} \mathcal{C}^H)(\zeta_\theta)), & i_{\theta-1} &= 0, \dots, v_{\theta-1}, \\ &\vdots \\ W_{\theta,i_1} &= \partial_{s^{i_1}} (\mathcal{K}^{-H} \widetilde{\mathcal{N}}^H)(\zeta_1) \\ &\quad \times \left(\prod_{i=2}^{\theta-1} \partial_{s^{v_i}} (I_{m^{i-1}} \otimes \mathcal{K}^{-H} \widetilde{\mathcal{N}}^H)(\zeta_i) \right) \\ &\quad \times (I_{m^{\theta-1}} \otimes \partial_{s^{v_\theta}} (\mathcal{K}^{-H} \mathcal{C}^H)(\zeta_\theta)), & i_1 &= 0, \dots, v_1, \\ \text{span}(W) &\supseteq \text{span}([W_{1,0}, \dots, W_{\theta,v_\theta}]), \end{aligned}$$

then the following interpolation conditions hold true:

$$\begin{aligned} \partial_{s_1^{i_\theta}} G_1(\zeta_\theta) &= \partial_{s_1^{i_\theta}} \widehat{G}_1(\zeta_\theta), & i_\theta &= 0, \dots, v_\theta, \\ &\vdots \\ \partial_{s_1^{i_1} s_2^{v_2} \dots s_\theta^{v_\theta}} G_\theta(\zeta_1, \dots, \zeta_\theta) &= \partial_{s_1^{i_1} s_2^{v_2} \dots s_\theta^{v_\theta}} \widehat{G}_\theta(\zeta_1, \dots, \zeta_\theta), & i_1 &= 0, \dots, v_1. \end{aligned}$$

(c) Let V be constructed as in part (a) and W as in part (b), then, additionally to the results in (a) and (b), the conditions

$$\begin{aligned} &\partial_{s_1^{\ell_1} \dots s_{q-1}^{\ell_{q-1}} s_q^{j_q} s_{q+1}^{i_{\theta-\eta+1}} s_{q+2}^{v_{\theta-\eta+2}} \dots s_{q+\eta}^{v_\theta}} G_{q+\eta}(\sigma_1, \dots, \sigma_q, \zeta_{\theta-\eta+1}, \dots, \zeta_\theta) \\ &= \partial_{s_1^{\ell_1} \dots s_{q-1}^{\ell_{q-1}} s_q^{j_q} s_{q+1}^{i_{\theta-\eta+1}} s_{q+2}^{v_{\theta-\eta+2}} \dots s_{q+\eta}^{v_\theta}} \widehat{G}_{q+\eta}(\sigma_1, \dots, \sigma_q, \zeta_{\theta-\eta+1}, \dots, \zeta_\theta), \end{aligned}$$

hold for $j_q = 0, \dots, \ell_q$; $i_{\theta-\eta+1} = 0, \dots, v_{\theta-\eta+1}$; $1 \leq q \leq k$ and $1 \leq \eta \leq \theta$.

Proof The results follow directly from Theorems 4, 5, and 7 with the same argumentation as in Theorem 8. $\square$

For completeness, also the implicit interpolation results are stated in the following corollary without additional proofs.

Corollary 2 (Two-sided matrix interpolation with identical point sets) *Let G be a bilinear system, described by (9), and $\widehat{G}$ the reduced-order bilinear system, constructed by (10). Given a set of interpolation points $\sigma_1, \dots, \sigma_k \in \mathbb{C}$, for which the matrix functions $\mathcal{C}(s)$, $\mathcal{K}(s)^{-1}$, $\mathcal{N}(s)$, $\mathcal{B}(s)$ are analytic and $\widehat{\mathcal{K}}(s)$ is full-rank, the following statements hold:*

- (a) *Let V and W be constructed as in Theorem 8 (a) and (b) for the interpolation points $\sigma_1, \dots, \sigma_k$, then additionally it holds*

$$\nabla G_k(\sigma_1, \dots, \sigma_k) = \nabla \widehat{G}_k(\sigma_1, \dots, \sigma_k).$$

- (b) *Let V and W be constructed as in Theorem 9 (a) and (b) for the interpolation points $\sigma_1, \dots, \sigma_k$ and orders of partial derivatives $\ell_1, \dots, \ell_k$, then additionally it holds*

$$\nabla \left(\partial_{s_1^{\ell_1} \dots s_k^{\ell_k}} G_k \right) (\sigma_1, \dots, \sigma_k) = \nabla \left(\partial_{s_1^{\ell_1} \dots s_k^{\ell_k}} \widehat{G}_k \right) (\sigma_1, \dots, \sigma_k).$$

