Data-driven balancing of linear dynamical systems

Ion Victor Gosea[*] Serkan Gugercin[†] Christopher Beattie[†]

[*]*Max Planck Institute for Dynamics of Complex Technical Systems,*
Sandtorstr. 1, 39106 Magdeburg, Germany.
Email: gosea@mpi-magdeburg.mpg.de, ORCID: 0000-0003-3580-4116
[†]*Department of Mathematics and Computational Modeling and Data Analytics Division,*
Academy of Integrated Science, Virginia Tech, Blacksburg, VA 24061, USA.
Email: gugercin@vt.edu, ORCID: 0000-0003-4564-5999
Email: beattie@vt.edu, ORCID: 0000-0003-3302-4845

We present a novel reformulation of balanced truncation, a classical model reduction method. The principal innovation that we introduce comes through the use of system response data that has been either measured or computed, without reference to any prescribed realization of the original model. Data are represented by sampled values of the transfer function or the impulse response corresponding to the original model. We discuss parallels that our approach bears with the Loewner framework, another popular data-driven model reduction method. We illustrate our approach numerically in both continuous-time and discrete-time cases.

Keywords: Balanced truncation, numerical quadrature, data-driven modeling, transfer functions, impulse responses, Gramians.

AMS subject classifications: 37M99, 41A20, 65D30, 65K99, 93A15, 93B15.

1 Introduction

Model reduction (MoR) here refers to system-theoretic techniques used to create compactly represented *reduced models* that are capable of reproducing the input-output behaviour of large-scale dynamical systems with high accuracy. This is accomplished by encoding fine scale dynamical features of the original systems efficiently into *reduced* dynamical systems, allowing them to closely mimic the input/output behavior of the original system for a wide range of input conditions, while having substantially lower order than the system whose behaviour they mimic. Such reduced models should be cheap to simulate and easy to manipulate and control. We refer the reader to [1,2,10,47] for details on a variety of MoR techniques.

Balanced truncation (BT) [38,39] is one of the most successful and commonly deployed MoR methods. For linear dynamical systems such as those that we consider in this paper, BT retains asymptotic stability of the original systems and provides *a priori* bounds for the model reduction error. BT effectively discards those states that are both difficult to reach and to observe, as quantified through the relative magnitude of the system's Hankel singular values. Typically, the loss of these states is expected to have little effect on the observed input-output dynamics of the system and the resulting reduced model may replace the original system in simulations or analysis without much lost with regard to accuracy. Extensions of BT to nonlinear systems have emerged over the years, starting with [51] and continuing with more recent approaches tailored to particular classes of nonlinearities, such as bilinear systems in [7], quadratic-bilinear systems in [8], and to switched systems in [24].

The major computational cost of BT is the need to solve large-scale Lyapunov equations in order to obtain the system Gramians, or better, their square-root factors. Many methods have been proposed to accomplish this efficiently; see, e.g., [11,34,35,43,50,54,61] and references therein. All such methods make central use of the system state-space representation and, in the context of BT, produce explicit state-space projections that ultimately are engaged to generate a reduced model. Thus, BT is "intrusive" to the extent that internal representations of the system dynamics play a central role in determining a final reduced model. This is to be contrasted with "nonintrusive" methods that are *data-driven*, requiring access only to system response and behavior data, e.g., transfer function or frequency response measurements.

In this paper, we develop a new, data-driven formulation for BT. We achieve this by recognizing that BT does not make independent use of the two system Gramians (which separately each depend on internal and hence generally inaccessible variables). BT rather makes fundamental use of their *product*, which preserves important system invariants that do not depend on the underlying system realization. In this work, we describe how the product of Gramians can be approximated directly through transfer function sampling to any accuracy desired. We explicitly derive reduced-order quantities using observed response data in this way and as consequence, arrive at a novel, *non-intrusive* formulation of BT.

The rest of the paper is organized as follows: In §2, we review the basic elements of BT and detail the usual steps involved for computing balanced ROMs. We introduce our main result in §3, a data-driven approach to BT and discuss how one estimates key quantities through quadrature in the frequency-domain, that in turn, involves quantities that are extracted from data. In §4, we include refinements that take into account symmetries implicit in real-valued dynamics as well as extensions to MIMO, discrete-time and application to infinite-dimensional systems. Even though our main focus is frequency domain data, in §5 we revisit the problem using time-domain data instead and make connections to earlier works. Conclusions are provided in §6, while appendices elaborating on the quadrature rules considered and detailing the proof of Proposition 3.2 follow the bibliography. Numerical experiments and various illustrations are provided in §§3, 4, and 5.

2 Balanced Truncation

Consider LTI systems

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

where the input mapping is given by $\mathbf{u} : \mathbb{R} \to \mathbb{R}^m$, the (generalized) state trajectory/variable is $\mathbf{x} : \mathbb{R} \to \mathbb{R}^n$, and the output mapping is $y : \mathbb{R} \to \mathbb{R}^p$. The system matrices are given by $\mathbf{E}$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, and $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$. Assume $\mathbf{E}$ is nonsingular and that the matrix pencil $(\mathbf{A},\mathbf{E})$ is *asymptotically stable*, meaning that the eigenvalues of $(\mathbf{A},\mathbf{E})$ (or equivalently of $\mathbf{E}^{-1}\mathbf{A}$) are located in the open left half-plane. The transfer function of the LTI system (1), defined as $\mathbf{H}(s) = \mathbf{C}(s\mathbf{E}-\mathbf{A})^{-1}\mathbf{B}$ is a $p \times m$ matrix-valued rational function in s.

The two fundamental quantities in BT are the reachability and observability Gramians. The reachability Gramian $\mathbf{P}$ provides a measure of how easily a state can be accessed from the zero state. In the time domain, $\mathbf{P}$ is given by

$$\mathbf{P} = \int_0^\infty e^{\mathbf{E}^{-1}\mathbf{A}t}\mathbf{E}^{-1}\mathbf{B}\mathbf{B}^T\mathbf{E}^{-T}e^{\mathbf{A}^T\mathbf{E}^{-T}t}dt. \tag{2}$$

Taking $\imath^2 = -1$, this can be represented equivalently in the frequency domain as

$$\mathbf{P} = \frac{1}{2\pi}\int_{-\infty}^{\infty}(\imath\zeta\mathbf{E}-\mathbf{A})^{-1}\mathbf{B}\mathbf{B}^T(-\imath\zeta\mathbf{E}^T-\mathbf{A}^T)^{-1}d\zeta. \tag{3}$$

In a complementary way, the observability Gramian describes how easily a state can be observed, and may be represented as $\mathbf{E}^T\mathbf{Q}\mathbf{E}$ where $\mathbf{Q}$ is defined as:

$$\mathbf{Q} = \frac{1}{2\pi}\int_{-\infty}^{\infty}(-\imath\omega\mathbf{E}^T - \mathbf{A}^T)^{-1}\mathbf{C}^T\mathbf{C}(\imath\omega\mathbf{E} - \mathbf{A})^{-1}d\omega. \tag{4}$$

$\mathbf{P}$ and $\mathbf{Q}$ satisfy the following generalized Lyapunov equations:

$$\mathbf{A}\mathbf{P}\mathbf{E}^T + \mathbf{E}\mathbf{P}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = \mathbf{0}, \tag{5}$$

$$\mathbf{A}^T\mathbf{Q}\mathbf{E} + \mathbf{E}^T\mathbf{Q}\mathbf{A} + \mathbf{C}^T\mathbf{C} = \mathbf{0}. \tag{6}$$

Even though $\mathbf{E}^T\mathbf{Q}\mathbf{E}$ is the observability Gramian in this case; the Lyapunov equation (6) for $\mathbf{Q}$ enjoys a fundamental structural similarity to (5) and so similar computational algorithms are applicable to both (see, e.g., [11, 43, 57]). The solutions, $\mathbf{P}$ and $\mathbf{Q}$, to (5) and (6), respectively, are symmetric positive-semidefinite matrices, and one may compute square factors $\mathbf{L}, \mathbf{U} \in \mathbb{R}^{n\times n}$ (e.g., via a Cholesky factorization) so that

$$\mathbf{P} = \mathbf{U}\mathbf{U}^T \quad \text{and} \quad \mathbf{Q} = \mathbf{L}\mathbf{L}^T. \tag{7}$$

This provides the essential elements allowing us to define *balanced truncation* (BT), which we summarize in Algorithm 1. The singular values of $\mathbf{L}^T\mathbf{E}\mathbf{U}$, i.e., the diagonal entries of $\mathsf{diag}(\mathbf{S}_1, \mathbf{S}_2)$ in (8), are the *Hankel singular values* of the associated dynamical system and these values are system invariants that are independent of realization. BT truncates the states that correspond to small Hankel singular values in $\mathbf{S}_2$. The rth-order reduced model $\mathbf{H}_r(s) = \mathbf{C}_r(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{B}_r$ resulting from BT in Algorithm 1 is asymptotically stable and satisfies $\|\mathbf{H} - \mathbf{H}_r\|_{\mathcal{H}_\infty} \leq 2\,\mathsf{trace}(\mathbf{S}_2)$, where $\|\cdot\|_{\mathcal{H}_\infty}$ denotes the $\mathcal{H}_\infty$-norm of a dynamical and where we assumed, for simplicity, that the Hankel singular values are distinct. For details, we refer the reader to [1].

Algorithm 1 Balanced truncation (BT) (square-root implementation)

```
Input: LTI system described by matrices $\mathbf{E}$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, and $\mathbf{C} \in \mathbb{R}^{p\times n}$.
Output: BT reduced-system: $\mathbf{A}_r \in \mathbb{R}^{r\times r}, \mathbf{B}_r \in \mathbb{R}^{r\times m}, \mathbf{C}_r \in \mathbb{R}^{p\times r}$.
1: Compute the Lyapunov factors $\mathbf{U}, \mathbf{L} \in \mathbb{R}^{n\times n}$ from (7) and pick a truncation index, $1 \leq r \leq \min(\mathsf{rank}(\mathbf{U}), \mathsf{rank}(\mathbf{L}))$.
2: Compute the SVD of the matrix $\mathbf{L}^T\mathbf{E}\mathbf{U}$, partitioned as follows
       $\mathbf{L}^T\mathbf{E}\mathbf{U} = \begin{bmatrix}\mathbf{Z}_1 & \mathbf{Z}_2\end{bmatrix}\begin{bmatrix}\mathbf{S}_1 & \\ & \mathbf{S}_2\end{bmatrix}\begin{bmatrix}\mathbf{Y}_1^T \\ \mathbf{Y}_2^T\end{bmatrix},$   (8)
   where $\mathbf{S}_1 \in \mathbb{R}^{r\times r}$ and $\mathbf{S}_2 \in \mathbb{R}^{(n-r)\times(n-r)}$.
3: Construct the model reduction bases
       $\mathbf{W}_r = \mathbf{L}\mathbf{Z}_1\mathbf{S}_1^{-1/2}$ and $\mathbf{V}_r = \mathbf{U}\mathbf{Y}_1\mathbf{S}_1^{-1/2}$.   (9)
4: The reduced-order system matrices are given by
       $\mathbf{E}_r = \mathbf{W}_r^T\mathbf{E}\mathbf{V}_r = \mathbf{I}_r$   (10)
       $\mathbf{A}_r = \mathbf{W}_r^T\mathbf{A}\mathbf{V}_r = \mathbf{S}_1^{-1/2}\mathbf{Z}_1^T\left(\mathbf{L}^T\mathbf{A}\mathbf{U}\right)\mathbf{Y}_1\mathbf{S}_1^{-1/2},$   (11)
       $\mathbf{B}_r = \mathbf{W}_r^T\mathbf{B} = \mathbf{S}_1^{-1/2}\mathbf{Z}_1^T\left(\mathbf{L}^T\mathbf{B}\right),$   (12)
       $\mathbf{C}_r = \mathbf{C}\mathbf{V}_r = \left(\mathbf{C}\mathbf{U}\right)\mathbf{Y}_1\mathbf{S}_1^{-1/2}.$   (13)
```

3 A Data Driven Framework for Balancing

The main innovation that we introduce centers on the observation that the key quantities $\mathbf{L}^T\mathbf{E}\mathbf{U}$, $\mathbf{L}^T\mathbf{A}\mathbf{U}$, $\mathbf{L}^T\mathbf{B}$, and $\mathbf{C}\mathbf{U}$, appearing in Steps 1 and 3 of Algorithm 1, may be replaced by unitarily equivalent quantities that are well approximated directly from data. "Data" in our setting correspond to sampling of the transfer function $\mathbf{H}(s)$; we assume access to the values of $\mathbf{H}(\imath\hat{\omega})$ at a finite number of frequencies $\hat{\omega}$. (We consider time-domain sampling in §5.) Our choice of frequency sampling, $\hat{\omega}$, will be associated with numerical quadratures used to approximate (3) and (4). Further details are provided below and in Appendix A, however we note here that some quadrature rules also engage asymptotics of the integrand which in our setting may then require measuring the leading two Markov parameters of $\mathbf{H}(s)$, i.e., the leading two coefficients in the expansion of $\mathbf{H}(s)$ around $s = \infty$. For $\mathbf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}$, $\mathbf{M}_0 = \mathbf{C}\mathbf{E}^{-1}\mathbf{B}$ is the zeroth Markov parameter (the first coefficient) and $\mathbf{M}_1 = \mathbf{C}\mathbf{E}^{-1}\mathbf{A}\mathbf{E}^{-1}\mathbf{B}$ is the first Markov parameter (the second expansion coefficient). These quantities can be measured nonintrusively, as explained in, e.g., [31] with practical applications ranging from Vector Network Analyzers [32] to 3D Laser Vibrometers. [62].

To simplify initial discussion, we consider first SISO systems, taking $\mathbf{B} = \mathbf{b}$ and $\mathbf{C} = \mathbf{c}^T$, for $\mathbf{b}, \mathbf{c} \in \mathbb{R}^n$ and the associated (scalar-valued) transfer function, $H(s)$[1]. Generalization to MIMO systems is straightforward and discussed in §4.2.

3.1 Computing key quantities from data

For large-scale dynamical systems, solving the Lyapunov equations (5) and (6) for the Gramians $\mathbf{P}$ and $\mathbf{Q}$ is computationally demanding. There are many methods for solving these equations; see, e.g., [1, 5, 11, 34, 50, 54] and the references therein. Then, using any of these techniques for Lyapunov equations, one proceeds by finding a low-rank approximation to $\mathbf{U}$ and $\mathbf{L}$, and Algorithm 1 proceeds by using these approximate low-rank factors. Regardless of the underlying Lyapunov solvers, these approximate balancing approaches are intrusive in the sense that they depend on explicit system realizations, model projection, i.e., $\mathbf{U}$ and $\mathbf{L}$ are approximated first, followed by an explicit projection step. What we discuss below sidesteps this by focusing on a quantity that is unitarily equivalent to $\mathbf{L}^T\mathbf{E}\mathbf{U}$, leading to an non-intrusive data-driven approach to BT.

Assume that we employ a numerical quadrature rule to approximate the frequency integral defining $\mathbf{P}$ in (3), producing an approximate Gramian

$$\mathbf{P} \approx \widetilde{\mathbf{P}} = \sum_{j=1}^{\ell_p} \rho_j^2 (\imath\zeta_j\mathbf{E} - \mathbf{A})^{-1}\mathbf{b}\mathbf{b}^T(-\imath\zeta_j\mathbf{E}^T - \mathbf{A}^T)^{-1} + \rho_\infty^2\mathbf{E}^{-1}\mathbf{b}\mathbf{b}^T\mathbf{E}^{-T} \tag{14}$$

with ρ_j^2 and ζ_j denoting, respectively, quadrature weights and nodes. The last term involving the weight ρ_∞ corresponds to a "node at infinity" and relates to the asymptotic decay of the integrand in (3) in the neighborhood of $\zeta = \infty$. The presence of this term in (14) will depend on the choice of numerical quadrature. We will investigate choices where this term is present and others where it is absent. In either case, we label the total number of quadrature points (including potentially a "node at infinity") as N_p. If the ρ_∞-term in (14) is absent, then $\ell_p = N_p$; otherwise, $\ell_p = N_p - 1$. Picking a distribution of quadrature nodes on the imaginary axis is natural in this context; we focus on two standard exemplars derived from the trapezoid rule and Clenshaw-Curtis quadrature and provide further details in Appendix A. Quadrature node distributions can be chosen on different integration contours as well (e.g.,[60]) and the approach that we propose is equally applicable in these modified circumstances as well.

The Balanced POD approach of [61] uses this type of approximation for $\mathbf{P}$, employing the trapezoid rule as an underlying numerical quadrature approximation. A modified version of Balanced POD based

[1] We distinguish matrix- and vector-valued quantities from scalar-valued quantities with boldface; hence $H(s)$ for SISO vs. $\mathbf{H}(s)$ for the MIMO counterpart, and similarly for Markov parameters, M_0 and M_1 for SISO vs. $\mathbf{M}_0$ and $\mathbf{M}_1$ for the MIMO counterparts.

on a time-domain quadrature applied to (2) was later proposed in [49] with an application to fluid dynamics. As is the case with other approximate balancing techniques, Balanced POD is *intrusive* in the sense that access to an explicit state-space realization is required. Any approximate balancing technique produces some sort of approximation to the Lyapunov factors $\mathbf{U}$ and $\mathbf{L}$ from (7), including ADI approaches (described in, e.g., [11, 35, 44, 56]).

