Springer International Series of Numerical Mathematics?? manuscript No.
(will be inserted by the editor)

Balanced truncation for parametric linear systems using interpolation of gramians: a comparison of linear algebraic and geometric approaches

SGMSA

the date of receipt and acceptance should be inserted later

Abstract In balanced truncation model order reduction, one has to solve a pair of Lyapunov equations for the two gramians and uses them for constructing a reduced-order model. Although advances in solving such equations have been made, it is still the most expensive step in this reduction method. For systems that depend on parameters, parametric model order reduction has to deal with the dependence on parameters simultaneously with approximation of the input-output behavior of the full-order system. The use of interpolation in parametric model order reduction has become popular. Nevertheless, interpolation of gramians is rarely mentioned, most probably due to the restriction to symmetric positive semi-definite matrices. In this talk, we will present two approaches for interpolating these structured matrices which are based on linear algebra and a recently developed Riemannian geometry. The result is then utilized in constructing parametric reduced-order systems. Their numerical performances are compared on different models

Keywords Parametric model order reduction · Balanced truncation · Interpolation · Gramians · Riemannian matrix manifold · Symmetric positive semi-definite matrices of fixed rank

Mathematics Subject Classification (2000) 65D05 · 65F30 · 93C05 · notdoneyet

1 Introduction

The need for increasingly accurate simulations in science and technology results in large-scale mathematical models. Simulation of those systems is usually time-consuming or even infeasible, especially with limited computer resources. Model order reduction (MOR) is well known as a tool to deal with such problems. Founded and continuously developed already for a couple of decades, this field is still getting attraction due to the fact that many complicated or large problems have not been considered and many advanced methods have not been invoked.

Address(es) of author(s) should be given

In many cases, the full order model (FOM) depends on parameters. The reduced-order model (ROM), preferably parameter-dependent as well, is therefore required to approximate the FOM on a given parameter domain. This problem, so-called parametric MOR (PMOR), has been addressed by various approaches that are based on Krylov subspaces [cite], interpolation, optimization [cite], just to name a few. The reader is referred to the survey [7] for more details. Of our interest in this report is the methods that use interpolation for the linear parametric system of the form

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

where $E(\mu)$, $A(\mu) \in \mathbb{R}^{n\times n}$, $B(\mu) \in \mathbb{R}^{n\times m}$, $C(\mu) \in \mathbb{R}^{p\times n}$ with $p, m \ll n$, and $\mu \in \Omega \subset \mathbb{R}^d$. We assume that the matrix $E(\mu)$ is nonsingular and all eigenvalues of the pencil $\lambda E(\mu) - A(\mu)$ have negative real part for all $\mu \in \Omega$. The goal is to approximate system (1) with a parametric model

$$\begin{aligned} \tilde{E}(\mu)\dot{\tilde{x}}(t,\mu) &= \tilde{A}(\mu)\tilde{x}(t,\mu) + \tilde{B}(\mu)u(t), \\ \tilde{y}(t,\mu) &= \tilde{C}(\mu)\tilde{x}(t,\mu), \end{aligned} \tag{2}$$

where $\tilde{E}(\mu)$, $\tilde{A}(\mu) \in \mathbb{R}^{r\times r}$, $\tilde{B}(\mu) \in \mathbb{R}^{r\times m}$, $\tilde{C}(\mu) \in \mathbb{R}^{p\times r}$ and $r \ll n$.

The idea of interpolation is straightforward. On a given grid μ_i, $i = 1, \ldots, q$ in the parameter domain Ω, one computes a ROM associated with each μ_i. These ROMs can be computed using any MOR method for non-parametric models [4] and characterized by either their projection subspaces, coefficient matrices, or transfer functions. Then, they are interpolated using standard methods. These topics have been discussed intensively in many publications with applications to various fields, see, e.g., [2,6,15,8,3,18,19]. Each of them has its own strength and acts well in some specific applications but fails to be superior to the others.