4.2 Numerical example

We illustrate the matrix interpolation results using a numerical example. The experiments reported here have been executed on the same machine and with the same MATLAB version as in Section 3.4. The error measures were accordingly adapted to the MIMO case by

$$\text{err}_t(t) := \frac{\|y(t) - \widehat{y}(t)\|_2}{\|y(t)\|_2},$$

for time domain simulations, and

$$\begin{aligned} \text{err}_{f,1}(\omega_1) &:= \frac{\|G_1(\omega_1 i) - \widehat{G}_1(\omega_1 i)\|_2}{\|G_1(\omega_1 i)\|_2} \quad \text{and} \\ \text{err}_{f,2}(\omega_1, \omega_2) &:= \frac{\|G_2(\omega_1 i, \omega_2 i) - \widehat{G}_2(\omega_1 i, \omega_2 i)\|_2}{\|G_2(\omega_1 i, \omega_2 i)\|_2}, \end{aligned}$$

for the first and second subsystem transfer functions for the frequency domain error measures.

We reconsider the damped mass-spring system example from Section 3.4.2 with the following modifications: The mass, damping, and stiffness matrices from (20) stay unchanged. The input forces are now applied to the first and last masses, i.e., the input term becomes $B_u = [e_1, -e_n]$, and we observe the displacement of the second and fifth masses, which gives the output matrix $C_p = [e_2, e_5]^T$. Therefore, we have 2 inputs and outputs. We consider the same idea of bilinear springs as before, but working in different directions, i.e., we have

$$N_{p,1} = -S_1 K S_1 \quad \text{and} \quad N_{p,2} = S_2 K S_2,$$

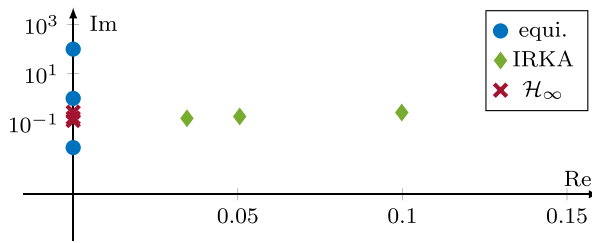


Fig. 9 Computed interpolation points for structured interpolation of the first subsystem transfer function for the MIMO damped mass-spring system (complex conjugates are not shown)

where S_1 is chosen, as before, as diagonal matrix with `linspace(0.2, 0, n)`, and S_2 is chosen to be a diagonal matrix with `linspace(0, 0.2, n)` as entries. Overall, we have a damped mass-spring system of the form

$$\begin{aligned} M\ddot{x}(t) + D\dot{x}(t) + Kx(t) &= N_{p,1}x(t)u_1(t) + N_{p,2}x(t)u_2(t) + B_u u(t), \\ y(t) &= C_p x(t), \end{aligned} \quad (21)$$

with $n = 1000$ masses for our experiments.

As in Section 3.4.2, we compare the structure-preserving interpolation method (StrInt) with two unstructured ones, using the first-order realization of (21) (FOInt, FOInt2). As before, the interpolation points are chosen as suggested in Section 3.3, where for the equidistant points for StrInt and FOInt were $\pm \logspace(-2, 2, 3)i$ and for FOInt2 $\pm \logspace(-2, 2, 6)i$. The interpolation points with nonnegative imaginary parts for the structured interpolation are plotted in Fig. 9 and closely resemble the points in Fig. 5. Also, we want again to interpolate the first and second subsystem transfer functions, and restrict ourselves to a one-sided projection as in part (a) of Theorem 8 by setting $W = V$, which yields the reduced orders $r = 36$ for StrInt and FOInt, and $r = 72$ for FOInt2.