To the extent that system response data may be utilized, generally speaking its use is directed toward explicit sampling of evolving states of the system which tacitly requires a defined state space realization. This contrasts significantly with the *nonintrusive* data-driven approach that we propose below. While there are important common themes that we adopt from Balanced POD, we do not require any knowledge of a state space realization; we do not sample state trajectories even implicitly, and we do not attempt to approximate the Lyapunov factors in (7). Our approach has much in common with TF-IRKA in [6] in that we are able to express a model reduction strategy that is traditionally formulated in terms of state space realizations in a manner that requires only transfer function evaluations. We revisit Balanced POD and interpret it within our framework in §3.3.

Evidently, we may decompose the quadrature-based Gramian approximation as $\widetilde{\mathbf{P}} = \widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^*$ with a square-root factor $\widetilde{\mathbf{U}} \in \mathbb{C}^{n\times N_p}$ defined as

$$\widetilde{\mathbf{U}} = \Big[\, \rho_1(\imath\zeta_1\mathbf{E}-\mathbf{A})^{-1}\mathbf{b} \quad \cdots \quad \rho_{\ell_p}(\imath\zeta_{\ell_p}\mathbf{E}-\mathbf{A})^{-1}\mathbf{b} \quad \rho_\infty\mathbf{E}^{-1}\mathbf{b}\Big]. \tag{15}$$

Note that both $\mathbf{P}$ and its quadrature approximation, $\widetilde{\mathbf{P}}$, are real-valued matrices, yet the explicit square-root factor, $\widetilde{\mathbf{U}}$, is overtly complex and subsequent computation engages complex floating point arithmetic. A reasonable concern that may arise at this point is the potential loss of an underlying structural system symmetry, leaving us possibly with a complex-valued reduced model as an artifact of rounding errors. We address this concern in §4.1 and show how the underlying system symmetry associated with real dynamics may be explicitly preserved, allowing also the whole procedure to be implemented in real arithmetic.

By applying a similar quadrature approximation to $\mathbf{Q}$, we obtain

$$\mathbf{Q} \approx \widetilde{\mathbf{Q}} = \sum_{k=1}^{\ell_q} \phi_k^2(-\imath\omega_k\mathbf{E}^T-\mathbf{A}^T)^{-1}\mathbf{c}\mathbf{c}^T(\imath\omega_k\mathbf{E}-\mathbf{A})^{-1} + \phi_\infty^2\mathbf{E}^{-T}\mathbf{c}\mathbf{c}^T\mathbf{E}^{-1}, \tag{16}$$

where ϕ_k^2 and ω_k denote, respectively, quadrature weights and nodes associated with approximating $\mathbf{Q}$ from (4). We introduce a corresponding square-root factor from $\widetilde{\mathbf{Q}} = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^*$ where

$$\tilde{\mathbf{L}}^* = \begin{bmatrix} \phi_1\mathbf{c}^T(\imath\omega_1\mathbf{E}-\mathbf{A})^{-1} \\ \phi_2\mathbf{c}^T(\imath\omega_2\mathbf{E}-\mathbf{A})^{-1} \\ \vdots \\ \phi_{\ell_q}\mathbf{c}^T(\imath\omega_{\ell_q}\mathbf{E}-\mathbf{A})^{-1} \\ \phi_\infty\mathbf{c}^T\mathbf{E}^{-1} \end{bmatrix} \in \mathbb{C}^{N_q\times n}. \tag{17}$$

Let us suppose for the moment that the quadratures described in (14) and (16) *exactly* recovered the Gramians described in (3) and (4), respectively, so that as a consequence we have

$$\mathbf{U}\mathbf{U}^T = \widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^* \quad \text{and} \quad \mathbf{L}\mathbf{L}^T = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^*. \tag{18}$$

This, in turn, assures us of the existence of partial isometries, $\mathbf{\Psi}_p \in \mathbb{C}^{N_p\times n}$ and $\mathbf{\Psi}_q \in \mathbb{C}^{N_q\times n}$ such that

$$\widetilde{\mathbf{U}}^* = \mathbf{\Psi}_p\mathbf{U}^T \quad \text{and} \quad \tilde{\mathbf{L}}^* = \mathbf{\Psi}_q\mathbf{L}^T.$$

Thus, $\tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}} = \mathbf{\Psi}_q(\mathbf{L}^T\mathbf{E}\mathbf{U})\mathbf{\Psi}_p^*$ and so, $\tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$ is unitarily equivalent to $\mathbf{L}^T\mathbf{E}\mathbf{U}$, modulo its kernel and cokernel. More precisely, $\mathbf{L}^T\mathbf{E}\mathbf{U}$ is isomorphic to a linear transformation induced by $\tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$ viewed as a mapping of the quotient space, $\mathbb{C}^{N_p}/\mathsf{Ker}(\widetilde{\mathbf{U}})$ onto $\mathsf{Ran}(\widetilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}})$. This is a significant observation since the singular values of $\mathbf{L}^T\mathbf{E}\mathbf{U}$ are the *Hankel singular values* of the system which are system invariants (i.e., independent of system realization) and play a fundamental role in BT (see, Step 2 of Algorithm 1). Although $\mathbf{L}^T\mathbf{E}\mathbf{U}$ itself is evidently tied closely to knowledge of a state-space realization, the unitarily equivalent matrix, $\widetilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$ will provide essentially equivalent information while being directly derivable from data, as we will see.

Proposition 3.1. *Let $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{L}}$ be as defined in (15) and (17). Define the matrix $\widetilde{\mathbb{L}} = \widetilde{\boldsymbol{L}}^*\boldsymbol{E}\widetilde{\boldsymbol{U}} \in \mathbb{C}^{N_q\times N_p}$. Then, for $1 \leq k \leq N_q$ and $1 \leq j \leq N_p$,*

$$\widetilde{\mathbb{L}}_{k,j} = \begin{cases} -\phi_k\rho_j\dfrac{H(\imath\omega_k) - H(\imath\zeta_j)}{\imath\omega_k - \imath\zeta_j}, & \text{for } 1 \leq k \leq \ell_q,\ 1 \leq j \leq \ell_p, \\ \phi_k\rho_\infty H(\imath\omega_k), & \text{for } 1 \leq k \leq \ell_q = N_q,\ j = N_p = \ell_p + 1, \\ \phi_\infty\rho_j H(\imath\zeta_j), & \text{for } k = N_q = \ell_q + 1,\ 1 \leq j \leq \ell_p = N_p, \\ \phi_\infty\rho_\infty M_0, & \text{for } k = N_q = \ell_q + 1,\ j = N_p = \ell_p + 1. \end{cases} \tag{19}$$

Proof. Let $\mathbf{e}_k$ denote the kth canonical vector (of conforming length). Then, it follows from the definitions of $\widetilde{\mathbf{L}}$ and $\widetilde{\mathbf{U}}$ that, for $1 \leq k \leq \ell_q$ and $1 \leq j \leq \ell_p$, the following holds true

$$\begin{aligned} \widetilde{\mathbb{L}}_{k,j} &= \mathbf{e}_k^T\widetilde{\mathbb{L}}\mathbf{e}_j = (\mathbf{e}_k^T\widetilde{\mathbf{L}}^*)\mathbf{E}(\widetilde{\mathbf{U}}\mathbf{e}_j) = \phi_k\rho_j\mathbf{c}^T(\imath\omega_k\mathbf{E} - \mathbf{A})^{-1}\,\mathbf{E}\,(\imath\zeta_j\mathbf{E} - \mathbf{A})^{-1}\mathbf{b} \\ &= \frac{\phi_k\rho_j}{\imath\omega_k - i\zeta_j}\mathbf{c}^T(\imath\omega_k\mathbf{E} - \mathbf{A})^{-1}\left[(\imath\omega_k\mathbf{E} - \mathbf{A}) - (\imath\zeta_j\mathbf{E} - \mathbf{A})\right](\imath\zeta_j\mathbf{E} - \mathbf{A})^{-1}\mathbf{b} \\ &= -\frac{\phi_k\rho_j}{\imath\omega_k - \imath\zeta_j}\mathbf{c}^T\left[(\imath\omega_k\mathbf{E} - \mathbf{A})^{-1} - (\imath\zeta_j\mathbf{E} - \mathbf{A})^{-1}\right]\mathbf{b} = -\phi_k\rho_j\frac{H(\imath\omega_k) - H(\imath\zeta_j)}{\imath\omega_k - \imath\zeta_j}. \end{aligned}$$

Next, if $1 \leq k \leq \ell_q = N_q$, and $j = N_p = \ell_p + 1$ we can write that

$$\begin{aligned} \widetilde{\mathbb{L}}_{k,j} &= \mathbf{e}_k^T\widetilde{\mathbb{L}}\mathbf{e}_j = (\mathbf{e}_k^T\widetilde{\mathbf{L}}^*)\mathbf{E}(\widetilde{\mathbf{U}}\mathbf{e}_j) = \phi_k\mathbf{c}^T(\imath\omega_k\mathbf{E} - \mathbf{A})^{-1}\,\mathbf{E}\,\rho_\infty\mathbf{E}^{-1}\mathbf{b} \\ &= \phi_k\rho_\infty\mathbf{c}^T(\imath\omega_k\mathbf{E} - \mathbf{A})^{-1}\mathbf{b} = \phi_k\rho_\infty H(\imath\omega_k). \end{aligned}$$

Similarly, we find the exact derivation of $\widetilde{\mathbb{L}}_{k,j}$ in Proposition 3.1 for the case $k = N_q = \ell_q + 1$ and $1 \leq j \leq \ell_p = N_p$. Finally, for $k = N_q = \ell_q + 1$, $j = N_p = \ell_p + 1$, it follows that:

$$\begin{aligned} \widetilde{\mathbb{L}}_{k,j} &= \mathbf{e}_k^T\widetilde{\mathbb{L}}\mathbf{e}_j = (\mathbf{e}_k^T\widetilde{\mathbf{L}}^*)\mathbf{E}(\widetilde{\mathbf{U}}\mathbf{e}_j) = \phi_\infty\mathbf{c}^T\mathbf{E}^{-1}\,\mathbf{E}\,\rho_\infty\mathbf{E}^{-1}\mathbf{b} \\ &= \phi_\infty\rho_\infty\mathbf{c}^T\mathbf{E}^{-1}\mathbf{b} = \phi_\infty\rho_\infty M_0. \end{aligned}$$

□

The observation that the quadrature-based approximate quantity $\widetilde{\mathbb{L}} = \widetilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$ may be obtained solely from transfer function samples is consistent with it being a system invariant, producing approximations to Hankel singular values that are also system invariants and not dependent on any specific state-space realization. Note that $\widetilde{\mathbb{L}}$ has at most rank-n, i.e., the McMillan degree of the underlying system.

Of course, it will not be the case that the quadratures described in (14) and (16) are exact, so before we proceed we seek some assurance that the error induced by the quadrature approximation can be controlled. We have the following result:

Proposition 3.2. *Suppose the quadratures in (14) and (16) produce approximations $\widetilde{\boldsymbol{P}}$ and $\widetilde{\boldsymbol{Q}}$ to $\boldsymbol{P}$ and $\boldsymbol{Q}$, respectively, that satisfy $\|\boldsymbol{Q} - \widetilde{\boldsymbol{Q}}\|_F \leq \frac{\delta}{1+\delta}\sigma_{\min}(\boldsymbol{Q})$ and $\|\boldsymbol{P} - \widetilde{\boldsymbol{P}}\|_F \leq \frac{\delta}{1+\delta}\sigma_{\min}(\boldsymbol{P})$ for some*

$\delta \in (0,1)$ *where* $\sigma_{\min}(\,\cdot\,)$ *denotes the smallest singular value. Then there exist isometries* $\boldsymbol{\Psi}_p \in \mathbb{C}^{N_p \times n}$ *and* $\boldsymbol{\Psi}_q \in \mathbb{C}^{N_q \times n}$ *such that*

$$||\widetilde{\boldsymbol{L}}^* \boldsymbol{E}\widetilde{\boldsymbol{U}} - \boldsymbol{\Psi}_q(\boldsymbol{L}^T \boldsymbol{E}\boldsymbol{U})\boldsymbol{\Psi}_p^*||_F \le 2\ \delta\ ||\boldsymbol{E}||_2\,||\boldsymbol{L}||_2\,||\boldsymbol{U}||_2 \tag{20}$$

Let $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_n$ *denote the singular values of* $\boldsymbol{L}^T\boldsymbol{E}\boldsymbol{U}$ *(i.e., the Hankel singular values of the system* (1)*), and let* $\tilde{\sigma}_1 \ge \tilde{\sigma}_2 \ge \cdots \ge \tilde{\sigma}_n$ *denote the singular values of* $\widetilde{\boldsymbol{L}}^* \boldsymbol{E}\widetilde{\boldsymbol{U}}$*. Then,*

$$\left(\sum_{k=1}^{n} (\sigma_k - \tilde{\sigma}_k)^2\right)^{\frac{1}{2}} \le 2\ \delta\ ||\boldsymbol{E}||_2\,||\boldsymbol{L}||_2\,||\boldsymbol{U}||_2$$

The proof of this proposition may be found in Appendix B.

This leads us to consider a *quadrature-based* approximation to BT that uses $\widetilde{\mathbf{U}}$ and $\widetilde{\mathbf{L}}$ in lieu of $\mathbf{U}$ and $\mathbf{L}$ in Step 2 of Algorithm 1, replacing the SVD of $\mathbf{L}^T\mathbf{E}\mathbf{U}$ instead with the SVD of the (nearly) unitarily equivalent matrix, $\widetilde{\mathbb{L}} = \widetilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$ that can be computed *nonintrusively*, generated essentially solely from transfer function measurements.

Recall the singular values of $\mathbf{L}^T\mathbf{E}\mathbf{U}$ (the Hankel singular values) are independent of system realization. Our nonintrusive, quadrature-based approach provides a new way of accessing the Hankel singular values without recourse to a system realization.

3.1.1 Numerical examples

We compare in what follows the computed Hankel singular values of two dynamical systems, for which the matrix representation is available in [40], namely

- The heat equation (referred to as [heat]) describing the thermal response of a thin rod - the dimension of the LTI is $n = 200$ and it has single input and single output.
- The International Space Station example (referred to as [iss1r]) that models the component 1r of the International Space Station - the dimension of the LTI is $n = 270$ and it has three inputs and three outputs.

For more details on these models, we refer the reader to [18]. In Figs. 1 and 2, we depict the true Hankel singular values together with the approximate ones obtained via the non-intrusive quadrature-based approach as described above. We use two representative quadrature rules described in Appendix A, labeled as [ExpTrap] and [B/CC]. As the figures illustrate, the quadrature-based estimates for the Hankel singular values approximate the true values accurately.

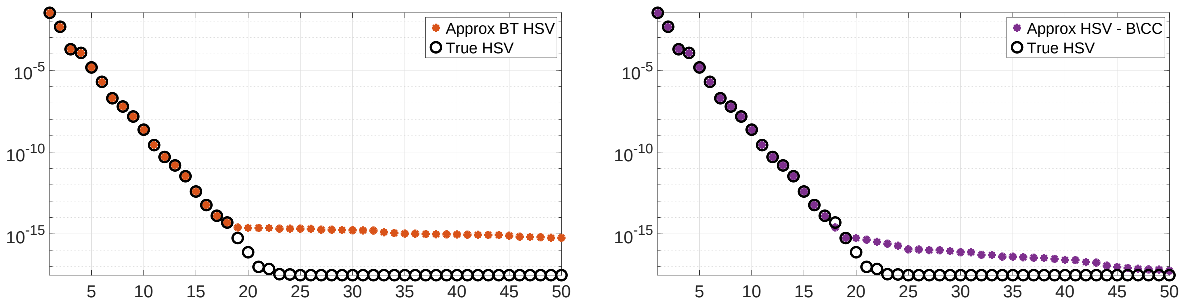

Figure 1: The computed (Hankel) singular values for the NICONET [heat] model using a trapezoid quadrature with 120 nodes in the interval $[10^{-3}, 10^{3}]\imath$ (left) and using 120 nodes with $L = 4$ and $L = 3$ in [B/CC] for $\widetilde{\mathbf{U}}$ and $\widetilde{\mathbf{L}}$, respectively (right).

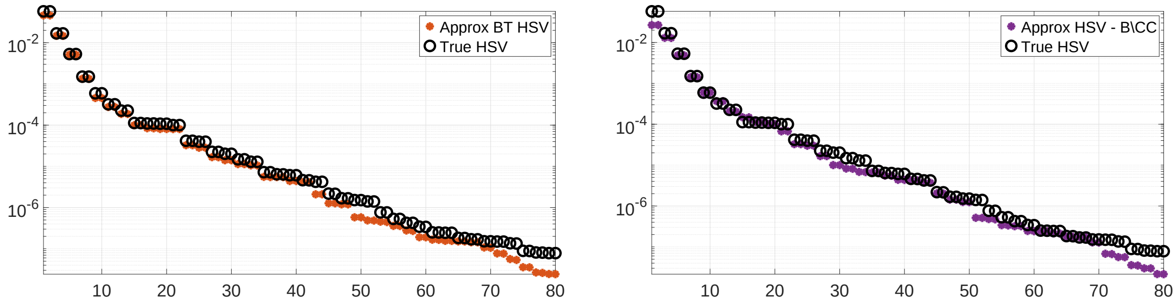

Figure 2: The computed (Hankel) singular values for the NICONET [iss1r] model using a trapezoid quadrature with 400 nodes in the interval $[10^{-1}, 10^{2}]\mathbb{i}$ (left) and using 400 nodes with $L = 10$ and $L = 10.5$ in [B/CC] for $\widetilde{\mathbf{U}}$ and $\tilde{\mathbf{L}}$, respectively (right).