When balanced truncation [13] is used, one has to solve a pair of Lyapunov equations for the two gramians. Although advances in solving such equations have been made, it is still the most expensive step in this reduction method. Therefore, any interpolation method that can circumvent this step is of interest. We suggest to interpolate the solution of these equations, so-called gramians. It is noteworthy that in large-scale setting, one should never work with full-rank solution matrices. Fortunately, in many practical cases, Lyapunov equaions accept low-rank symmetric positive semi-definite (spsd) approximation [17,5]. It not only makes the computation more efficient but also enables the squaring procedure in balanced truncation [21]. A resulting difficulty arises in the interpolation approach is that the interpolant is also expected to be spsd. That is, the spsd property must be preserved during the interpolation. To this end, we propose in this paper two approaches. In the first one, we first invoke some positive interpolation scheme to preserve the semi-definiteness and then design an offline-online decomposition so that the interpolation is deeply "embedded" in both stages through which the online stage is accelerated. We refer to this as linear algebraic approach. The second one is based completely on differential geometry, so the name. It was shown in [22, 12] that the set of spsd matrices of fixed rank can be turned into a Riemannian manifold by equipping it with a differential structure. As a result, if all solutions of the Lyapunov equations at grid points are approximated by spsd matrices of a prescribed low rank, we then encounter interpolation on a Riemannian manifold. Based on a recent result [12], we will explain in detail how to do it.

The rest of the paper is organized as follows. In section 2 we first briefly recall the squaring procedure for balancing equation and then present in detail how interpolate the gramians while preserve its low-rank semi-positiveness and how to prepare data in the offline step so that we can speed up the online step. A quotient geometry for the Riemannian manifold of spsd matrices of fixed rank is constructed in the first part of section 3. We then explain in detail how to interpolate on this manifold using the tools just developed. We also discuss the possibility of using a embedded geometry for this talk in this section. The proposed approaches are illustrated by numerical examples in section 4 in which we also compare their behaviors and numerical efficiency. Conclusion is given in section 5.

Throughout this paper, we will use ... for the nations.

2 Brief balanced truncation for parametric linear systems and standard interpolation

2.1 Balanced truncation

As other projection-based method, a balancing projection for system (1) must be constructed. To this end, one has to solve a pair of the generalized Lyapunov equations

$$E(\mu)P(\mu)A^T(\mu) + A(\mu)P(\mu)E^T(\mu) = -B(\mu)B^T(\mu), \tag{3}$$
$$E^T(\mu)Q(\mu)A(\mu) + A^T(\mu)Q(\mu)E(\mu) = -C^T(\mu)C(\mu), \tag{4}$$

for the *controllability Gramian* $P(\mu)$ and the *observability Gramian* $Q(\mu)$. In practice, these Gramians are computed in the factored form

$$P(\mu) = X(\mu)X^T(\mu), \qquad Q(\mu) = Y(\mu)Y^T(\mu)$$

with $X(\mu) \in \mathbb{R}^{n\times k_c}$ and $Y(\mu) \in \mathbb{R}^{n\times k_o}$. One can show that eigenvalues of the matrix $P(\mu)E^T(\mu)Q(\mu)E(\mu)$ are real and non-negative. The square roots of the eigenvalues of this matrix $\sigma_1(\mu) \geq \cdots \geq \sigma_n(\mu) \geq 0$ are called the *Hankel singular values* of system (1). Consider the singular value decomposition

$$Y^T(\mu)E(\mu)X(\mu) = [U_1(\mu)\ \ U_0(\mu)] \begin{bmatrix} \Sigma_1(\mu) & 0 \\ 0 & \Sigma_0(\mu) \end{bmatrix} [V_1(\mu)\ \ V_0(\mu)]^T, \tag{5}$$

where $[U_1(\mu)\ \ U_0(\mu)]$ and $[V_1(\mu)\ \ V_0(\mu)]$ are orthogonal, and

$$\Sigma_1(\mu) = \mathrm{diag}(\sigma_1(\mu), \ldots, \sigma_r(\mu)), \quad \Sigma_0(\mu) = \mathrm{diag}(\sigma_{r+1}(\mu), \ldots, \sigma_{k_{co}}(\mu))$$

with $k_{co} = \min(k_c, k_o)$. Then the reduced-order model is computed by projection

$$\begin{array}{ll} \tilde{E}(\mu) = W^T(\mu)E(\mu)T(\mu), & \tilde{A}(\mu) = W^T(\mu)A(\mu)T(\mu), \\ \tilde{B}(\mu) = W^T(\mu)B(\mu), & \tilde{C} = C(\mu)T(\mu), \end{array} \tag{6}$$

where the projection matrices are given by

$$W(\mu) = Y(\mu)U_1(\mu)\Sigma_1^{-1/2}(\mu), \qquad T(\mu) = X(\mu)V_1(\mu)\Sigma_1^{-1/2}(\mu). \tag{7}$$