Table 5 shows the relative approximation errors for the first subsystem transfer functions. In contrast to the corresponding numerical example in the SISO case, here FOInt2 with equidistant points has the smallest maximum relative error, directly followed by StrInt(IRKA) and StrInt(equi.). Also, it is interesting to observe that again the unstructured approach of double order performs worse, more specifically two orders of magnitude worse, for the choice of IRKA interpolation points. We decided to plot again only the results for the IRKA interpolation points as the first subsystem

Table 5 Maximum relative approximation errors of the first subsystem transfer functions $\max(\text{err}_{i,1})$ for the MIMO damped mass-spring system

	StrInt	FOInt	FOInt2
equi.	5.3753e-07	7.4705e-04	1.0971e-07
IRKA	1.5318e-07	1.8550e+02	2.1566e-05
$\mathcal{H}_\infty$	2.2538e-03	4.6784e+02	3.6330e-02

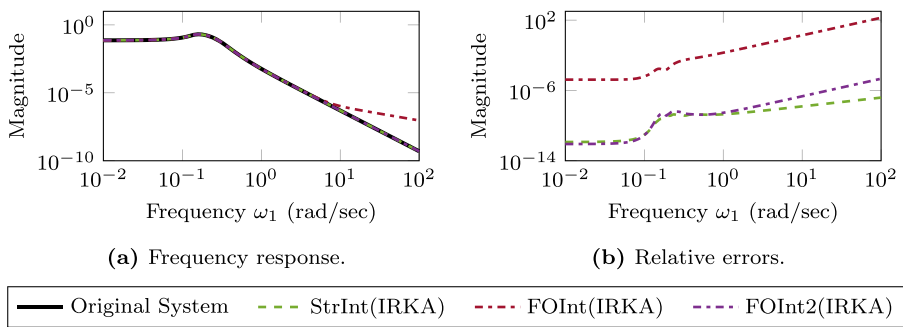


Fig. 10 Frequency domain results of the first subsystem transfer functions for the MIMO damped mass-spring system

transfer functions can be seen in Fig. 10. The results for the second subsystem transfer functions are given in Table 6. Here, StrInt(IRKA) is now the best with a similar maximum relative approximation error as for the first subsystem transfer function. Also, for the choice of equidistant interpolation points, the maximum relative approximation errors of all three approaches increased by around one order of magnitude. Figure 11 illustrates the relative approximation errors for the IRKA points.

Finally, we compare the computed reduced-order models in a time simulation in the interval $[0, 100]$ s with the input signal

$$u(t) = \begin{bmatrix} \sin(200t) + 200 \\ -\cos(200t) - 200 \end{bmatrix}.$$

The maximum relative approximation errors can be found in Table 7. Note that the reduced-order models with positive (large) exponents in the approximation errors, in fact, correspond to the unstable dynamics in the time simulation for the given input signal. Here, StrInt(IRKA) performs best, where actually all structure-preserving methods are compatible. Of the unstructured reduced-order models, only FOInt2 with either equidistant or IRKA points have a stable simulation behavior. Those two reduced-order models are one order of magnitude worse in the approximation error than StrInt(IRKA), but compatible with the other structure-preserving reduced-order models. The time simulations with errors for the IRKA-based interpolation points are displayed in Fig. 12.

Table 6 Maximum relative approximation errors of the second subsystem transfer functions $\max(\text{err}_{f,2})$ for the MIMO damped mass-spring system

	StrInt	FOInt	FOInt2
equi.	6.7652e-06	4.9870e-03	1.1460e-06
IRKA	1.1663e-07	1.4009e+02	1.6429e-05
$\mathcal{H}_\infty$	1.7164e-03	4.2192e+02	1.0338e-02