3.2 Data-driven Balanced Truncation

Although BT yields reduced models in a manner that reflects system invariants and in that sense is independent of system realizations, BT typically does require explicit access to a state-space representation and thus, in a practical sense, remains tied to system realizations. In particular, note that in Step 3 of Algorithm 1, model reduction bases $\mathbf{W}_r$ and $\mathbf{V}_r$ are constructed and then used to project state space quantities. Through our (implicit) use of quadrature approximations for the reachability and observability Gramians, we are able to avoid any explicit reference to model reduction bases or to specific system realizations, and effect BT in a coordinate system determined solely by input-output data.

Notice that in Step 4 of Algorithm 1, we have $\mathbf{A}_r = \mathbf{S}_1^{-1/2}\mathbf{Z}_1^T\left(\mathbf{L}^T\mathbf{A}\mathbf{U}\right)\mathbf{Y}_1\mathbf{S}_1^{-1/2}$, and so the construction of $\mathbf{A}_r$ makes reference to both $\mathbf{L}^T\mathbf{A}\mathbf{U}$ and the SVD of $\mathbf{L}^T\mathbf{E}\mathbf{U}$, reflecting equivalent basis transformations on both $\mathbf{A}$ and $\mathbf{E}$. Proposition 3.1 showed how we can replace $\mathbf{L}^T\mathbf{E}\mathbf{U}$ with the (nearly) unitarily equivalent matrix, $\tilde{\mathbb{L}} = \tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$ (which could then be computed directly from data). So, we might expect that we could construct a realization equivalent to $\mathbf{A}_r$ by replicating on $\mathbf{A}$ an equivalent change of basis that had been visited upon $\mathbf{E}$ by $\tilde{\mathbb{L}} = \tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}}$. Pleasantly, we find this to be the case in our next result by examining $\widetilde{\mathbb{M}} = \tilde{\mathbf{L}}^*\mathbf{A}\widetilde{\mathbf{U}}$. Similar observations will hold for $\mathbf{b}_r$ and $\mathbf{c}_r$.

Proposition 3.3. *Let $\widetilde{\boldsymbol{U}}$ and $\tilde{\boldsymbol{L}}$ be as defined in (15) and (17). Define the matrix $\widetilde{\mathbb{M}} = \tilde{\boldsymbol{L}}^*\boldsymbol{A}\widetilde{\boldsymbol{U}} \in \mathbb{C}^{N_q\times N_p}$. Then, the (k,j) entry of $\widetilde{\mathbb{M}}$, for $1 \le k \le N_q$ and $1 \le j \le N_p$, is given by*

$$\widetilde{\mathbb{M}}_{k,j} = \begin{cases} -\phi_k\rho_j\dfrac{\mathbb{i}\omega_k H(\mathbb{i}\omega_k) - \mathbb{i}\zeta_j H(\mathbb{i}\zeta_j)}{\mathbb{i}\omega_k - \mathbb{i}\zeta_j}, & \textit{for } 1 \le k \le \ell_q,\ 1 \le j \le \ell_p, \\ \phi_k\rho_\infty\left(\mathbb{i}\omega_k H(\mathbb{i}\omega_k) - M_0\right), & \textit{for } 1 \le k \le \ell_q = N_q,\ j = N_p = \ell_p + 1, \\ \phi_\infty\rho_j\left(\mathbb{i}\zeta_j H(\mathbb{i}\zeta_j) - M_0\right), & \textit{for } k = N_q = \ell_q + 1,\ 1 \le j \le \ell_p = N_p, \\ \phi_\infty\rho_\infty M_1, & \textit{for } k = N_q = \ell_q + 1,\ j = N_p = \ell_p + 1. \end{cases} \tag{21}$$

Likewise, defining $\tilde{\mathbb{h}} = \tilde{\boldsymbol{L}}^*\boldsymbol{b} \in \mathbb{C}^{N_q\times 1}$ and $\tilde{\mathbf{g}}^T = \boldsymbol{c}^T\widetilde{\boldsymbol{U}} \in \mathbb{C}^{1\times N_p}$, we find component-wise,

$$\tilde{\mathbb{h}}_k = \begin{cases} \phi_k H(\mathbb{i}\omega_k), & \textit{for } 1 \le k \le \ell_q, \\ \phi_\infty M_0, & \textit{for } k = N_q = \ell_q + 1. \end{cases} \quad \textit{and} \tag{22}$$

$$\tilde{\mathbf{g}}_j = \begin{cases} \rho_j H(\mathbb{i}\zeta_j), & \textit{for } 1 \le j \le N_p, \\ \rho_\infty M_0, & \textit{for } j = N_p = \ell_p + 1. \end{cases} \tag{23}$$

Proof. The proof is similar to that of Proposition 3.1. Notice first that for any square matrix, $\mathbf{T}$, and any ς, $\mu \in \mathbb{C}$ that are not eigenvalues of $\mathbf{T}$, one may verify

$$\varsigma(\varsigma\mathbf{I}-\mathbf{T})^{-1} - \mu(\mu\mathbf{I}-\mathbf{T})^{-1} = -(\varsigma-\mu)(\varsigma\mathbf{I}-\mathbf{T})^{-1}\mathbf{T}(\mu\mathbf{I}-\mathbf{T})^{-1}.$$

By setting $\mathbf{T} = \mathbf{A}\mathbf{E}^{-1}$ and simplifying, one finds

$$\varsigma(\varsigma\mathbf{E}-\mathbf{A})^{-1} - \mu(\mu\mathbf{E}-\mathbf{A})^{-1} = -(\varsigma-\mu)(\varsigma\mathbf{E}-\mathbf{A})^{-1}\mathbf{A}(\mu\mathbf{E}-\mathbf{A})^{-1}.$$

Now, using the definitions of $\tilde{\mathbf{L}}$ and $\widetilde{\mathbf{U}}$,

$$\begin{aligned}
\widetilde{\mathbb{M}}_{k,j} &= \mathbf{e}_k^T\widetilde{\mathbb{M}}\mathbf{e}_j = (\mathbf{e}_k^T\tilde{\mathbf{L}}^*)\mathbf{A}(\widetilde{\mathbf{U}}\mathbf{e}_j) \\
&= \phi_k\rho_j\mathbf{c}^T(\imath\omega_k\mathbf{E}-\mathbf{A})^{-1}\mathbf{A}(\imath\zeta_j\mathbf{E}-\mathbf{A})^{-1}\mathbf{b} \\
&= -\frac{\phi_k\rho_j}{\imath\omega_k-\imath\zeta_j}\mathbf{c}^T\left[\imath\omega_k(\imath\omega_k\mathbf{E}-\mathbf{A})^{-1} - \imath\zeta_j(\imath\zeta_j\mathbf{E}-\mathbf{A})^{-1}\right]\mathbf{b} \\
&= -\phi_k\rho_j\frac{\imath\omega_k H(\imath\omega_k) - \imath\zeta_j H(\imath\zeta_j)}{\imath\omega_k-\imath\zeta_j}.
\end{aligned}$$

Next, if $1 \le k \le \ell_q = N_q$, and $j = N_p = \ell_p + 1$ we can write that

$$\begin{aligned}
\widetilde{\mathbb{M}}_{k,j} &= \mathbf{e}_k^T\widetilde{\mathbb{M}}\mathbf{e}_j = (\mathbf{e}_k^T\tilde{\mathbf{L}}^*)\mathbf{A}(\widetilde{\mathbf{U}}\mathbf{e}_j) = \phi_k\mathbf{c}^T(\imath\omega_k\mathbf{E}-\mathbf{A})^{-1}\mathbf{A}\rho_\infty\mathbf{E}^{-1}\mathbf{b} \\
&= \phi_k\rho_\infty\mathbf{c}^T\left[\imath\omega_k(i\omega_k\mathbf{E}-\mathbf{A})^{-1} - \mathbf{E}^{-1}\right]\mathbf{b} = \phi_k\rho_\infty\left(\imath\omega_k H(\imath\omega_k) - M_0\right).
\end{aligned}$$

Similarly, we can find the exact derivation of $\tilde{\mathbb{L}}_{k,j}$ in (21) for the case $k = N_q = \ell_q + 1,\ 1 \le j \le \ell_p = N_p$. Next, for $k = N_q = \ell_q + 1,\ j = N_p = \ell_p + 1$, it follows that:

$$\begin{aligned}
\widetilde{\mathbb{M}}_{k,j} &= \mathbf{e}_k^T\widetilde{\mathbb{M}}\mathbf{e}_j = (\mathbf{e}_k^T\tilde{\mathbf{L}}^*)\mathbf{A}(\widetilde{\mathbf{U}}\mathbf{e}_j) = \phi_\infty\mathbf{c}^T\mathbf{E}^{-1}\,\mathbf{A}\,\rho_\infty\mathbf{E}^{-1}\mathbf{b} \\
&= \phi_\infty\rho_\infty\mathbf{c}^T\mathbf{E}^{-1}\mathbf{A}\mathbf{E}^{-1}\mathbf{b} = \phi_\infty\rho_\infty M_1.
\end{aligned}$$

Finally, one can write

$$\tilde{\mathbb{h}}_k = \mathbf{e}_k^T\tilde{\mathbb{h}} = (\mathbf{e}_k^T\tilde{\mathbf{L}})\mathbf{b} = \begin{cases}\phi_k\mathbf{c}^T(\imath\omega_k\mathbf{E}-\mathbf{A})^{-1}\mathbf{b} = \phi_k H(\imath\omega_k), & \text{for } 1 \le k \le \ell_q, \\ \phi_\infty\mathbf{c}^T\mathbf{E}^{-1}\mathbf{b} = \phi_\infty M_0, & \text{for } k = N_q = \ell_q + 1.\end{cases}$$

and

$$\tilde{\mathbb{g}}_j = \tilde{\mathbb{g}}^T\mathbf{e}_j = \mathbf{c}^T(\widetilde{\mathbf{U}}\mathbf{e}_j) = \begin{cases}\rho_j\mathbf{c}^T(\imath\zeta_j\mathbf{E}-\mathbf{A})^{-1}\mathbf{b} = \rho_j H(\imath\zeta_j), & \text{for } 1 \le j \le N_p, \\ \rho_\infty\mathbf{c}^T\mathbf{E}^{-1}\mathbf{b} = \rho_\infty M_0, & \text{for } j = N_p = \ell_p + 1.\end{cases}$$

□

At this point, we are able to replace the quantities $\mathbf{L}^T\mathbf{E}\mathbf{U}$, $\mathbf{L}^T\mathbf{A}\mathbf{U}$, $\mathbf{L}^T\mathbf{b}$, and $\mathbf{c}^T\mathbf{U}$ in Algorithm 1 with equivalent quantities that are derivable directly from data (i.e., from transfer function samples). We assemble this together in Algorithm 2, yielding a computationally feasible strategy for approximate balanced truncation (QuadBT), using only transfer function sampling without access to internal properties, yet still capable of recovering to any accuracy desired the same (or state-space equivalent) reduced models as BT.

It is revealing to contrast the construction of $\mathbf{A}_r$ in (11) with the construction of $\widetilde{\mathbf{A}}_r$ in (26). While the former computes the inner term $\mathbf{L}^T\mathbf{A}\mathbf{U}$ and requires access to internal quantities characterizing the dynamics, the latter approximates this term through $\widetilde{\mathbb{M}}$, which may be constructed directly from transfer function samples. Moreover, the quantities $\tilde{\mathbf{S}}_1^{-1/2}\tilde{\mathbf{Z}}_1^*$ and $\widetilde{\mathbf{Y}}_1\tilde{\mathbf{S}}_1^{-1/2}$ are available from the SVD of the data matrix $\tilde{\mathbb{L}}$ in (19).

Algorithm 2 Quadrature-based (data-driven) balanced truncation (QuadBT)

```
Input: LTI system described through a transfer function evaluation map, $H(s)$;
    quadrature nodes, $\zeta_j$, and weights, $\rho_j$, for $j = 1, 2, \ldots, N_p$;
    quadrature nodes, $\omega_k$, and weights, $\phi_k$, for $k = 1, 2, \ldots, N_q$,
    and a truncation index, $1 \leq r \leq \min(N_p, N_q)$.
Output: QuadBT reduced-system: $\widetilde{\mathbf{A}}_r \in \mathbb{R}^{r\times r}$, $\tilde{\mathbf{b}}_r, \tilde{\mathbf{c}}_r \in \mathbb{R}^r$.
 1: Sample transfer function values, $\{H(\mathbb{i}\zeta_j)\}_{j=1}^{N_p}$ and $\{H(\mathbb{i}\omega_k)\}_{k=1}^{N_q}$. Using the samples and quadrature
    weights, $\{\rho_j\}$ and $\{\phi_k\}$, construct $\widetilde{\mathbb{L}} \in \mathbb{C}^{N_q\times N_p}$, $\widetilde{\mathbb{M}} \in \mathbb{C}^{N_q\times N_p}$, $\widetilde{\mathbb{h}}$, and $\tilde{\mathbf{g}}$ as in (19), (21), (22), and
    (23), respectively,
 2: Compute the SVD of $\widetilde{\mathbb{L}}$:
    $$\widetilde{\mathbb{L}} = \begin{bmatrix}\widetilde{\mathbf{Z}}_1 & \widetilde{\mathbf{Z}}_2\end{bmatrix}\begin{bmatrix}\widetilde{\mathbf{S}}_1 & \\ & \widetilde{\mathbf{S}}_2\end{bmatrix}\begin{bmatrix}\widetilde{\mathbf{Y}}_1^* \\ \widetilde{\mathbf{Y}}_2^*\end{bmatrix}, \qquad (24)$$
    where $\widetilde{\mathbf{S}}_1 \in \mathbb{R}^{r\times r}$ and $\widetilde{\mathbf{S}}_2 \in \mathbb{R}^{(N_q-r)\times(N_p-r)}$.
 3: Construct the reduced order matrices:
    $$\widetilde{\mathbf{E}}_r = \widetilde{\mathbf{S}}_1^{-1/2}\widetilde{\mathbf{Z}}_1^*\ \widetilde{\mathbb{L}}\ \widetilde{\mathbf{Y}}_1\widetilde{\mathbf{S}}_1^{-1/2} = \mathbf{I}_r, \qquad (25)$$
    $$\widetilde{\mathbf{A}}_r = \widetilde{\mathbf{S}}_1^{-1/2}\widetilde{\mathbf{Z}}_1^*\ \widetilde{\mathbb{M}}\ \widetilde{\mathbf{Y}}_1\widetilde{\mathbf{S}}_1^{-1/2}, \qquad (26)$$
    $$\tilde{\mathbf{b}}_r = \widetilde{\mathbf{S}}_1^{-1/2}\widetilde{\mathbf{Z}}_1^*\widetilde{\mathbb{h}}, \quad \text{and} \qquad (27)$$
    $$\tilde{\mathbf{c}}_r = \tilde{\mathbf{g}}^T\widetilde{\mathbf{Y}}_1\widetilde{\mathbf{S}}_1^{-1/2}. \qquad (28)$$
```

Remark 3.1. *In the analysis above, we have assumed that the quadrature nodes $\{\zeta_j\}$ for $\widetilde{\boldsymbol{P}}$ and $\{\omega_k\}$ for $\widetilde{\boldsymbol{Q}}$ are distinct from each other. However, it may be more convenient to choose the same quadrature nodes (and weights) for both $\widetilde{\boldsymbol{P}}$ and $\widetilde{\boldsymbol{Q}}$. This leads to only minor changes to the analysis and in Algorithm 2. The vectors $\widetilde{\mathbb{h}}$ in (22) and $\tilde{\mathbf{g}}$ in (23) stay unchanged. The principal change occurs on the diagonal entries of $\widetilde{\mathbb{L}}$ and $\widetilde{\mathbb{M}}$: assuming that $\omega_k = \zeta_k$, then the (k,k) entry of $\widetilde{\mathbb{L}}$ in (19) is replaced by*

$$\widetilde{\mathbb{L}}_{k,k} = -\phi_k\rho_k H'(\mathbb{i}\omega_k),$$

where H' denotes the derivative of H. The (k,k) entry of $\widetilde{\mathbb{M}}$ in (21) is replaced by

$$\widetilde{\mathbb{M}}_{k,k} = -\phi_k\rho_k\left(\omega_k H'(\mathbb{i}\omega_k) + H(\mathbb{i}\omega_k)\right).$$

The remaining entries in $\widetilde{\mathbb{L}}$ and $\widetilde{\mathbb{M}}$ are unchanged.

3.2.1 Numerical examples - the SISO case

In this section we will test the approximation capability of our newly proposed method and compare it with that of classical balanced truncation.

1. [BT-classic]: The classical BT approach (Algorithm 1).

2. [ExpTrap-N]: The new method QuadBT (Algorithm 2) computed using the exponential trapezoid rule with N points (without derivative samples); more details in Appendix A.

3. [B/CC-N]: The new method QuadBT (Algorithm 2) computed using Boyd/ Clenshaw-Curtis rule with N points (without derivative samples); more details in Appendix A.