The error of the approximation is shown to satisfy

$$\|H(\cdot,\mu)-\tilde{H}(\cdot,\mu)\|_{\mathcal{H}_\infty} \leq 2\big(\sigma_{r+1}(\mu)+\cdots+\sigma_{k_{co}}(\mu)\big),$$

where

$$H(s,\mu) = C(\mu)(sE(\mu)-A(\mu))^{-1}B(\mu),$$
$$\tilde{H}(s,\mu) = \tilde{C}(\mu)(s\tilde{E}(\mu)-\tilde{A}(\mu))^{-1}\tilde{B}(\mu)$$

are the transfer functions of systems (1) and (2), respectively.

2.2 Interpolation of gramians for parametric model order reduction

On the chosen grid $\mu_1,\ldots,\mu_q \in \Omega$, we solve the equations (3) and (4) for $P_j = X_jX_j^T$ and $Q_j = Y_jY_j^T$ $j=1,\ldots,q$. Note that the ranks of local gramians $P_j, j=1,\ldots,q$ and $Q_j, j=1,\ldots,q$ need not be the same. Then the parameter-dependent gramians can be approximated by interpolation as

$$P(\mu)=\sum_{j=1}^{q} w_j(\mu)X_jX_j^T, \quad Q(\mu)=\sum_{j=1}^{q} w_j(\mu)Y_jY_j^T,$$

where $w_j(\mu)$ are some weights. To preserve the semi-definiteness of gramians, we propose to use non-negative weights [1]. Moreover, it allows us to retain the factorization structure. To wit

$$\begin{aligned} P(\mu) = &= \sum_{j=1}^{q} \sqrt{w_j(\mu)}X_j\sqrt{w_j(\mu)}X_j^T \\ &= \begin{bmatrix}\sqrt{w_1(\mu)}X_1 & \cdots & \sqrt{w_q(\mu)}X_q\end{bmatrix}\begin{bmatrix}\sqrt{w_1(\mu)}X_1^T \\ \cdots \\ \sqrt{w_q(\mu)}X_q^T\end{bmatrix} \\ &=: X(\mu)X^T(\mu). \end{aligned} \tag{8}$$

Likewise,

$$Q(\mu) = Y(\mu)Y^T(\mu). \tag{9}$$

Note that computation of parametric gramians is not the ultimate goal of the task. After interpolation, we still have to proceed steps (5) and (6) to get the reduced-order model; computation explicitly involves large matrices which may reduce the efficiency of the proposed method. A solution to overcome this is to rigorously separate all computation into two stages. The first stage can be expensive but must be independent of μ so that it can serve as a preparation step and the derived data can be used for any value of μ. The second step, where one has to compute the ROM at any new value of μ in the parameter domain, must be fast. A criterion for being fast is that its computational complexity is independent of large order n. This approach is mentioned as offline-online decomposition and quite well-known in the reduced basis community [16,9]. Details are presented in the next subsection. We also would like to drive the reader's attention to a related result

2.3 Offline-online decomposition

To enable the online-offline decomposition, we need an assumption that the matrices of system (1) depend affinely on the parameter p. More precisely,

$$E(\mu) = \sum_{i=1}^{q_E} f_i^E(\mu) E_i, \quad A(\mu) = \sum_{i=1}^{q_A} f_i^A(\mu) A_i,$$
$$B(\mu) = \sum_{i=1}^{q_B} f_i^B(\mu) B_i, \quad C(\mu) = \sum_{i=1}^{q_C} f_i^C(\mu) C_i,$$

where q_E, q_A, q_B, q_C are small and the evaluations of $f_i^E, f_i^A, f_i^B, f_i^C$ are cheap. Once the interpolated gramians are available, it follows that

$$\begin{aligned} Y^T(\mu)E(\mu)X(\mu) &= \begin{bmatrix} \sqrt{w_1(\mu)}Y_1^T \\ \cdots \\ \sqrt{w_q(\mu)}Y_q^T \end{bmatrix} \sum_{i=1}^{q_E} f_i^E(\mu) E_i \begin{bmatrix} \sqrt{w_1(\mu)}X_1 & \cdots & \sqrt{w_q(\mu)}X_q \end{bmatrix} \\ &= \sum_{i=1}^{q_E} f_i^E(\mu) \begin{bmatrix} w_1(\mu)Y_1^T E_i X_1 & \cdots & \sqrt{w_1(\mu)w_q(\mu)}Y_1^T E_i X_q \\ \vdots & \vdots & \vdots \\ \sqrt{w_k(\mu)w_1(\mu)}Y_q^T E_i X_1 & \cdots & w_k(p)Y_q^T E_i X_q \end{bmatrix}. \end{aligned} \tag{10}$$