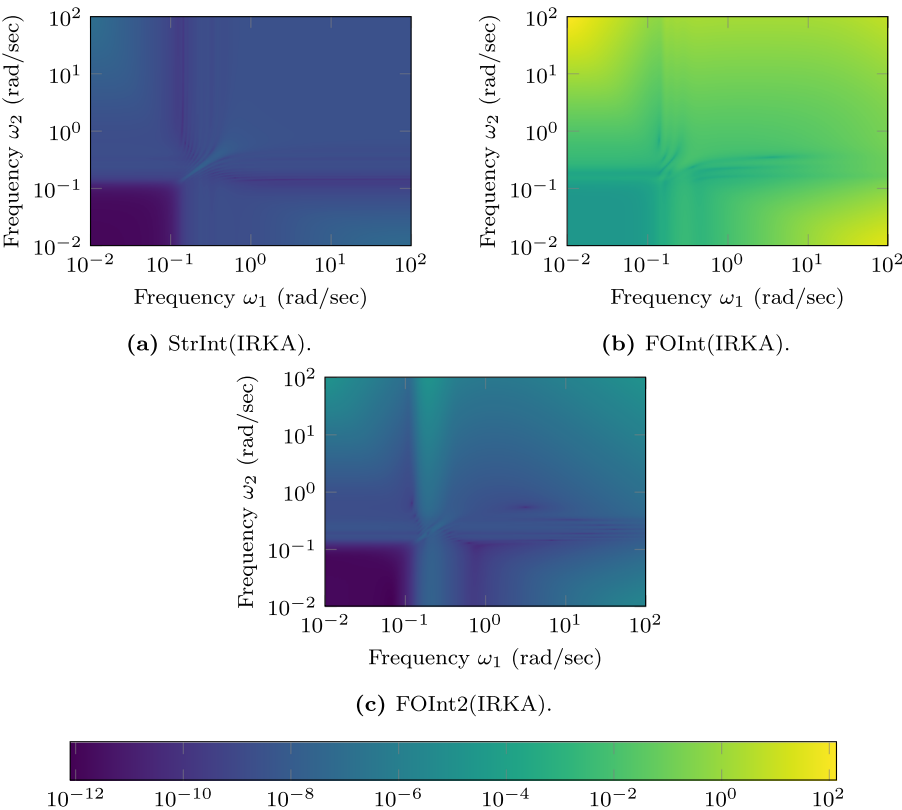


Fig. 11 Relative errors of the second subsystem transfer functions for the MIMO damped mass-spring system

Table 7 Maximum relative approximation errors of the time simulation $\max(\text{err}_t)$ for the MIMO damped mass-spring system

	StrInt	FOInt	FOInt2
equi.	4.3568e-05	3.6023e+04	7.5375e-04
IRKA	2.7527e-06	2.4655e+03	2.5988e-05
$\mathcal{H}_\infty$	1.5408e-04	1.4151e+16	1.4218e+09

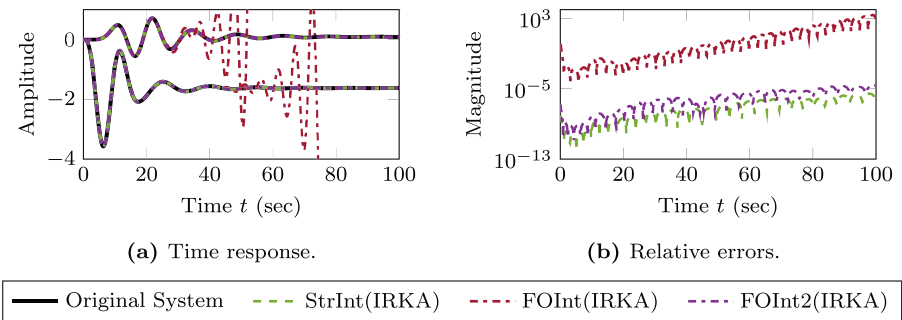


Fig. 12 Time simulation results for the MIMO damped mass-spring system

5 Conclusions

We extended the structure-preserving interpolation framework to bilinear control systems. First, we developed the subspace conditions for structured interpolation for single-input single-output systems, both for simple and Hermite interpolation. These results were extended to structured multi-input multi-output bilinear systems as well in the setting of full matrix interpolation. The effectiveness of the proposed approach was illustrated for two structured bilinear dynamical systems: a mass-spring-damper system and a model with internal delay. The theory developed here can be applied to a much broader class of structures than these two examples.

In our examples, we compared three strategies for choosing interpolation points inspired by the techniques developed for linear systems. The question for a guaranteed high-fidelity or optimal choice of interpolation points in the considered structured bilinear setting remains open. This issue is not fully resolved even for structure-preserving interpolation of linear dynamical systems. Another issue to further investigate is the rapidly enlarging reduced-order dimension in case of the matrix interpolation approach for multi-input multi-output systems. While in the linear case, tangential interpolation can be used to control the growth of the basis, there is no uniform treatment of tangential interpolation for bilinear systems yet. This issue will be studied in a forthcoming work.

Acknowledgments We would like to thank Jens Saak for constructive discussions about the notation and structure of this paper, and Igor Pontes Duff Pereira and Ion Victor Gosea for providing MATLAB codes used for generating the bilinear time-delay example.