First, we start with the [heat] numerical test case. Choose both 60 and 120 logarithmically-spaced points in the interval $[10^{-3}, 10^{3}]\imath$ as nodes for [ExpTrap] (the left and right nodes are different). In [B/CC], choose $L = 4$ and $L = 3$ for $\widetilde{\mathbf{U}}$ and $\tilde{\mathbf{L}}$, respectively, with both 60 and 120 nodes. Then, we vary the reduction order from $r = 2$ to $r = 14$ in increments of two and collect both the $\mathcal{H}_\infty$ and the $\mathcal{H}_2$ norms of the error systems corresponding to [BT-classic] and both quadrature schemes. The results are presented in 4, illustrating that both QuadBT models (with [ExpTrap-N] and [B/CC-N]) almost exactly replicate the performance of [BT-classic].

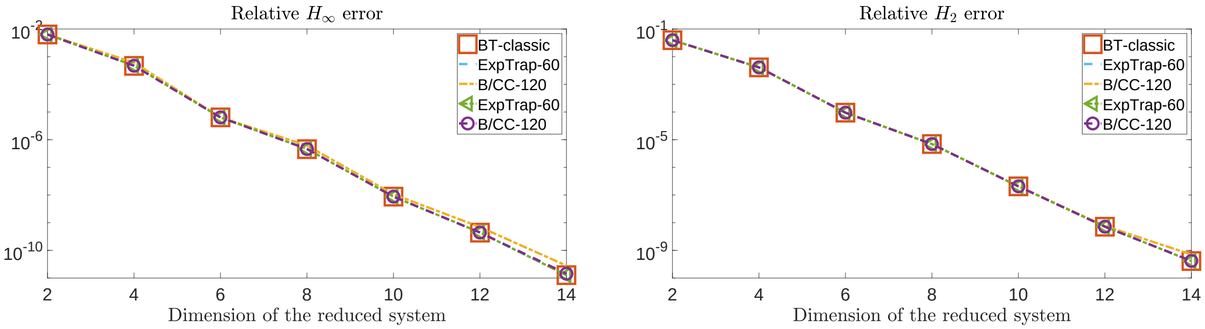

Figure 3: The relative $\mathcal{H}_\infty$ (left) and $\mathcal{H}_2$ (right) error approximation for different quadratures and nodes applied to the Niconet [heat] model.

We continue with the [iss1r] numerical test case. To enforce a SISO system, we restrict our attention to the first input and first output. Then, choose either 200 or 400 points logarithmically-spaced points in the interval $[10^{-1}, 10^{2}]\imath$ as nodes for the [ExpTrap] scheme (the left and right nodes are different). In [B/CC], we choose $L = 10$ and $L = 9$ for $\widetilde{\mathbf{U}}$ and $\tilde{\mathbf{L}}$, respectively and use both 200 and 400 nodes. We then vary the reduction order in the interval $[2, 24]$ (in increments of two) and collect both the $\mathcal{H}_\infty$ norm and also the $\mathcal{H}_2$ norm of the error systems corresponding to all three reduced models. The results are presented in Fig. 4. As in the [heat] case, these figures show that transfer-function based QuadBT very closely mimics (in many cases exactly replicates) the $\mathcal{H}_\infty$ and $\mathcal{H}_2$ performance of the projection-based BT. In this example, [B/CC] quadrature outperforms [ExpTrap].

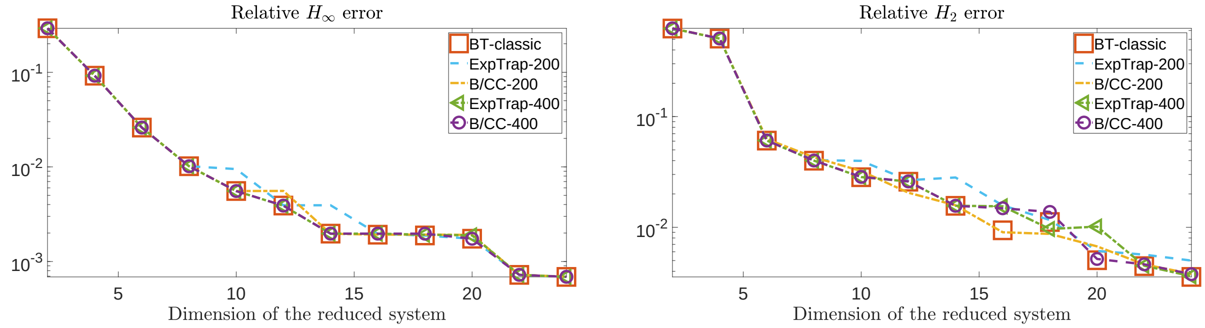

Figure 4: The relative $\mathcal{H}_\infty$ (left) and $\mathcal{H}_2$ (right) error approximation for different quadratures and nodes applied to the Niconet [iss1r] model.

3.3 Connections to approximate BT methods

Due to its compelling theoretical properties, an immense amount of literature has developed around effective methods for approximating BT for large-scale dynamical systems, often focusing on efficient handling of the main computational bottleneck in BT, namely, the computation of Gramians, which requires the solution of two large-scale Lyapunov equations (5)-(6). We list a small selection of these works here [11, 34, 35, 44, 54, 57]; further references and detail can be found within these resources.

A common approach for approximate BT involves constructing low-rank approximate factors to $\mathbf{P}$ and $\mathbf{Q}$, i.e., $\mathbf{P} \approx \widehat{\mathbf{U}}\widehat{\mathbf{U}}^T$ and $\mathbf{Q} \approx \widehat{\mathbf{L}}\widehat{\mathbf{L}}^T$, with $\mathsf{rank}(\widehat{\mathbf{U}})$ and $\mathsf{rank}(\widehat{\mathbf{L}})$ much smaller than n. Then we use $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{L}}$ in Algorithm 1 in place of the true square-root factors $\mathbf{U}$ and $\mathbf{L}$. Unfortunately, regardless of how $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{L}}$ are computed, this construction requires access to internal dynamics and state-space quantities. The reduced model is then obtained by an explicit projection as in (10)-(13). This is fundamentally different from what we develop in this work; we do not compute $\widehat{\mathbf{U}}$ or $\widehat{\mathbf{L}}$ and we do not require access to state-space quantities. We only assume access to transfer function samples. Despite these major contrasts, there are striking similarities that motivate the underlying approximation theory.

The quadrature approximation (14) to $\mathbf{P}$ has been a major force behind the frequency-domain version of Balanced POD [61]. Indeed, the Balanced POD approach of [61] engages a numerical quadrature for the Gramians, $\mathbf{P}$ and $\mathbf{Q}$, using unit quadrature weights; $\rho_j = 1$ for $j = 1, \ldots, \ell_p$ (14) and $\phi_k = 1$ for $j = 1, \ldots, \ell_q$ in (16), and omitting the node at infinity in both cases. The approach of [61] was later referred to as Poor Man's Truncated Balanced Reduction in [45]; however, the implementation there is restricted to symmetric systems. Significantly, Balanced POD is not data-driven in the sense that we explore here, but rather it is both *projection-based* and *intrusive*, requiring explicit access to a state-space representation. Nonetheless, it is clear that one may immediately arrive at a transfer-function based *data-driven* formulation of Balanced POD by using the QuadBT framework developed here.

Along similar lines, Gauss-Kronrod quadrature was used to approximate the Hankel singular values in [12], but as with earlier methods, the use of explicit state-space representations is still required. Similar quadrature strategies were later used in [17] to extend BT to structure-preserving model reduction for integro-differential equations. In all these methods, numerical quadrature in the frequency domain to approximate the Gramians can be viewed as a common starting point for our framework. Significantly, we have avoided an explicit projection strategy by directly working with transfer function evaluations.

Evidently, numerical quadrature can also be applied to the time-domain representation of $\mathbf{P}$ as in (2) and this has also been considered as a strategy for approximate BT, see, e.g., [13, 14, 27, 38, 41, 55]. We explore these connections in §5 where a time-domain formulation of our data-driven BT framework is considered.

ADI-type methods (see, e.g.,[9, 11, 21, 35, 44, 54, 56]) are among the most common tools in use for producing approximate low-rank Gramian factors $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{L}}$. These methods are iterative in character and depend on ADI-shift parameters whose choice determines the speed of convergence. Even though the resulting low-rank factor $\widehat{\mathbf{U}}$ obtained via ADI methods will have a different structure than $\widetilde{\mathbf{U}}$ in (15), one may view either result as a low-rank factor for frequency-domain Balanced POD. Both factors span the same (Krylov) subspaces provided that the ADI parameters are chosen to be the negative of the quadrature nodes [5]. Significantly, low-rank ADI solutions have also been connected to time-domain quadrature methods [13, 14]. These connections highlight the powerful approximation-theoretic framework unifying these approaches and that is the reason we choose to emphasize it. Nonetheless, our approach is distinct from other methods in the sense that it never constructs approximate solutions to $\mathbf{P}$ or $\mathbf{Q}$, nor attempts to approximate low-rank factors). Our approximate BT reduced model is constructed directly from data.

3.4 Comparison with interpolatory Loewner model reduction

For the sampling points $\{\mu_k\}_{k=1}^{N_q}$ and $\{\lambda_j\}_{j=1}^{N_p}$, consider $\mathbb{L} \in \mathbb{C}^{N_q \times N_p}$, $\mathbb{M} \in \mathbb{C}^{N_q \times N_p}$, defined elementwise as:

$$\mathbb{L}_{k,j} = -\frac{H(\imath\mu_k) - H(\imath\lambda_j)}{\imath\mu_k - \imath\lambda_j} \quad \text{and} \quad \mathbb{M}_{k,j} = -\frac{\imath\mu_k H(\imath\mu_k) - \imath\lambda_j H(\imath\lambda_j)}{\imath\mu_k - \imath\lambda_j}, \qquad (29)$$
$$\text{for} \quad k = 1, 2, \ldots, N_q \quad \text{and} \quad j = 1, 2, \ldots, N_p.$$

The matrices $\mathbb{L}$ and $\mathbb{M}$ in (29) are called, respectively, the Loewner and shifted-Loewner matrices. Notice that $\mathbb{L}$ and $\mathbb{M}$ bear evident similarities to the key matrices $\widetilde{\mathbb{L}}$ in (19) and $\widetilde{\mathbb{M}}$ in (21) that are central to QuadBT. Indeed, assuming that the node at infinity is absent from the quadrature rules in (14) and (16), and that the quadrature nodes in QuadBT determine $\mu_k = i\omega_k$ for $k = 1, 2, \ldots, N_q$ and $\lambda_j = i\zeta_j$ for $j = 1, 2, \ldots, N_p$ in (29), the matrices $\widetilde{\mathbb{L}}$ in (19) and $\widetilde{\mathbb{M}}$ in (21) used in QuadBT will be diagonal scalings of $\mathbb{L}$ and $\mathbb{M}$:

$$\widetilde{\mathbb{L}} = \mathsf{diag}(\phi_1, \ldots, \phi_{N_q})\, \mathbb{L}\, \mathsf{diag}(\rho_1, \ldots, \rho_{N_p}),$$

and

$$\widetilde{\mathbb{M}} = \mathsf{diag}(\phi_1, \ldots, \phi_{N_q})\, \mathbb{M}\, \mathsf{diag}(\rho_1, \ldots, \rho_{N_p}).$$

Despite these equivalences, the matrices $\mathbb{L}$ and $\mathbb{M}$ of (29) were originally conceived to play a role in quite a different model reduction scheme which we refer to as the *interpolatory Loewner model reduction framework*. This model reduction framework was introduced in [37] as a data-driven interpolation-based system identification and complexity reduction technique. It has become a popular and powerful tool for model reduction over the last decade, and has been successfully extended to parametrized linear dynamical systems [30], structured linearized systems [52], bilinear [3] and quadratic-bilinear systems [23], and has been formulated to work with time-domain data as well [42].

The principal feature of the *interpolatory Loewner framework* is the construction of a rational interpolant matching the given sampled transfer function data. This provides immediately a realization of a reduced model that matches the system response at specified driving frequencies (or equivalently, at specified complex interpolation points). The quality of the final reduced model is strongly tied to the choice of interpolation points and in the MIMO setting, also to the choice of tangent directions. If some additional flexibility exists in where the system may be sampled, $\mathcal{H}_2$-optimal reduced order interpolants may be feasibly computed [6]. For a detailed account of theoretical developments as well as implementation details for Loewner-based methods relating especially to rational interpolation, we refer the reader to [2,4], and the references therein.

Although our QuadBT framework makes use of matrices that are, at least in many cases of interest, diagonally scaled Loewner and shifted Loewner matrices, rational interpolation plays no role for us. Instead, we have observed that there are strong links among (a) quadrature rules (14) and (16), (b) the (limiting) implicit Gramian factorizations described in (18), and (c) the related Loewner / shifted Loewner factorizations. These links couple together the features developed in Propositions 3.1, 3.2, and 3.3, and allow us ultimately to mimic BT in a purely data-driven manner.

3.4.1 Numerical examples

We use the [heat] and [$\mathsf{iss1r}$] models from §3.1.1 as test cases. QuadBT is implemented using only the [$\mathsf{ExpTrap}$] quadrature.

We choose 120 logarithmically-spaced points in the interval $[10^{-3}, 10^3]\imath$ for the [heat] model and 400 logarithmically-spaced points in the interval $[10^{-1}, 10^2]\imath$ for the [$\mathsf{iss1r}$] model. In the interpolatory Loewner framework, the singular values of $\mathbb{L}$ and/or the augmented matrix $[\ \mathbb{L}\ \ \mathbb{M}\ ]$ play a fundamental role in determining reduction order, analogous to Hankel singular values in BT; see [2, §4.3]. Therefore, in addition to the QuadBT quantities, we construct $\mathbb{L}$ and $\mathbb{M}$ corresponding to the same transfer function samples and compute (i) the true Hankel singular values, (ii) QuadBT-based Hankel singular values, and (iii) the singular values of $\mathbb{L}$ and $[\ \mathbb{L}\ \ \mathbb{M}\ ]$. Results are depicted in Fig. 5. While the Hankel singular values computed via [$\mathsf{ExpTrap}$] approximate the true ones fairly well, the singular values corresponding to the Loewner method appear to follow the trend, only up to a scaling factor. This simple example shows the importance of the quadrature weights appearing in the QuadBT framework.

Next, we consider the SISO version of the [$\mathsf{iss1r}$] model and apply both QuadBT with [$\mathsf{ExpTrap}$] using 200 logarithmically-spaced points in the interval $[10^{-1}, 10^2]\imath$ and the Loewner-based interpolation method. We vary the reduction order from $r = 2$ to $r = 24$, and collect the $\mathcal{H}_\infty$ and the $\mathcal{H}_2$ norms of

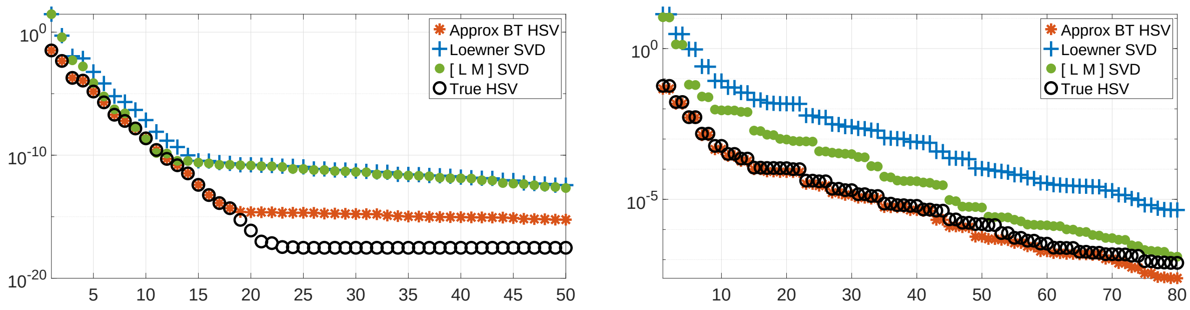

Figure 5: The computed singular values for the heat model (left) and the [iss1r] model (right).

the error systems for both methods. The results are presented in Fig. 6 where the label [Loewner-200] refers to the Loewner method with $N = 200$ points. Fig. 6 shows that *for the same transfer function data*, it pays off to use appropriate quadrature weights to perform QuadBT as QuadBT outperforms the Loewner approach and mimics BT much more closely.

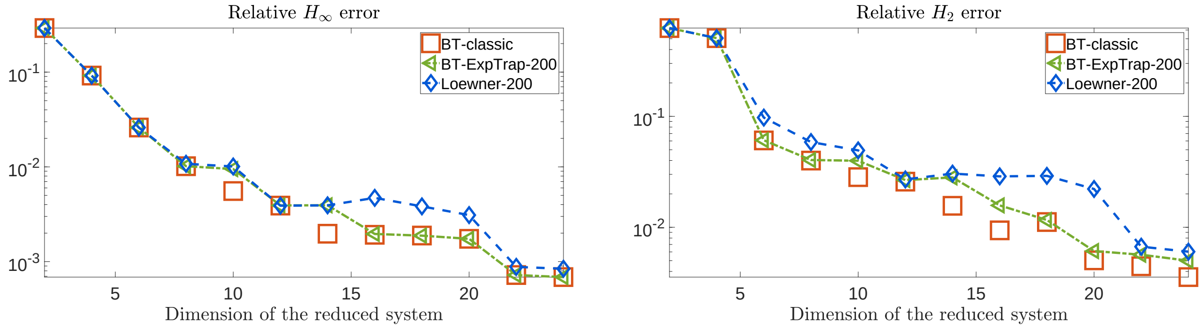

Figure 6: Relative $\mathcal{H}_\infty$ (left) and $\mathcal{H}_2$ (right) approximation errors for different methods applied to the [iss1r] model.

4 Refinements and Extensions

4.1 Keeping it real

The original system has been presumed to be real, insofar as real-valued inputs and real-valued initial states assure real-valued outputs. This guarantees in turn that a *real* realization involving matrices $\mathbf{E}$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$ is possible, though not necessarily explicitly available. Equivalently, the associated system transfer function $\mathsf{H}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}$ displays a conjugate symmetry of system poles and residues (i.e., complex poles and residues must occur in conjugate pairs). If we utilize *symmetric quadrature rules* in the approximations given in (14) and (16) — that is, if the sets of quadrature nodes and associated weights, $\{(\zeta_j, \rho_j)\}_{j=1}^{N_p}$ and $\{(\omega_k, \phi_k\}_{k=1}^{N_q}$, are symmetrically distributed on the real axis with respect to 0 — then we will be able to retain this conjugate symmetry, accomplish computations equivalent to Steps 2, 3, and 4 of QuadBT using only real arithmetic, and be assured that the final reduced model will inherit the conjugate symmetry of the original system, thus representing a reduced-order real dynamical system.

For simplicity, we return to the SISO case and suppose that both node sets have an even number of points that are symmetrically distributed across the real axis, say $N_p = 2\nu_p$ and $N_q = 2\nu_q$. (More general settings are considered in [2], p. 203). Relabel these node sets as

$$\zeta_{-\nu_p} < \zeta_{-\nu_p+1} < \cdots < \zeta_{-1} < 0 < \zeta_1 < \cdots < \zeta_{\nu_p-1} < \zeta_{\nu_p},$$
$$\omega_{-\nu_q} < \omega_{-\nu_q+1} < \cdots < \omega_{-1} < 0 < \omega_1 < \cdots < \omega_{\nu_q-1} < \omega_{\nu_q},$$

noting that $\zeta_{-j} = -\zeta_j$ and for the corresponding weights, $\rho_{-j} = \rho_j$, for $j = 1, ..., \nu_p$. Similarly, we have $\omega_{-k} = -\omega_k$ and $\phi_{-k} = \phi_k$ for $k = 1, ..., \nu_q$. The nodes/weights are then reordered with respect to this ± node pairing:

$$\{\zeta_1, \zeta_{-1}, \zeta_2, \zeta_{-2}, \dots, \zeta_{\nu_p}, \zeta_{-\nu_p}\} \quad \text{and} \quad \{\omega_1, \omega_{-1}, \omega_2, \omega_{-2}, \dots, \omega_{\nu_q}, \omega_{-\nu_q}\}. \tag{30}$$

When the samples obtained in Step 1 of QuadBT are reordered in the same way, they appear now as a sequence of conjugate pairs:

$$\{H(\boldsymbol{\imath}\zeta_j), \overline{H(\boldsymbol{\imath}\zeta_j)}\}_{j=1}^{\nu_p} \quad \text{and} \quad \{H(\boldsymbol{\imath}\omega_k), \overline{H(\boldsymbol{\imath}\omega_k)}\}_{k=1}^{\nu_q}.$$

Notice that one need only explicitly evaluate the first function of each pair.

Now instead of (19), we reconsider the construction of $\tilde{\mathbb{L}}$ taking into account the new node ordering given in (30). With respect to this node ordering, $\tilde{\mathbb{L}}$ can be partitioned into 2×2 blocks, $\tilde{\mathbb{L}}_{k,j}^{(2)}$, conforming to the conjugate pairing of quadrature nodes in (30) — left data at $(\boldsymbol{\imath}\omega_k, -\boldsymbol{\imath}\omega_k)$; right data at $(\boldsymbol{\imath}\zeta_j, -\boldsymbol{\imath}\zeta_j)$:

$$\tilde{\mathbb{L}}_{k,j}^{(2)} = -\phi_k\rho_j \begin{bmatrix} \frac{H(\boldsymbol{\imath}\,\omega_k) - H(\boldsymbol{\imath}\,\zeta_j)}{\boldsymbol{\imath}\,\omega_k - \boldsymbol{\imath}\,\zeta_j} & \frac{H(\boldsymbol{\imath}\,\omega_k) - \overline{H(\boldsymbol{\imath}\,\zeta_j)}}{\boldsymbol{\imath}\,\omega_k - (-\boldsymbol{\imath}\,\zeta_j)} \\ \frac{\overline{H(\boldsymbol{\imath}\,\omega_k)} - H(\boldsymbol{\imath}\,\zeta_j)}{(-\boldsymbol{\imath}\,\omega_k) - \boldsymbol{\imath}\,\zeta_j} & \frac{\overline{H(\boldsymbol{\imath}\,\omega_k)} - \overline{H(\boldsymbol{\imath}\,\zeta_j)}}{(-\boldsymbol{\imath}\,\omega_k) - (-\boldsymbol{\imath}\,\zeta_j)} \end{bmatrix},$$

for $j = 1, ..., \nu_p$ and $k = 1, ..., \nu_q$. Note that each $\tilde{\mathbb{L}}_{k,j}^{(2)}$ satisfies a Sylvester equation:

$$\begin{aligned} &\tilde{\mathbb{L}}_{k,j}^{(2)} \begin{bmatrix} \boldsymbol{\imath}\,\omega_k & 0 \\ 0 & -\boldsymbol{\imath}\,\omega_k \end{bmatrix} - \begin{bmatrix} \boldsymbol{\imath}\,\zeta_j & 0 \\ 0 & -\boldsymbol{\imath}\,\zeta_j \end{bmatrix} \tilde{\mathbb{L}}_{k,j}^{(2)} \\ &= \phi_k\rho_j \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} H(\boldsymbol{\imath}\,\zeta_j) & \overline{H(\boldsymbol{\imath}\,\zeta_j)} \end{bmatrix} - \begin{bmatrix} H(\boldsymbol{\imath}\,\omega_k) \\ \overline{H(\boldsymbol{\imath}\,\omega_k)} \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \right). \end{aligned}$$

This uniquely determines each $\tilde{\mathbb{L}}_{k,j}^{(2)}$ (and hence, $\tilde{\mathbb{L}}$) when the sets of quadrature nodes, $\{\zeta_j\}$ and $\{\omega_k\}$, are disjoint as we have assumed.

$\tilde{\mathbb{L}}$ is unitarily equivalent to a *real* matrix as we next demonstrate explicitly. Define $\mathbf{J} = \frac{1}{\sqrt{2}} \left[\begin{smallmatrix} 1 & -\boldsymbol{\imath} \\ 1 & \boldsymbol{\imath} \end{smallmatrix}\right]$ and consider a unitary equivalence defined by

$$\tilde{\mathbb{L}}^R = (\mathbf{I}_{\nu_q} \otimes \mathbf{J}^\star)\, \tilde{\mathbb{L}}\, (\mathbf{I}_{\nu_p} \otimes \mathbf{J}).$$

$\tilde{\mathbb{L}}^R$ can be partitioned into 2×2 blocks, conforming precisely to the earlier partitioning of $\tilde{\mathbb{L}}$, and indeed one finds for each block, $\tilde{\mathbb{L}}_{k,j}^{(2)R} = \mathbf{J}^\star \tilde{\mathbb{L}}_{k,j}^{(2)} \mathbf{J}$ for $j = 1, ..., \nu_p$ and $k = 1, ..., \nu_q$. One may check directly that $\tilde{\mathbb{L}}_{k,j}^{(2)R}$ is real. Indeed, each block may be determined uniquely from a corresponding (real) Sylvester equation:

$$\tilde{\mathbb{L}}_{k,j}^{(2)R} \begin{bmatrix} 0 & \omega_k \\ -\omega_k & 0 \end{bmatrix} - \begin{bmatrix} 0 & \zeta_j \\ -\zeta_j & 0 \end{bmatrix} \tilde{\mathbb{L}}_{k,j}^{(2)R} = 2\phi_k\rho_j \begin{bmatrix} \mathsf{Re}\left(H(\boldsymbol{\imath}\,\zeta_j) - H(\boldsymbol{\imath}\,\omega_k)\right) & \mathsf{Im}\left(H(\boldsymbol{\imath}\,\zeta_j)\right) \\ \mathsf{Im}\left(H(\boldsymbol{\imath}\,\omega_k)\right) & 0 \end{bmatrix},$$

for $j = 1, ..., \nu_p$ and $k = 1, ..., \nu_q$. One may solve these Sylvester equations for $\tilde{\mathbb{L}}_{k,j}^{(2)R}$, in real arithmetic, independently of one another in parallel.

Similar considerations may be applied to (21), (22), and (23), which define respectively, the complementary matrix $\widetilde{\mathbb{M}}$, and vectors $\tilde{\mathbb{h}}$ and $\tilde{\mathbf{g}}$. For $\widetilde{\mathbb{M}}$, we find as before, a unitarily equivalent real matrix:

$$\widetilde{\mathbb{M}}^R = (\mathbf{I}_{\nu_q} \otimes \mathbf{J}^\star)\, \widetilde{\mathbb{M}}\, (\mathbf{I}_{\nu_p} \otimes \mathbf{J}),$$

which may be partitioned into sub-matrices, $\widetilde{\mathbb{M}}_{k,j}^{(2)R} = \mathbf{J}^\star \widetilde{\mathbb{M}}_{k,j}^{(2)} \mathbf{J}$ for $j = 1, \ldots, \nu_p$ and $k = 1, \ldots, \nu_q$, conforming to the partitioning for $\tilde{\mathbb{L}}^R$. These sub-matrices are directly computable from real Sylvester equations:

$$\begin{aligned}
&\widetilde{\mathbb{M}}_{k,j}^{(2)R} \begin{bmatrix} 0 & \omega_k \\ -\omega_k & 0 \end{bmatrix} - \begin{bmatrix} 0 & \zeta_j \\ -\zeta_j & 0 \end{bmatrix} \widetilde{\mathbb{M}}_{k,j}^{(2)R} \\
&= 2\phi_k \rho_j \begin{bmatrix} \mathsf{Im}\left(\omega_k H(\mathbb{i}\,\omega_k) - \zeta_j H(\mathbb{i}\,\zeta_j)\right) & \mathsf{Re}\left(\zeta_j H(\mathbb{i}\,\zeta_j)\right) \\ \mathsf{Re}\left(\omega_k H(\mathbb{i}\,\omega_k)\right) & 0 \end{bmatrix},
\end{aligned}$$

for $j = 1, \ldots, \nu_p$, $k = 1, \ldots, \nu_q$. For $\tilde{\mathbb{h}}$ and $\tilde{\mathbf{g}}$, we may define corresponding real vectors,

$$\tilde{\mathbb{h}}^{(R)} = (\mathbf{I}_{\nu_q} \otimes \mathbf{J}^\star)\tilde{\mathbb{h}} \quad \text{and} \quad \tilde{\mathbf{g}}^{(R)T} = \tilde{\mathbf{g}}^T (\mathbf{I}_{\nu_p} \otimes \mathbf{J}).$$

If we return to QuadBT (Algorithm 2) and replace $\tilde{\mathbb{L}}$, $\widetilde{\mathbb{M}}$, $\tilde{\mathbb{h}}$, and $\tilde{\mathbf{g}}$ with their real counterparts derived above, $\tilde{\mathbb{L}}^{(R)}$, $\widetilde{\mathbb{M}}^{(R)}$, $\tilde{\mathbb{h}}^{(R)}$, and $\tilde{\mathbf{g}}^{(R)}$, then the computations in Steps (3) and (4) may be done entirely in real arithmetic, and the final reduced model defined in Step (4) will be real, even in the presence of rounding errors.

4.2 MIMO Systems

We have thus far presented an analysis framework specific to SISO dynamical systems, however the MIMO case follows from this nearly immediately; the only change occurs in the construction of $\tilde{\mathbb{L}}$, $\widetilde{\mathbb{M}}$, $\tilde{\mathbb{h}}$ and $\tilde{\mathbf{c}}_r$, and not in Algorithm 2 itself. Specifically, assume that the underlying dynamical system has m inputs and p outputs (e.g., as in (1)). In this case, the transfer function function is a matrix-valued rational function and we will denote it by $\mathbf{H}(s)$ so that for any frequency $\hat{s} \in \mathbb{C}$, we have $\mathbf{H}(\hat{s}) \in \mathbb{C}^{p \times m}$. Therefore, assuming the same set-up as in Proposition 3.1, in defining $\tilde{\mathbb{L}}$, we have

$$\tilde{\mathbb{L}}_{k,j} = -\phi_k \rho_j \frac{\mathbf{H}(\mathbb{i}\omega_k) - \mathbf{H}(\mathbb{i}\zeta_j)}{\mathbb{i}\omega_k - \mathbb{i}\zeta_j} \in \mathbb{C}^{p \times m}, \tag{31}$$

as the (k, j)th *block* of $\tilde{\mathbb{L}}$ and therefore $\tilde{\mathbb{L}} \in \mathbb{C}^{(p \cdot N_q) \times (m \cdot N_p)}$. Similarly, in Proposition 3.3, we will have

$$\widetilde{\mathbb{M}}_{k,j} = -\phi_k \rho_j \frac{\mathbb{i}\omega_k \mathbf{H}(\mathbb{i}\omega_k) - \mathbb{i}\zeta_j \mathbf{H}(i\zeta_j)}{\mathbb{i}\omega_k - \mathbb{i}\zeta_j} \in \mathbb{C}^{p \times m}, \tag{32}$$

as the (k, j)th *block* of $\widetilde{\mathbb{M}}$ and thus $\widetilde{\mathbb{M}} \in \mathbb{C}^{(p \cdot N_q) \times (m \cdot N_p)}$. Since we are dealing with MIMO systems, we replace $\tilde{\mathbb{h}}$ with $\widetilde{\mathbb{H}} = \tilde{\mathbf{L}}^* \mathbf{B} \in \mathbb{C}^{(N_q \cdot p) \times m}$ and $\tilde{\mathbf{g}}^T$ with $\widetilde{\mathbb{G}} = \mathbf{C}\widetilde{\mathbf{U}} \in \mathbb{C}^{p \times (N_p \cdot m)}$ with the kth row block of $\widetilde{\mathbb{H}}$ and jth columns block of $\widetilde{\mathbb{G}}$ given by

$$\widetilde{\mathbb{H}}_k = \phi_k \mathbf{H}(\mathbb{i}\omega_k) \in \mathbb{C}^{p \times m} \quad \text{and} \quad \widetilde{\mathbb{G}}_j = \rho_j \mathbf{H}(\mathbb{i}\zeta_j) \in \mathbb{C}^{p \times m}.$$

Then, Algorithm 2 simply proceeds with these new definitions.

4.2.1 Numerical example: the MIMO case

For this scenario, we utilize the [iss1r] numerical test case, having $p = 3$ outputs and $m = 3$ inputs. We follow the specifications in §4.2 to adapt the newly-proposed method to the MIMO case.

Choose both 200 and 400 logarithmically-spaced points in the interval $[10^{-1}, 10^{2}]\imath$ as nodes for [ExpTrap] (the left and right nodes are chosen to be distinct). For [B/CC], choose $L = 10$ and $L = 10.5$ for $\widetilde{\mathbf{U}}$ and $\widetilde{\mathbf{L}}$, respectively, using both 200 and 400 nodes. We then vary the reduction order from $r = 2$ to $r = 24$, and compute the relative $\mathcal{H}_\infty$ and $\mathcal{H}_2$ errors, corresponding to both quadrature schemes. The results are presented in Fig. 7. As in the SISO case, QuadBT accurately mimics BT performance. In this MIMO example, [B/CC] outperforms [ExpTrap].

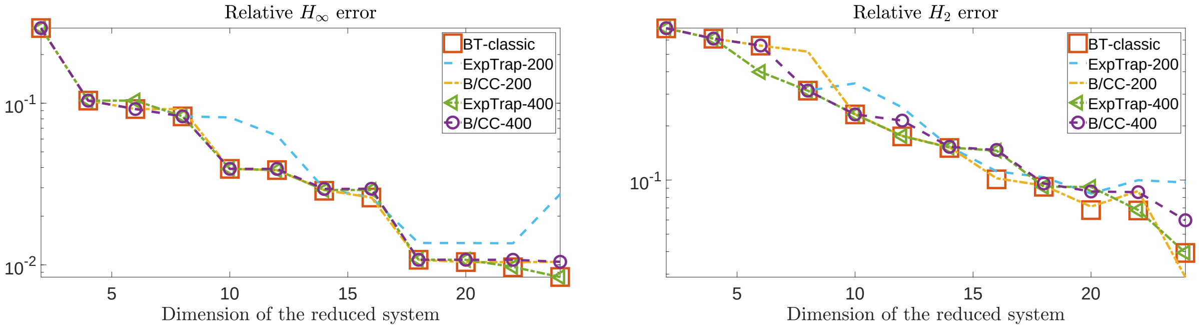

Figure 7: Relative $\mathcal{H}_\infty$ (left) and $\mathcal{H}_2$ (right) approximation errors for different quadrature schemes ([ExpTrap] vs [B/CC]) and different numbers of nodes (200 vs 400) for [iss1r] MIMO model.