Funding Open Access funding enabled and organized by Projekt DEAL. Benner and Werner were supported by the German Research Foundation (DFG) Research Training Group 2297 “MathCoRe,” Magdeburg, and the German Research Foundation (DFG) Priority Program 1897: “Calm, Smooth and Smart – Novel Approaches for Influencing Vibrations by Means of Deliberately Introduced Dissipation.” Gugercin was supported in parts by National Science Foundation under Grant No. DMS-1720257 and DMS-1819110. Part of this material is based upon work supported by the National Science Foundation under Grant No. DMS-1439786 and by the Simons Foundation Grant No. 50736 while Gugercin and Benner were in residence at the Institute for Computational and Experimental Research in Mathematics in Providence, RI, during the “Model and dimension reduction in uncertain and dynamic systems” program.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Mohler, R.R.: Natural bilinear control processes. *IEEE Transactions on Systems Science and Cybernetics* 6(3), 192–197 (1970). <https://doi.org/10.1109/TSSC.1970.300341>

2. Mohler, R.R.: Bilinear control processes: With applications to engineering, ecology and medicine. Mathematics in Science and Engineering, vol. 106. Academic Press, New York, London (1973)
3. Al-Baiyat, S., Farag, A.S., Bettayeb, M.: Transient approximation of a bilinear two-area interconnected power system. *Electr. Power Syst. Res.* **26**(1), 11–19 (1993). [https://doi.org/10.1016/0378-7796\(93\)90064-L](https://doi.org/10.1016/0378-7796(93)90064-L)
4. Ou, Y.: Optimal control of a class of nonlinear parabolic PDE systems arising in fusion plasma current profile dynamics. Ph.D. Thesis, Lehigh University (2010)
5. Qian, K., Zhang, Y.: Bilinear model predictive control of plasma keyhole pipe welding process. *J. Manuf. Sci. Eng.* **136**(3), 031002 (2014). <https://doi.org/10.1115/1.4025337>
6. Saputra, J., Saragih, R., Handayani, D.: Robust H_∞ controller for bilinear system to minimize HIV concentration in blood plasma. *J. Phys.: Conf. Ser.* **1245**, 012055 (2019). <https://doi.org/10.1088/1742-6596/1245/1/012055>
7. Carleman, T.: Application de la théorie des équations intégrales linéaires aux systèmes d'équations différentielles non linéaires. *Acta Math.* **59**, 63–87 (1932). <https://doi.org/10.1007/BF02546499>
8. Kowalski, K., Steeb, W.-H.: Nonlinear dynamical systems and Carleman linearization. World Scientific, Singapore (1991). <https://doi.org/10.1142/1347>
9. Khapalov, A.Y.: Controllability of the semilinear parabolic equation governed by a multiplicative control in the reaction term: a qualitative approach. In: 42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475), vol. 2, pp. 1487–1491 (2003)
10. Korpeoglu, S.G., Kucuk, I.: Optimal control of a bilinear system with a quadratic cost functional. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBE), pp. 1–6 (2018). <https://doi.org/10.1109/ICCUBE.2018.8697554>
11. Benner, P., Damm, T.: Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems. *SIAM Control Opt.* **49**(2), 686–711 (2011). <https://doi.org/10.1137/09075041X>
12. Benner, P., Breiten, T.: On H_2 -model reduction of linear parameter-varying systems. *PAMM* **11**(1), 805–806 (2011). <https://doi.org/10.1002/pamm.201110391>
13. Bruns, A., Benner, P.: Parametric model order reduction of thermal models using the bilinear interpolatory rational Krylov algorithm. *MCMDS* **21**(2), 103–129 (2015). <https://doi.org/10.1080/13873954.2014.924534>
14. Benner, P., Cao, X., Schilders, W.: A bilinear $\mathcal{H}_2$ model order reduction approach to linear parameter-varying systems. *Adv. Comp. Math.* **45**, 2241–2271 (2019). <https://doi.org/10.1007/s10444-019-09695-9>
15. Hsu, C.S., Desai, U.B., Crawley, C.A.: Realization algorithms and approximation methods of bilinear systems. In: The 22nd IEEE Conference on Decision and Control, San Antonio, TX, USA, pp. 783–788 (1983). <https://doi.org/10.1109/CDC.1983.269628>
16. Bai, Z., Skoogh, D.: A projection method for model reduction of bilinear dynamical systems. *Lin. Alg. App.* **415**(2–3), 406–425 (2006). <https://doi.org/10.1016/j.laa.2005.04.032>
17. Condon, M., Ivanov, R.: Krylov subspaces from bilinear representations of nonlinear systems. *Compel-Int. J. Comp. Math. Electr. Electron. Eng.* **26**(2), 399–406 (2007). <https://doi.org/10.1108/03321640710727755>
18. Feng, L., Benner, P.: A note on projection techniques for model order reduction of bilinear systems. In: AIP Conference Proceedings, vol. 936, pp. 208–211 (2007). <https://doi.org/10.1063/1.2790110>
19. Breiten, T., Damm, T.: Krylov subspace methods for model order reduction of bilinear control systems. *SCL* **59**(8), 443–450 (2010). <https://doi.org/10.1016/j.sysconle.2010.06.003>
20. Antoulas, A.C., Beattie, C.A., Gugercin, S.: Interpolatory methods for model reduction. Computational Science & Engineering. Society for Industrial and Applied Mathematics, Philadelphia, PA (2020). <https://doi.org/10.1137/1.9781611976083>
21. Zhang, L., Lam, J.: On H_2 model reduction of bilinear systems. *Automatica* **38**(2), 205–216 (2002). [https://doi.org/10.1016/S0005-1098\(01\)00204-7](https://doi.org/10.1016/S0005-1098(01)00204-7)
22. Benner, P., Breiten, T.: Interpolation-based $\mathcal{H}_2$ -model reduction of bilinear control systems. *SIAM Matrix* **33**(3), 859–885 (2012). <https://doi.org/10.1137/110836742>
23. Flagg, G.M., Gugercin, S.: Multipoint Volterra series interpolation and $\mathcal{H}_2$ optimal model reduction of bilinear systems. *SIAM Matrix* **36**(2), 549–579 (2015). <https://doi.org/10.1137/130947830>
24. Antoulas, A.C., Gosea, I.V., Ionita, A.C.: Model reduction of bilinear systems in the Loewner framework. *SIAM Sci. Comp.* **38**(5), B889–B916 (2016). <https://doi.org/10.1137/15M1041432>

25. Gosea, I.V., Pontes Duff, I., Benner, P., Antoulas, A.C.: Model order reduction of bilinear time-delay systems. In: Proc. of 18th European Control Conference (ECC), pp. 2289–2294 (2019). <https://doi.org/10.23919/ECC.2019.8796085>
26. Beattie, C.A., Gugercin, S.: Interpolatory projection methods for structure-preserving model reduction. *SCL* **58**(3), 225–232 (2009). <https://doi.org/10.1016/j.sysconle.2008.10.016>
27. Rugh, W.J.: Nonlinear system theory: The Volterra/Wiener approach. Johns Hopkins Press, Baltimore (1981)
28. Gugercin, S., Stykel, T., Wyatt, S.: Model reduction of descriptor systems by interpolatory projection methods. *SIAM Sci. Comp.* **35**(5), B1010–B1033 (2013). <https://doi.org/10.1137/130906635>
29. Benner, P., Goyal, P.: Multipoint interpolation of Volterra series and $\mathcal{H}_2$ -model reduction for a family of bilinear descriptor systems. *SCL* **97**, 1–11 (2016). <https://doi.org/10.1016/j.sysconle.2016.08.008>
30. Ahmad, M.I., Benner, P., Goyal, P.: Krylov subspace-based model reduction for a class of bilinear descriptor systems. *Comp. Appl. Math.* **315**, 303–318 (2017). <https://doi.org/10.1016/j.cam.2016.11.009>
31. Gugercin, S., Antoulas, A.C., Beattie, C.: $\mathcal{H}_2$ model reduction for large-scale linear dynamical systems. *SIAM Matrix* **30**(2), 609–638 (2008). <https://doi.org/10.1137/060666123>
32. Benner, P., Breiten, T.: Two-sided projection methods for nonlinear model order reduction. *SIAM Sci. Comp.* **37**(2), B239–B260 (2015). <https://doi.org/10.1137/14097255X>
33. Wyatt, S.: Issues in interpolatory model reduction: Inexact solves, second-order systems and DAEs. Ph.D. Thesis, VTech. <http://hdl.handle.net/10919/27668> (2012)
34. Beattie, C.A., Benner, P.: $\mathcal{H}_2$ -optimality conditions for structured dynamical systems. Preprint MPIMD/14-18, Max Planck Institute Magdeburg. <https://csc.mpi-magdeburg.mpg.de/preprints/2014/18/> (2014)
35. Mlinarić, P.: Structure-preserving model order reduction for network systems. Dissertation, Department of Mathematics, Otto von Guericke University. <https://doi.org/10.25673/33570> (2020)
36. Pontes Duff, I., Gugercin, S., Beattie, C., Poussot-Vassal, C., Seren, C.: $\mathcal{H}_2$ -optimality conditions for reduced time-delay systems of dimensions one. *IFAC-Papers Online* **49**(10), 7–12 (2016). <https://doi.org/10.1016/j.ifacol.2016.07.464>. 13th IFAC on Time Delay Systems TDS 2019
37. Pontes Duff, I., Poussot-Vassal, C., Seren, C.: Realization independent single time-delay dynamical model interpolation and $\mathcal{H}_2$ -optimal approximation. In: 54th IEEE Conference on Decision and Control (CDC), pp. 4662–4667 (2015). <https://doi.org/10.1109/CDC.2015.7402946>
38. Pontes Duff, I., Poussot-Vassal, C., Seren, C.: $\mathcal{H}_2$ -optimal model approximation by input/output-delay structured reduced-order models. *SCL* **117**, 60–67 (2018). <https://doi.org/10.1016/j.sysconle.2018.05.003>
39. 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>
40. Mayo, A.J., Antoulas, A.C.: A framework for the solution of the generalized realization problem. *Lin. Alg. App.* **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
41. Beddig, R.S., Benner, P., Dorschky, I., Reis, T., Schwerdtner, P., Voigt, M., Werner, S.W.R.: Model reduction for second-order dynamical systems revisited. *PAMM* **19**(1), e201900224 (2019). <https://doi.org/10.1002/pamm.201900224>
42. 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. *arXiv:2010.06331*. math.OC (2020)
43. Aliyev, N., Benner, P., Mengi, E., Schwerdtner, P., Voigt, M.: Large-scale computation of $\mathcal{L}_\infty$ -norms by a greedy subspace method. *SIAM Matrix* **38**(4), 1496–1516 (2017). <https://doi.org/10.1137/16M1086200>
44. Aliyev, N., Benner, P., Mengi, E., Schwerdtner, P., Voigt, M.: A greedy subspace method for computing the $\mathcal{L}_\infty$ -norm. *PAMM* **17**(1), 751–752 (2017). <https://doi.org/10.1002/pamm.201710343>
45. Aliyev, N., Benner, P., Mengi, E., Voigt, M.: A subspace framework for $\mathcal{H}_\infty$ -norm minimization. *SIAM Matrix* **41**(2), 928–956 (2020). <https://doi.org/10.1137/19M125892X>
46. Feng, L., Antoulas, A.C., Benner, P.: Some a posteriori error bounds for reduced order modelling of (non-)parametrized linear systems. *ESAIM: M2AN* **51**(6), 2127–2158 (2017). <https://doi.org/10.1051/m2an/2017014>
47. Antoulas, A.C., Benner, P., Feng, L.: Model reduction by iterative error system approximation. *MCMDS* **24**(2), 103–118 (2018). <https://doi.org/10.1080/13873954.2018.1427116>