4.3 Discrete-time systems

Consider discrete-time dynamical systems characterized by the difference equations

$$\begin{aligned} \mathbf{E}\mathbf{x}_{k+1} &= \mathbf{A}\,\mathbf{x}_k + \mathbf{B}\,\mathbf{u}_k, \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k, \end{aligned} \tag{33}$$

where the input mapping is given by $\mathbf{u} : \mathbb{Z} \to \mathbb{R}^m$, the state trajectory/variable is $\mathbf{x} : \mathbb{Z} \to \mathbb{R}^n$, and the output mapping is $y : \mathbb{Z} \to \mathbb{R}^p$. The system matrices, as in the continuous case, are given by $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n\times n}$, and $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$. We assume that the discrete system is *asymptotically stable*, i.e., the eigenvalues of the matrix pencil $(\mathbf{A}, \mathbf{E})$ are located in the open unit disc.

The reachability Gramian $\mathbf{P}$ in the frequency domain is given by

$$\mathbf{P} = \frac{1}{2\pi}\int_0^{2\pi} (e^{\imath\zeta}\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{B}^T(e^{-\imath\zeta}\mathbf{E}^T - \mathbf{A}^T)^{-1} d\zeta. \tag{34}$$

Similarly, the observability Gramian can be represented as

$$\mathbf{Q} = \frac{1}{2\pi}\int_0^{2\pi} (e^{-\imath\omega}\mathbf{E}^T - \mathbf{A}^T)^{-1}\mathbf{C}^T\mathbf{C}(e^{\imath\omega}\mathbf{E} - \mathbf{A})^{-1} d\omega \tag{35}$$

$\mathbf{P}$ and $\mathbf{Q}$ satisfy the following Stein equations:

$$\mathbf{A}\mathbf{P}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = \mathbf{E}\mathbf{P}\mathbf{E}^T, \quad \mathbf{A}^T\mathbf{Q}\mathbf{A} + \mathbf{C}^T\mathbf{C} = \mathbf{E}^T\mathbf{Q}\mathbf{E}. \tag{36}$$

As in the continuous-time case, we employ a numerical quadrature rule to approximate the the Gramian $\mathbf{P}$ defined in (34) to obtain the approximant

$$\mathbf{P} \approx \widetilde{\mathbf{P}} = \sum_{j=1}^{N_p} \rho_j^2 (e^{\imath\omega_j}\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{B}^T(e^{-\imath\omega_j}\mathbf{E}^T - \mathbf{A}^T)^{-1} \tag{37}$$

with ρ_j^2 and $\omega_j \in [0, 2\pi]$ denoting, respectively, quadrature weights and nodes.

We apply a similar quadrature approximation for $\mathbf{Q}$ and obtain

$$\mathbf{Q} \approx \widetilde{\mathbf{Q}} = \sum_{k=1}^{N_q} \phi_k^2 (e^{-\boldsymbol{i}\zeta_k}\mathbf{E}^T - \mathbf{A}^T)^{-1}\mathbf{C}^T\mathbf{C}(e^{\boldsymbol{i}\zeta_k}\mathbf{E} - \mathbf{A})^{-1}, \tag{38}$$

where ϕ_k^2 and $\zeta_k \in [0, 2\pi]$ denote, respectively, quadrature weights and nodes associated with approximating $\mathbf{Q}$. The rest of the QuadBT formulation for discrete systems follows immediately as in Algorithm 2. We omit those details here for brevity.

4.3.1 Numerical example: the discrete-time case

We analyze an nth-order low-pass digital Butterworth filter with normalized cutoff frequency W_c. Choose the parameters $n = 40$ and $W_c = 0.6$. The realization can be obtained in MATLAB using the command butter. In Fig. 8 we depict the frequency response and the poles of the resulting discrete-time system.

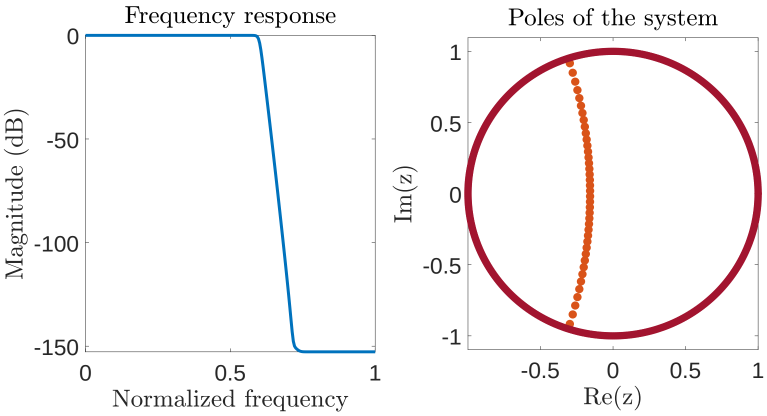

Figure 8: The frequency response and the poles of the digital Butterworth filter model.

Next, choose both 150 and 300 linearly-spaced points in the interval $[0, 2\pi)$ as the ω_j's and ζ_k's in (37) and (38), for [ExpTrap]. Here, the left and right nodes are chosen to be the same. For the [B/CC] quadrature, the values of ω_j's and ζ_k's are chosen to be Chebyshev nodes instead; again both using 150 and 300 nodes. We vary the reduction order from $r = 2$ to $r = 40$ and collect the relative $\mathcal{H}_\infty$ and $\mathcal{H}_2$ errors, corresponding to both quadrature schemes and classical BT. The results are presented in Fig. 9. QuadBT via both [ExpTrap] and [B/CC] almost exactly replicates the BT behavior, illustrating the success of our proposed method in the discrete-time setting as well.

4.4 An infinite-dimensional example

We revisit a numerical example from [6] describing the response of an infinite-dimensional linear system, in this case a one-dimensional heat equation. The evolution of the temperature distribution on a semi-infinite rod is characterized by the PDE $\frac{\partial y}{\partial t} - \frac{\partial^2 y}{\partial z^2} = 0$, where $y(z, t)$ denotes the temperature at position z and time t. The temperature is controlled at the location $z = 0$, while the temperature at $z = 1$ is the quantity of interest. By enforcing particular initial and boundary conditions as in [6], one may explicitly derive a transfer function describing the mapping from $y(0, t)$ to $y(1; t)$:

$$H(s) = \frac{Y(1, s)}{Y(0, s)} = e^{-\sqrt{s}}, \tag{39}$$

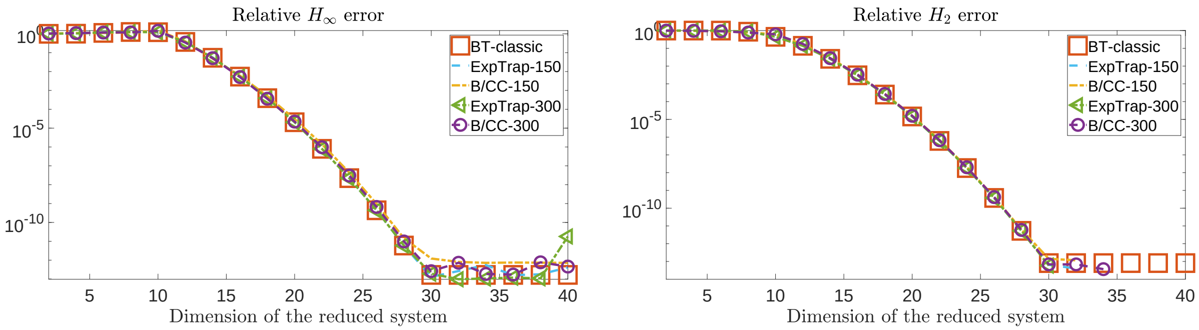

Figure 9: The relative $\mathcal{H}_\infty$ approximation error for different quadratures applied to the Butterworth digital filter model.

where $Y(\cdot, s)$ denotes the Laplace transform of $y(\cdot, t)$. Balanced truncation can be approached within an operator-theoretic framework using ideas found for example in [19, 48]. However, we are able to bypass these subtleties, since the transfer function $H(s)$ in (39) is immediately available, as well as its derivative $H'(s) = -\frac{e^{-\sqrt{s}}}{2\sqrt{s}}$.

We directly apply QuadBT with [ExpTrap] using 50 logarithmically-spaced points in the interval $[10^{-2}, 10^{1}]\imath$. The results are depicted in Fig. 10. In the left plot, we show the approximation error, $|\ H(\imath\omega) - H_r(\imath\omega)\ |$, for $r = 8, 12, 16, 20$. Moreover, in the right side of Fig. 10, we present the approximate Hankel singular values of the transfer function $H(s) = e^{-\sqrt{s}}$. The reduced models via QuadBT successfully approximates the original transfer function without requiring a discretization of the underlying PDE.

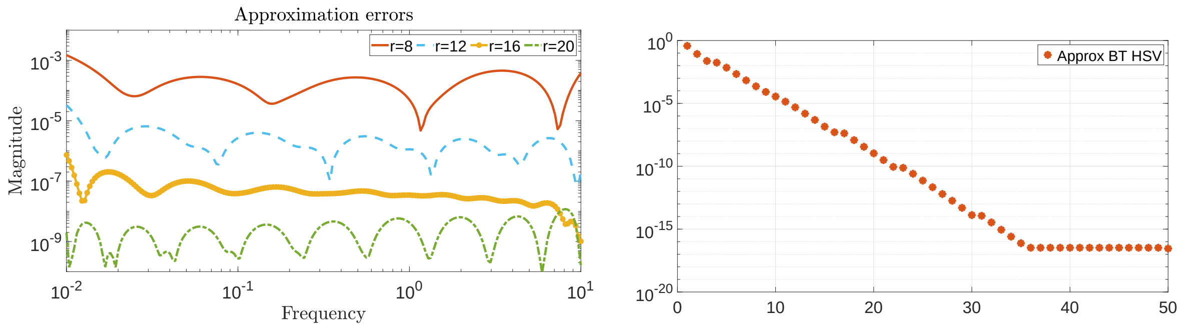

Figure 10: The approximation error for different reduction orders (left) and the approximated Hankel singular values (right) for $H(s) = e^{-\sqrt{s}}$.

5 Time-domain formulation

Thus far, we have developed ideas that are central to QuadBT presuming access to *frequency-domain data*, i.e., data resulting from transfer function sampling at selected points in the complex plane. This formulation utilized a frequency domain description of $\mathbf{P}$ and $\mathbf{Q}$, expressed in (3) and (4). In this section, we establish an analogous framework in the time domain, sampling the system response at selected instants in time and then using a time-domain representation of $\mathbf{P}$ and $\mathbf{Q}$ to derive a realization corresponding to an approximate balanced truncation. There are elements in common with time-domain Balanced POD [49, 55, 61], which uses time-domain state snapshots. What we show that, as in the frequency domain version, those implementations can be performed without access to state (internal) data. In our presentation, we will adapt the notation from [49] to our formulation. We will focus on the single-input/single-output case for simplicity. However, unlike in [49], we will include the $\mathbf{E}$ term in our formulation to keep the paper consistent with the earlier frequency-domain framework of QuadBT.

5.1 Time-domain QuadBT for continuous-time systems

As illustrated in [49, 55, 61], the Balanced POD approach can be applied in the time-domain by considering a numerical quadrature approximation to time-domain definitions of the Gramians, as given, e.g., in (2) for the reachability Gramian. We also note that in the seminal paper [38] where BT was originally introduced, Moore had already mentioned the potential of quadrature-based approximation to Gramians, thus motivating time-domain Balanced POD via empirical Gramians. We refer to the recent work [28] for a comparative study on BT-related methods employing empirical Gramians. Following on to themes developed here, we will illustrate that the time-domain Balanced POD formulation can be fully implemented using only input/output data and in particular, without access to state variables or internally characterized state realizations.

Assume that the time domain formula (2) for $\mathbf{P}$ is approximated via a numerical quadrature on $(0,\infty)$:

$$\mathbf{P} = \int_0^\infty e^{\mathbf{E}^{-1}\mathbf{A}t}\mathbf{E}^{-1}\mathbf{b}\mathbf{b}^T\mathbf{E}^{-T}e^{\mathbf{A}^T\mathbf{E}^{-T}t}dt \approx \sum_{j=1}^{N_p}\rho_j e^{\mathbf{E}^{-1}\mathbf{A}t_j}\mathbf{E}^{-1}\mathbf{b}\mathbf{b}^T\mathbf{E}^{-T}e^{\mathbf{A}^T\mathbf{E}^{-T}t_j}, \tag{40}$$

where (ρ_j, t_j) for $j = 1,\ldots,N_p$ denote quadrature weight/node pairs on $(0,\infty)$. Define

$$\widetilde{\mathbf{U}} = \begin{bmatrix} \rho_1 e^{\mathbf{E}^{-1}\mathbf{A}t_1}\mathbf{E}^{-1}\mathbf{b} & \cdots & \rho_{N_p} e^{\mathbf{E}^{-1}\mathbf{A}t_{N_p}}\mathbf{E}^{-1}\mathbf{b}\end{bmatrix} \in \mathbb{R}^{n\times N_p}. \tag{41}$$

Then, $\widetilde{\mathbf{P}} = \widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^*$ is an approximation to $\mathbf{P}$ with $\widetilde{\mathbf{U}}$ as the (approximate) square-root factor. As pointed out in [49], the columns of $\widetilde{\mathbf{U}}$ without the quadrature weights can be viewed as the snapshot data obtained from simulating the system

$$\mathbf{E}\dot{\mathbf{w}} = \mathbf{A}\mathbf{w}, \quad \mathbf{w}(0) = \mathbf{E}^{-1}\mathbf{b}. \tag{42}$$

One may form a time-domain quadrature approximation to $\mathbf{Q}$ in a similar fashion:

$$\mathbf{Q} = \int_0^\infty e^{\mathbf{E}^{-T}\mathbf{A}^T t}\mathbf{E}^{-T}\mathbf{c}\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}t}dt \approx \sum_{i=1}^{N_q}\phi_i e^{\mathbf{E}^{-T}\mathbf{A}^T\tau_i}\mathbf{E}^{-T}\mathbf{c}\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_i} \tag{43}$$

with (ϕ_i, τ_j) for $i = 1,\ldots,N_q$ being quadrature weight/node pairs on $(0,\infty)$.

Evidently, the corresponding square-root factor is

$$\tilde{\mathbf{L}}^T = \begin{bmatrix} \phi_1\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_1} \\ \phi_2\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_2} \\ \vdots \\ \phi_{N_q}\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_{N_q}} \end{bmatrix} \in \mathbb{R}^{N_q\times n}, \tag{44}$$

so that $\widetilde{\mathbf{Q}} = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^*$ approximates $\mathbf{Q}$. Similar to $\widetilde{\mathbf{U}}$, the columns of $\tilde{\mathbf{L}}$ without quadrature weights can be viewed as the snapshot data from simulating the dual system given by

$$\mathbf{E}^T\dot{\mathbf{z}} = \mathbf{A}^T\mathbf{z}, \quad \mathbf{z}(0) = \mathbf{E}^{-T}\mathbf{c}. \tag{45}$$

The square-root implementation of time-domain Balanced POD [49] proceeds by using $\widetilde{\mathbf{U}}$ and $\tilde{\mathbf{L}}$ in place of $\mathbf{U}$ and $\mathbf{L}$ in the square-root implementation of BT in Algorithm 1. So far, this still reflects a realization-*dependent* implementation, i.e., we require access to state-space representation. We will avoid this requirement below.

As we did with frequency-domain data, we inspect $\widetilde{\mathbb{L}} = \tilde{\mathbf{L}}^T\mathbf{E}\widetilde{\mathbf{U}} \in \mathbb{R}^{N_q\times N_p}$. Using (41) and (44), the (i,j)th component of $\widetilde{\mathbb{L}}$ is obtained as

$$\widetilde{\mathbb{L}}_{ij} = \phi_i\rho_j\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_i}\mathbf{E}e^{\mathbf{E}^{-1}\mathbf{A}t_j}\mathbf{E}^{-1}\mathbf{b} = \phi_i\rho_j h(\tau_i + t_j), \tag{46}$$

where $h:\mathbb{R}\to\mathbb{R}$ is the impulse response of the underlying dynamical system, as

$$h(t) = \mathbf{c}^T e^{\mathbf{E}^{-1}\mathbf{A}t}\mathbf{E}^{-1}\mathbf{b}. \tag{47}$$

The last equality in (46) follows from the observation

$$\left(\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_i}\right)\mathbf{E}e^{\mathbf{E}^{-1}\mathbf{A}t_j} = \left(e^{\mathbf{E}^{-1}\mathbf{A}\tau_i}\mathbf{E}^{-1}\right)\mathbf{E}e^{\mathbf{E}^{-1}\mathbf{A}t_j} = e^{\mathbf{E}^{-1}\mathbf{A}(\tau_i+t_j)}, \tag{48}$$

which directly follows from inspection of the partial sums of the series expansion of the parenthesized expression.