48. Mehrmann, V., Stykel, T.: Balanced truncation model reduction for large-scale systems in descriptor form. In: Benner, P., Mehrmann, V., Sorensen, D.C. (eds.) *Dimension reduction of large-scale systems*, Lect. Notes Comput. Sci. Eng., vol. 45, pp. 83–115. Springer-Verlag, Berlin/Heidelberg, Germany (2005). https://doi.org/10.1007/3-540-27909-1_3
49. Benner, P., Saak, J.: Efficient balancing-based MOR for large-scale second-order systems. *MCMDS* **17**(2), 123–143 (2011). <https://doi.org/10.1080/13873954.2010.540822>
50. Gallivan, K., Vandendorpe, A., Van Dooren, P.: Model reduction of MIMO systems via tangential interpolation. *SIAM Matrix* **26**(2), 328–349 (2004). <https://doi.org/10.1137/S0895479803423925>
51. Benner, P., Breiten, T., Damm, T.: Generalized tangential interpolation for model reduction of discrete-time MIMO bilinear systems. *Int. Control* **84**(8), 1398–1407 (2011). <https://doi.org/10.1080/00207179.2011.601761>
52. Rodriguez, A.C., Gugercin, S., Boggaard, J.: Interpolatory model reduction of parameterized bilinear dynamical systems. *Adv. Comp. Math.* **44**(6), 1887–1916 (2018). <https://doi.org/10.1007/s10444-018-9611-y>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.