Using similar ideas, one can also show that

$$\tilde{\mathbf{L}}^T\mathbf{b} = \begin{bmatrix} \phi_1\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_1}\mathbf{b} \\ \phi_2\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_2}\mathbf{b} \\ \vdots \\ \phi_{N_q}\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_{N_q}}\mathbf{b} \end{bmatrix} = \begin{bmatrix} \phi_1 h(\tau_1) \\ \phi_2 h(\tau_2) \\ \vdots \\ \phi_{N_q} h(\tau_{N_q}) \end{bmatrix}, \tag{49}$$

and

$$\begin{aligned} \mathbf{c}^T\widetilde{\mathbf{U}} &= \begin{bmatrix} \rho_1\mathbf{c}^T e^{\mathbf{E}^{-1}\mathbf{A}t_1}\mathbf{E}^{-1}\mathbf{b} & \cdots & \mathbf{c}^T\rho_{N_p}e^{\mathbf{E}^{-1}\mathbf{A}t_{N_p}}\mathbf{E}^{-1}\mathbf{b} \end{bmatrix} \\ &= \begin{bmatrix} \rho_1 h(t_1) & \cdots & \rho_{N_p}h(t_{N_p}) \end{bmatrix}. \end{aligned} \tag{50}$$

Now, let us consider the (i,j)th component of $\widetilde{\mathbb{M}}$:

$$\widetilde{\mathbb{M}}_{ij} = \phi_i\rho_j\mathbf{c}^T\mathbf{E}^{-1}e^{\mathbf{A}\mathbf{E}^{-1}\tau_i}\mathbf{A}e^{\mathbf{E}^{-1}\mathbf{A}t_j}\mathbf{E}^{-1}\mathbf{b} = \phi_i\rho_j h'(\tau_i + t_j) \tag{51}$$

where

$$h'(t) = \frac{d}{dt}h(t) = \mathbf{c}^T e^{\mathbf{E}^{-1}\mathbf{A}t}\mathbf{E}^{-1}\mathbf{A}\mathbf{E}^{-1}\mathbf{b}. \tag{52}$$

As with the other quantities, $\widetilde{\mathbb{M}}$ can also be obtained from the impulse response samples; yet in this case one will need to measure the derivative of $h'(t)$. Notice that both $\widetilde{\mathbb{L}}$ and $\widetilde{\mathbb{M}}$ are *Hankel matrices*.

We have just shown that Balanced POD can be obtained directly from sampling the impulse response $h(t)$ and its derivative $h'(t)$ without access to state-snapshots data. A time-domain version of QuadBT in Algorithm 2 immediately follows, which we do not explicitly write here for brevity.

Remark 5.1. *Opmeer in [41] established a connection between Balanced POD and interpolatory model reduction when the state data from the simulations of (42) and (45) obtained via a numerical ODE solver such as forward and backward Euler, and other multi-stage implicit methods. In [13], Bertram and Faßbender have further generalized and solidified this connection, by giving explicit connections between the Butcher tableau of the underlying Runge-Kutta method and the resulting interpolation points.*

Remark 5.2. *We note that the impulse response measurements $h(t_i)$ and $h'(t_i)$, needed to construct the data matrices in (46), (49), (50), and (51), are assumed to be available. In practical scenarios, we would need to rely on estimating such values from experimental data or from numerical simulations. These issues will be investigated in a future work.*

Remark 5.3. *In related current work [46], impulse response sampling of $h(t)$ and $h'(t)$ are employed to extend the Eigenvalue Realization Algorithm [33] to continuous-time dynamical systems.*

5.2 Time-domain QuadBT for discrete-time systems

For the (SISO) discrete-time dynamical system

$$\mathbf{E}\mathbf{x}_{k+1} = \mathbf{A}\,\mathbf{x}_k + \mathbf{b}\,u_k, \quad y_k = \mathbf{c}^T\mathbf{x}_k, \tag{53}$$

time-domain formulation simplifies drastically due to the uniform discrete time-stepping. The reachability Gramian $\mathbf{P}$ for (53), in time-domain, is given by

$$\mathbf{P} = \sum_{k=0}^{\infty}(\mathbf{E}^{-1}\mathbf{A})^k\mathbf{E}^{-1}\mathbf{b}\mathbf{b}^T\mathbf{E}^{-T}(\mathbf{A}^T\mathbf{E}^{-T})^k. \tag{54}$$

Time-domain Balanced POD for discrete-time systems [36] uses

$$\widetilde{\mathbf{U}} = \begin{bmatrix}\mathbf{E}^{-1}\mathbf{b} & (\mathbf{E}^{-1}\mathbf{A})\mathbf{E}^{-1}\mathbf{b} & \cdots & (\mathbf{E}^{-1}\mathbf{A})^{N_p-1}\mathbf{E}^{-1}\mathbf{b}\end{bmatrix} \in \mathbb{R}^{n\times N_p}$$

as an approximate square-root factor for $\mathbf{P}$, i.e.,

$$\mathbf{P} \approx \widetilde{\mathbf{P}} = \widetilde{\mathbf{U}}\widetilde{\mathbf{U}}^T = \sum_{k=0}^{N_p-1}(\mathbf{E}^{-1}\mathbf{A})^k\mathbf{E}^{-1}\mathbf{b}\mathbf{b}^T\mathbf{E}^{-T}(\mathbf{A}^T\mathbf{E}^{-T})^k.$$

One follows the same discussion for $\mathbf{Q}$. For (53), $\mathbf{Q}$ is given by

$$\mathbf{Q} = \sum_{k=0}^{\infty}(\mathbf{E}^{-T}\mathbf{A}^T)^k\mathbf{E}^{-T}\mathbf{c}\mathbf{c}^T\mathbf{E}^{-1}(\mathbf{A}\mathbf{E}^{-1})^k. \tag{55}$$

Then, an approximate square-root factor for $\mathbf{Q}$ is

$$\tilde{\mathbf{L}}^T = \begin{bmatrix}\mathbf{c}^T\mathbf{E}^{-1}\\ \mathbf{c}^T\mathbf{E}^{-1}(\mathbf{A}\mathbf{E}^{-1})\\ \vdots\\ \mathbf{c}^T\mathbf{E}^{-1}(\mathbf{A}\mathbf{E}^{-1})^{N_q-1}\end{bmatrix} \in \mathbb{R}^{N_q\times n},$$

such that

$$\mathbf{Q} \approx \widetilde{\mathbf{Q}} = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^T = \sum_{k=0}^{N_q-1}(\mathbf{E}^{-T}\mathbf{A}^T)^k\mathbf{E}^{-T}\mathbf{c}\mathbf{c}^T\mathbf{E}^{-1}(\mathbf{A}\mathbf{E}^{-1})^k.$$

Note that these truncated sums are also the main tool in obtaining low-rank approximations to $\mathbf{P}$ and $\mathbf{Q}$ (see, e.g., [25, 44, 56]). Then, [36] observes that $\tilde{\mathbb{L}} = \tilde{\mathbf{L}}^T\mathbf{E}\widetilde{\mathbf{U}}$ can be directly obtained from data: Direct computation shows that

$$\tilde{\mathbb{L}}_{ij} = \mathbf{c}^T(\mathbf{E}^{-1}\mathbf{A})^{i+j-2}\mathbf{E}^{-1}\mathbf{b}.$$

In other words, in time-domain Balanced POD for discrete-time systems $\tilde{\mathbb{L}}$ is the Hankel matrix of Markov parameters. Similarly, one can directly shows that $\widetilde{\mathbb{M}} = \tilde{\mathbf{L}}^T\mathbf{E}\widetilde{\mathbf{U}}$ is the shfited Hankel matrix of Markov parameters, i.e.,

$$\widetilde{\mathbb{M}}_{ij} = \mathbf{c}^T(\mathbf{E}^{-1}\mathbf{A})^{i+j-1}\mathbf{E}^{-1}\mathbf{b}.$$

Similarly, $\tilde{\mathbf{L}}^T\mathbf{b}$ and $\mathbf{c}^T\widetilde{\mathbf{U}}$ can also be directly obtained from Markov parameters. Therefore, [36] observed that for the special case of discrete-time dynamical systems, time-domain Balanced POD can be obtained from input-output data using the Markov parameters. For discrete-time dynamical systems, the Markov parameters are the samples of the impulse response function $h(k) = \mathbf{c}^T(\mathbf{E}^{-1}\mathbf{A})^k\mathbf{E}^{-1}\mathbf{b}$ for

$k = 0, 1, 2, \ldots$. [36] have also pointed out that this formulation of the time-domain Balanced POD is equivalent to Eigenvalue Realization Algorithm [33].

Our formulation in §5.1 generalizes these concepts from [36] to continuous-time dynamical systems and unifies them under the umbrella of impulse-response sampling. Time-domain Balanced POD [49,61] for continuous-time systems depends on state-space quantities; our framework in §5.1 removes that requirement. Also note that continuous-time formulation gives the added flexibility that the time-stepping scheme (quadrature node) need not be uniform.

This analysis and these connections in time-domain QuadBT formulation open up many interesting venues to investigate: Can the ERA connection in the discrete-time be used for continuous-time system with finite-frequency sampling? In the case of experimental noisy data, how do the frequency-domain or time-domain QuadBT perform; is one preferable? Hankel matrices of Markov parameters are fundamental in classical realization algorithms, such as Silverman realization [53] and Ho-Kalman algorithm [29], and in optimal Hankel norm approximation [22]. How do we interpret and exploit these connections under the lenses of time-domain and frequency-domain numerical quadrature for data-driven balanced truncation? These theoretical and numerical questions are beyond the focus of this work and will be investigated in a future work. Here our goal was to set-up the basic theoretical framework for the time-domain interpretation.

6 Conclusion

In this contribution, we introduce a novel reformulation of *balanced truncation*, which is among the most important of system-theoretic model reduction tools in use today. The usual formulation of this method requires access to the original system realization, i.e., matrices that describe the state evolution as well as input and output. By way of contrast, the data-driven adaptation of balanced truncation that we propose here requires only system response data (which could be either measured or computed). The central theme that drives our development involves the approximation of Gramian-related quantities by means of numerical quadratures in the frequency domain. Depending on the convergence behaviour of the quadrature scheme and the number of nodes and weights that are used, the newly-proposed method produces results that can be made as close as desired to those of classical BT. We also discussed parallels with the interpolatory Loewner framework, since key intermediate quantities that we make use of are indeed Loewner-structured matrices. We demonstrated the application of our approach to a multitude of test cases, for both SISO and MIMO systems, and for both continuous-time and discrete-time systems. Finally, we propose time-domain extensions of our method, connecting our approach to existing methods. Future research themes will include fleshing out practical details of this time domain extension, together with developments that include select classes of nonlinear systems.

Acknowledgements

The work of Beattie and Gugercin was supported in parts by National Science Foundation under Grant No. DMS-1923221 and DMS-1819110.

References

[1] Antoulas, A.C.: Approximation of large-scale dynamical systems. SIAM, Philadelphia (2005)

[2] Antoulas, A.C., Beattie, C., Gugercin, S.: Interpolatory methods for model reduction. Computational Science and Engineering 21. SIAM, Philadelphia (2020)

[3] Antoulas, A.C., Gosea, I.V., Ionita, A.C.: Model reduction of bilinear systems in the Loewner framework. SIAM J. Scientific Computing **38**(5), B889–B916 (2016)

[4] Antoulas, A.C., Lefteriu, S., Ionita, A.C.: A tutorial introduction to the Loewner framework for model reduction (Chap. 8). In: Model Reduction and Approximation. SIAM (2017)

[5] Baur, U., Benner, P., Feng, L.: Model order reduction for linear and nonlinear systems: A system-theoretic perspective. Archives of Computational Methods in Engineering **21**(4), 331–358 (2014)

[6] Beattie, C., Gugercin, S.: Realization-independent $\mathcal{H}_2$-approximation. In: 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), pp. 4953–4958. IEEE (2012)

[7] Benner, P., Damm, T.: Lyapunov equations, energy functionals and model order reduction of bilinear and stochastic systems. SIAM J. on Control and Optimization **49**, 686–711 (2011)

[8] Benner, P., Goyal, P.: Balanced truncation model order reduction for quadratic-bilinear control systems. arXiv preprint available at https://arxiv.org/abs/1705.00160 (2017)

[9] Benner, P., Kürschner, P., Saak, J.: Self-generating and efficient shift parameters in ADI methods for large Lyapunov and Sylvester equations. Electronic Transactions on Numerical Analysis **43**, 142–162 (2014)

[10] Benner, P., Ohlberger, M., Cohen, A., Willcox, K.: Model Reduction and Approximation. SIAM, Philadelphia, PA (2017)

[11] Benner, P., Saak, J.: Numerical solution of large and sparse continuous time algebraic matrix Riccati and Lyapunov equations: A state of the art survey. GAMM Mitteilungen **36**(1), 32–52 (2013)

[12] Benner, P., Schneider, A.: Balanced truncation model order reduction for LTI systems with many inputs or outputs. In: Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems–MTNS, vol. 5 (2010)

[13] Bertram, C., Faßbender, H.: A link between Gramian-based model order reduction and moment matching. arXiv preprint available at https://arxiv.org/abs/2003.03101 (2020)

[14] Bertram, C., Faßbender, H.: Lyapunov and Sylvester equations: A quadrature framework. arXiv preprint available at https://arxiv.org/abs/1903.05383 (2020)

[15] Bhatia, R.: Matrix analysis, vol. 169. Springer (2013)

[16] Boyd, J.P.: Exponentially convergent Fourier-Chebshev quadrature schemes on bounded and infinite intervals. J. Sci. Comput **2**, 99–109 (1987)

[17] Breiten, T.: Structure-preserving model reduction for integro-differential equations. SIAM J. Control and Optimization **54**(6), 2992–3015 (2016)

[18] Chahlaoui, Y., Van Dooren, P.: Benchmark examples for model reduction of linear time-invariant dynamical systems. In: In: Benner P., Sorensen D.C., Mehrmann V. (eds), Dimension Reduction of Large-Scale Systems, vol. 45, pp. 379–392. Springer (2005)

[19] Curtain, R.F., Glover, K.: Balanced realisations for infinite dimensional systems. Operator Theory: Advances and Applications **19**, 87–104 (1986)

[20] Drmač, Z., Gugercin, S., Beattie, C.: Quadrature-based vector fitting for discretized $\mathcal{H}_2$ approximation. SIAM J. Scientific Computing **37**(2), A625–A652 (2015)

[21] Druskin, V., Knizhnerman, L., Simoncini, V.: Analysis of the rational Krylov subspace and ADI methods for solving the Lyapunov equation. SIAM J. on Numerical Analysis **49**(5), 1875–1898 (2011)

[22] Glover, K.: All optimal Hankel-norm approximationsof linear multivariable systerns and their $\mathcal{L}_\infty$-error bounds. Int. J. Control **39**, 1115–1193 (1984)

[23] Gosea, I.V., Antoulas, A.C.: Data-driven model order reduction of quadratic-bilinear systems. Numerical Linear Algebra with Applications **25**(6), e2200 (2018)

[24] Gosea, I.V., Petreczky, M., Antoulas, A.C., Fiter, C.: Balanced truncation for linear switched systems. Advances in Computational Mathematics **44**(6), 1845–1886 (2018)

[25] Gugercin, S., Sorensen, D.C., Antoulas, A.C.: A modified low-rank Smith method for large-scale Lyapunov equations. Numerical Algorithms **32**(1), 27–55 (2003)

[26] Higham, N.J.: Accuracy and stability of numerical algorithms. SIAM (2002)

[27] Himpe, C.: emgr—The Empirical Gramian Framework. Algorithms **11**(7), 91 (2018)

[28] Himpe, C.: Comparing (empirical-Gramian-based) model order reduction algorithms. In: P. Benner, et al. (eds.) Model Reduction of Complex Dynamical Systems. Springer (2021)

[29] Ho, B.L., Kalman, R.E.: Effective construction of linear state-variable models from input/output data. In: Proc. Third Annual Allerton Conf. on Circuit and System Theory, pp. 449–459. Univ. Illinois, Urbana, Ill. (1965)

[30] Ionita, A.C., Antoulas, A.C.: Data-driven parametrized model reduction in the Loewner framework. SIAM J. Scientific Computing **36**(3), A984–A1007 (2014)

[31] Juang, J.: Applied system identification. Prentice Hall, Inc. (1994)

[32] Keysight Technologies, Santa Rosa, California, United States: S-Parameter Measurements: Basics for High Speed Digital Engineers. URL http://literature.cdn.keysight.com/litweb/pdf/5991-3736EN.pdf

[33] Kung, S.: A new identification and model reduction algorithm via singular value decomposition. In: Proceedings of 12th Asilomar Conference on Circuits, Systems & Computers, Pacific Grove, CA, 1978, pp. 705–714 (1978)

[34] Kürschner, P.: Efficient low-rank solution of large-scale matrix equations. Ph.D. thesis, Otto von Guericke Universität Magdeburg (2016)

[35] Li, J.R., White, J.: Low rank solution of Lyapunov equations. SIAM J. Matrix Analysis and Applications **24**(1), 260–280 (2002)

[36] Ma, Z., Ahuja, S., Rowley, C.W.: Reduced-order models for control of fluids using the eigensystem realization algorithm. Theoretical and Computational Fluid Dynamics **25**(1-4), 233–247 (2011)

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

[38] Moore, B.: Principal component analysis in linear systems: Controllability, observability, and model reduction. IEEE Transactions on Automatic Control **26**(1), 17–32 (1981)

[39] Mullis, C., Roberts, R.: Synthesis of minimum roundoff noise fixed point digital filters. Circuits and Systems, IEEE Transactions on **23**(9), 551–562 (1976)

[40] Niconet: Benchmark examples for model reduction of linear time invariant dynamical systems. URL http://slicot.org/20-site/126-benchmark-examples-for-model-reduction

[41] Opmeer, M.R.: Model order reduction by balanced proper orthogonal decomposition and by rational interpolation. IEEE Transactions on Automatic Control **57**(2), 472–477 (2011)

[42] Peherstorfer, B., Gugercin, S., Willcox, K.: Data-driven reduced model construction with time-domain Loewner models. SIAM J. Scientific Computing **39**(5), A2152 – A2178 (2017)

[43] Penzl, T.: Numerical solution of generalized Lyapunov equations. Advances in Computational Mathematics **8**(1-2), 33–48 (1998)

[44] Penzl, T.: A cyclic low-rank smith method for large sparse Lyapunov equations. SIAM J. Scientific Computing **21**(4), 1401–1418 (1999)

[45] Phillips, J.R., Silveira, L.M.: Poor man's TBR: a simple model reduction scheme. IEEE Trans. Comput. Aided Design Integr. Circuits Syst. **24**(1), 43–55 (2004)

[46] Pontes Duff, I.: (2021). Private communication.

[47] Quarteroni, A., Manzoni, A., Negri, F.: Reduced basis methods for partial differential equations: an introduction, vol. 92. Springer (2015)

[48] Reis, T., Selig, T.: Balancing transformations for infinite-dimensional systems with nuclear Hankel operator. Integral Equations and Operator Theory **79**(1), 67–105 (2014)

[49] Rowley, C.W.: Model reduction for fluids, using balanced proper orthogonal decomposition. Int. J. Bifurcat. Chaos **15**(3), 997—-1013 (2005)

[50] Sabino, J.: Solution of large-scale Lyapunov equations via the block modified Smith method. Ph.D. thesis, Rice University (2007)

[51] Scherpen, J.M.A.: Balancing for nonlinear systems. Systems & Control Letters **21**(2), 143–153 (1993)

[52] Schulze, P., Unger, B., Beattie, C.A., Gugercin, S.: Data-driven structured realization. Linear Algebra and its Applications **537**, 250–286 (2018)

[53] Silverman, L.: Realization of linear dynamical systems. IEEE Transactions on Automatic Control **16**(6), 554–567 (1971)

[54] Simoncini, V.: Computational methods for linear matrix equations. SIAM Review **58**(3), 377–441 (2016)

[55] Singler, J.R., Batten, B.A.: A proper orthogonal decomposition approach to approximate balanced truncation of infinite dimensional linear systems. Intern. J. Computer Mathematics **86**(2), 355–371 (2009)

[56] Smith, R.: Matrix equation $\mathbf{XA} + \mathbf{BX} = \mathbf{C}$. SIAM J. Appl. Math. **16**(1), 198–201 (1968)

[57] Stykel, T.: Low-rank iterative methods for projected generalized Lyapunov equations. Electron. Trans. Numer. Anal. (ETNA) **30**, 187–202 (2008)

[58] Sun, J.G.: Perturbation bounds for the Cholesky and QR factorizations. BIT Numerical Mathematics **31**(2), 341–352 (1991)

[59] Trefethen, L.N., Weideman, J.: The exponentially convergent trapezoidal rule. SIAM Review **56**(3), 385–458 (2014)

[60] Weideman, J., Trefethen, L.: Parabolic and hyperbolic contours for computing the Bromwich integral. Mathematics of Computation **76**(259), 1341–1356 (2007)

[61] Willcox, K., Peraire, J.: Balanced model reduction via the proper orthogonal decomposition. AIAA Journal **40**(11), 2323–2330 (2002)

[62] Yang, S., Allen, M.S.: Harmonic transfer function to measure translational and rotational velocities with continuous-scan laser Doppler vibrometry. J. Vibration and Acoustics **136**(2), 021025 (2014)

Appendix A: Two numerical quadrature rules

The approach to balanced truncation that we develop here makes central use of numerical quadrature rules. We illustrate our approach using two particular quadrature rules, the exponential trapezoid rule and a refinement of the Clenshaw–Curtis quadrature rule developed by Boyd. These approaches are effective in approximating matrix-valued integrals which generically appear as: $\int_{-\infty}^{\infty}\mathbf{F}(\zeta)d\zeta$ where we assume that, for $\zeta\in\mathbb{R}$, the integrand $\mathbf{F}(\zeta)$ is continuously differentiable, $\mathbf{F}(-\zeta)=\mathbf{F}(\zeta)^T$, and $\lim_{\zeta\to\pm\infty}\zeta^2\mathbf{F}(\zeta)=\mathbf{M}=\mathbf{M}^T\in\mathbb{R}^{n\times n}$ (so that, in particular, $\|\mathbf{F}(\zeta)\|=\mathcal{O}\left(\frac{1}{\zeta^2}\right)$ as $\zeta\to\pm\infty$). This assumed form covers the Gramian integrals, (3) and (4), which are the principal cases of interest for us. The two quadrature rules that we describe below are typical of the possible choices though certainly there will be many other refined approaches that one may consider (see especially [59]).

[ExpTrap]: The exponential trapezoid rule. Note first that

$$\begin{aligned}\int_{-\infty}^{\infty}\mathbf{F}(\zeta)d\zeta &= \int_{0}^{\infty}\mathbf{F}(\zeta)+\mathbf{F}(\zeta)^T\ d\zeta\\ &= L\int_{-\infty}^{\infty}e^{L\tau}\left(\mathbf{F}(e^{L\tau})+\mathbf{F}(e^{L\tau})^T\right)\ d\tau=\int_{-\infty}^{\infty}\mathbf{G}(\tau)\ d\tau\end{aligned}$$

where $\mathbf{G}(\tau)=L\,e^{L\tau}\left(\mathbf{F}(e^{L\tau})+\mathbf{F}(e^{L\tau})^T\right)$; the first equality has used the symmetry of $\mathbf{F}(\cdot)$; and the second equality has introduced a change-of-variable for $\zeta>0$ as $\zeta=e^{L\tau}$. For any $h>0$ and $t\in\mathbb{R}$, $\sum_{k=-\infty}^{\infty}\mathbf{G}(t+khL)$, viewed as a function of t, is well-defined, continuously differentiable, and hL-periodic. In particular, it has a pointwise convergent Fourier series,

$$\sum_{k=-\infty}^{\infty}\mathbf{G}(t+khL)=\sum_{\ell=0}^{\infty}\mathbf{C}_\ell\ e^{\imath\frac{2\pi\ell}{hL}t}$$

with (matrix-valued) coefficients that may be evaluated as: $\mathbf{C}_\ell=\frac{2\pi}{hL}\widehat{\mathbf{G}}\left(\frac{2\pi\ell}{hL}\right)$, where $\widehat{\mathbf{G}}(\omega)$ denotes the Fourier transform of $\mathbf{G}(t)$. Evaluating at $t=0$, this expression may be rearranged to

$$hL\sum_{k=-\infty}^{\infty}\mathbf{G}(khL)-\int_{-\infty}^{\infty}\mathbf{G}(\tau)\ d\tau=2\pi\sum_{\ell=1}^{\infty}\widehat{\mathbf{G}}\left(\frac{2\pi\ell}{hL}\right)$$

or equivalently,

$$hL\sum_{k=-\infty}^{\infty}e^{khL}\left(\mathbf{F}(e^{khL})+\mathbf{F}(e^{khL})^T\right)-\int_{-\infty}^{\infty}\mathbf{F}(\zeta)\ d\tau=2\pi\sum_{\ell=1}^{\infty}\widehat{\mathbf{G}}\left(\frac{2\pi\ell}{hL}\right).$$

Significantly, the right-hand side becomes exponentially close to zero as $h\to0$. The first sum on the left-hand side manifests the trapezoid rule, which becomes computationally tractable with suitable truncation of the sum: say, $\nu_0\le k\le\nu_0+\varpi$ This leads to a quadrature rule:

$$\int_{-\infty}^{\infty}\mathbf{F}(\zeta)d\zeta\approx hLe^{\nu_0hL}\mathbf{F}(0)+hLe^{\nu_0hL}\sum_{\ell=0}^{\varpi}e^{\ell hL}\left(\mathbf{F}(e^{(\nu_0+\ell)hL})+\mathbf{F}(-e^{(\nu_0+\ell)hL})\right)$$

This includes a correction that weights the value of $\mathbf{F}$ at $\zeta=0$ as well.

[B/CC]: The Boyd/Clenshaw–Curtis rule. Boyd adapted the Clenshaw-Curtis quadrature rule to infinite integration domains in [16]. We summarize the application of this rule to our setting. Starting with the change-of-variable $\zeta=L\cot(\tau)$, we find:

$$\int_{-\infty}^{\infty}\mathbf{F}(\zeta)d\zeta=\int_{0}^{\pi}\frac{\mathbf{F}(L\cot(\tau))L}{\sin^2(\tau)}d\tau \tag{56}$$

The right-hand integrand is evidently π-periodic. With the asymptotics assumed for $\mathbf{F}$ we have that limits exist for the right-hand integrand both at 0 and π and they are real, symmetric, and equal:

$$\lim_{\tau\to 0}\frac{\mathbf{F}(L\cot(\tau))L}{\sin^2(\tau)} = \frac{1}{L}\lim_{\tau\to 0}\left(L\cot(\tau)\right)^2\mathbf{F}(L\cot(\tau)) = \frac{1}{L}\lim_{\zeta\to\infty}\zeta^2\mathbf{F}(\zeta) = \frac{1}{L}\mathbf{M}$$

$$\lim_{\tau\to \pi}\frac{\mathbf{F}(L\cot(\tau))L}{\sin^2(\tau)} = \frac{1}{L}\lim_{\tau\to \pi}\left(L\cot(\tau)\right)^2\mathbf{F}(L\cot(\tau)) = \frac{1}{L}\lim_{\zeta\to-\infty}\zeta^2\mathbf{F}(\zeta) = \frac{1}{L}\mathbf{M}.$$

The trapezoid rule for the right-hand integral in (56) becomes a compelling choice. Defining, $h = \frac{\pi}{\varpi+1}$ for some $\varpi\in\mathbb{N}$, and then we take equally spaced points $\tau_\ell = \ell h$ for $\ell = 0,1,\ldots,\varpi+1$, giving

$$\int_{-\infty}^{\infty}\mathbf{F}(\zeta)d\zeta \approx \sum_{\ell=1}^{\varpi} hL\frac{\mathbf{F}(L\cot(\tau_\ell))}{\sin^2(\tau_\ell)} + \frac{h}{L}\mathbf{M} = \sum_{\ell=1}^{\varpi}\rho_\ell^2\mathbf{F}(\omega_\ell) + \rho_\infty^2\mathbf{M} \tag{57}$$

where $\rho_\ell^2 = \frac{L\pi}{(\varpi+1)\sin^2(\tau_\ell)}$ and $\omega_\ell = L\cot(\tau_\ell))$ for $\ell = 1,\ldots,\varpi$ and $\rho_\infty^2 = \frac{\pi}{L(\varpi+1)}$. The last term in (57) is $\frac{h}{2}$ multiplying the combined evaluation of the periodicized integrand at the endpoints, 0 and π, corresponding to the "nodes at infinity" for the original integral.

We note that a similar derivation was described in [20, §3.3] for the computation of the $\mathcal{H}_2$ norm for LTI systems.

Appendix B: Proof of Proposition 3.2

The hypotheses imply that

$$\|\mathbf{Q}-\widetilde{\mathbf{Q}}\|_2 < \sigma_{\mathsf{min}}(\mathbf{Q}) \quad\text{and}\quad \|\mathbf{P}-\widetilde{\mathbf{P}}\|_2 < \sigma_{\mathsf{min}}(\mathbf{P}).$$

We then use results from Sun [58, Thm. 1.4] (see also [26, Thm. 10.8]) to build isometries $\mathbf{\Psi}_p\in\mathbb{C}^{N_p\times n}$ and $\mathbf{\Psi}_q\in\mathbb{C}^{N_q\times n}$ and perturbations $\Delta\mathbf{L}$, $\Delta\mathbf{U}\in\mathbb{C}^{n\times n}$ such that

$$\widetilde{\mathbf{U}}^* = \mathbf{\Psi}_p(\mathbf{U}+\Delta\mathbf{U})^T \quad\text{and}\quad \tilde{\mathbf{L}}^* = \mathbf{\Psi}_q(\mathbf{L}+\Delta\mathbf{L})^T.$$

Defining $\varepsilon_q = \frac{\|\mathbf{Q}-\widetilde{\mathbf{Q}}\|_F}{\|\mathbf{Q}\|_2}$ and $\varepsilon_p = \frac{\|\mathbf{P}-\widetilde{\mathbf{P}}\|_F}{\|\mathbf{P}\|_2}$, the hypotheses assure that both

$$\mathsf{cond}(\mathbf{Q})\varepsilon_q \le \frac{\delta}{1+\delta} < 1 \quad\text{and}\quad \mathsf{cond}(\mathbf{P})\varepsilon_p \le \frac{\delta}{1+\delta} < 1,$$

and then, referring again to [58, Thm. 1.4], we have further

$$\frac{\|\Delta\mathbf{U}\|_F}{\|\mathbf{U}\|_2} \le \left(\frac{1}{\sqrt{2}}\right)\frac{\mathsf{cond}(\mathbf{P})\varepsilon_p}{1-\mathsf{cond}(\mathbf{P})\varepsilon_p} \le \frac{\delta}{\sqrt{2}} \quad\text{and}\quad \frac{\|\Delta\mathbf{L}\|_F}{\|\mathbf{L}\|_2} \le \left(\frac{1}{\sqrt{2}}\right)\frac{\mathsf{cond}(\mathbf{Q})\varepsilon_q}{1-\mathsf{cond}(\mathbf{Q})\varepsilon_q} \le \frac{\delta}{\sqrt{2}}$$

Now observe

$$\begin{aligned}\tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}} &= \mathbf{\Psi}_q(\mathbf{L}+\Delta\mathbf{L})^T\mathbf{E}(\mathbf{U}+\Delta\mathbf{U})\mathbf{\Psi}_p^T\\ &= \mathbf{\Psi}_q(\mathbf{L}^T\mathbf{E}\mathbf{U})\mathbf{\Psi}_p^T + \mathbf{\Psi}_q\,(\Delta\mathbf{L})^T(\mathbf{E}\mathbf{U})\mathbf{\Psi}_p^T + \mathbf{\Psi}_q(\mathbf{L}^T\mathbf{E})(\Delta\mathbf{U})\,\mathbf{\Psi}_p^T + \mathbf{\Psi}_q\,(\Delta\mathbf{L})^T\,\mathbf{E}\,(\Delta\mathbf{U})\,\mathbf{\Psi}_p^T\end{aligned}$$

and as a result,

$$\begin{aligned}\|\tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}} - \mathbf{\Psi}_q(\mathbf{L}^T\mathbf{E}\mathbf{U})\mathbf{\Psi}_p^T\|_F &\le \|\mathbf{\Psi}_q\,\Delta\mathbf{L}^T(\mathbf{E}\mathbf{U})\mathbf{\Psi}_p^T\|_F \quad +\\ &\qquad \|\mathbf{\Psi}_q(\mathbf{L}^T\mathbf{E})(\Delta\mathbf{U})\,\mathbf{\Psi}_p^T\|_F + \|\mathbf{\Psi}_q\,(\Delta\mathbf{L})^T\,\mathbf{E}\,(\Delta\mathbf{U})\,\mathbf{\Psi}_p^T\|_F\\ &\le \|\Delta\mathbf{L}\|_F\,\|\mathbf{E}\mathbf{U}\|_2 + \|\mathbf{L}^T\mathbf{E}\|_2\,\|\Delta\mathbf{U}\|_F + \|\Delta\mathbf{L}\|_F\,\|\mathbf{E}\|_2\|\Delta\mathbf{U}\|_F\\ &\le \frac{\delta}{\sqrt{2}}\,\|\mathbf{L}\|_2\,\|\mathbf{E}\mathbf{U}\|_2 + \frac{\delta}{\sqrt{2}}\,\|\mathbf{L}^T\mathbf{E}\|_2\,\|\mathbf{U}\|_2 + \left(\frac{\delta}{\sqrt{2}}\right)^2\|\mathbf{L}\|_2\,\|\mathbf{E}\|_2\|\mathbf{U}\|_2\\ &\le \left(\sqrt{2}\,\delta + \frac{\delta^2}{2}\right)\|\mathbf{L}\|_2\,\|\mathbf{E}\|_2\,\|\mathbf{U}\|_2 < 2\,\delta\,\|\mathbf{L}\|_2\,\|\mathbf{E}\|_2\,\|\mathbf{U}\|_2\end{aligned}$$

where the last inequality takes into account that $0 < \delta < 1$.

For the final assertion, note that the singular values of $\boldsymbol{\Psi}_q(\mathbf{L}^T\mathbf{E}\mathbf{U})\boldsymbol{\Psi}_p^*$ are $\{\sigma_1, \sigma_2, \cdots, \sigma_n\}$ augmented with 0. A consequence of Lidskii's Majorization Theorem (e.g., [15, Thm.III.4.4]) gives

$$\sum_{k=1}^{n}(\tilde{\sigma}_k - \sigma_k)^2 \leq \|\tilde{\mathbf{L}}^*\mathbf{E}\widetilde{\mathbf{U}} - \boldsymbol{\Psi}_q(\mathbf{L}^T\mathbf{E}\mathbf{U})\boldsymbol{\Psi}_p^*\|_F^2.$$

Then (20), the first assertion of Proposition 3.2, gives the conclusion. ■