Obviously, all $q_E q^2$ blocks $Y_\alpha^T E_i X_\beta \quad \alpha, \beta = 1, \cdots, q, i = 1, \cdots, q_E$, can be computed, stored and used for any values of μ since they are independent of μ. After computing the SVD of (10) as in (5), the projection matrices for the reduction are given by

$$W^T(\mu) = \Sigma_1^{-1/2}(\mu) U_1^T(\mu) \begin{bmatrix} \sqrt{w_1(\mu)}Y_1^T \\ \cdots \\ \sqrt{w_q(\mu)}Y_q^T \end{bmatrix},$$
$$T(p) = \begin{bmatrix} \sqrt{w_1(\mu)}X_1 & \cdots & \sqrt{w_k(\mu)}X_q \end{bmatrix} V_1(p) \Sigma_1^{-1/2}(p).$$

The reduced matrices are then computed in the same manner like (10)

$$\tilde{E}(\mu) = W^T(\mu)E(\mu)T(\mu) = \sum_{i=1}^{q_E} f_i^E(\mu)\Sigma_1^{-1/2}(\mu)U_1^T(\mu)\times$$
$$\begin{bmatrix} w_1(\mu)Y_1^T E_i X_1 & \cdots & \sqrt{w_1(\mu)w_q(\mu)}Y_1^T E_i X_q \\ \vdots & \vdots & \vdots \\ \sqrt{w_k(\mu)w_1(\mu)}Y_q^T E_i X_1 & \cdots & w_k(\mu)Y_q^T E_i X_q \end{bmatrix} V_1(\mu)\Sigma_1^{-1/2}(\mu), \tag{11}$$
$$\tilde{A}(\mu) = W^T(\mu)A(\mu)T(\mu) = \sum_{i=1}^{q_A} f_i^A(\mu)\Sigma_1^{-1/2}(\mu)U_1^T(\mu)\times$$
$$\begin{bmatrix} w_1(\mu)Y_1^T A_i X_1 & \cdots & \sqrt{w_1(\mu)w_q(\mu)}Y_1^T A_i X_q \\ \vdots & \vdots & \vdots \\ \sqrt{w_k(\mu)w_1(\mu)}Y_q^T A_i X_1 & \cdots & w_k(\mu)Y_q^T A_i X_q \end{bmatrix} V_1(\mu)\Sigma_1^{-1/2}(\mu), \tag{12}$$
$$\tilde{B}(\mu) = W^T(\mu)B()\mu) = \sum_{i=1}^{q_B} f_i^B(\mu)\Sigma_1^{-1/2}(\mu)U_1^T(\mu) \begin{bmatrix} \sqrt{w_1(\mu)}Y_1^T B_i \\ \vdots \\ \sqrt{w_q(\mu)}Y_q^T B_i \end{bmatrix}, \tag{13}$$
$$\tilde{C}(\mu) = C(\mu)T(\mu)$$
$$= \sum_{i=1}^{q_C} f_i^C(\mu)[\sqrt{w_1(\mu)}C_i X_1 \cdots \sqrt{w_q(\mu)}C_i X_q]V_1(\mu)\Sigma_1^{-1/2}(\mu). \tag{14}$$

Again, all matrix blocks, that are independent of μ, can be computed and stored before hand. Based on the above analyses, the offline-online procedure is given below.

Offline Do the following
- Solve Lyapunov equations (3) and (4) for all X_j and $Y_j, j = 1, \cdots, q$.
- Compute and store all the parameter independent matrix blocks mentioned in (10)-(14).

Online Given a value μ for the parameter
- Assemble precomputed matrix blocks and compute the SVD of (10).
- Assemble precomputed matrix blocks and compute the reduced matrices (11)-(14).

Assume that all approximate solutions of Lyapunov equations have rank l_r, one can see that the computational complexity of the online stage is scaled with $\mathcal{O}(rk^2l_r^2)$.

perhaps better with an algorithm?

3 Manifold $\mathcal{S}_+(k,n)$ and its interpolation scheme

3.1 A quotient geometry of $\mathcal{S}_+(k,n)$

3.2 Curve and surface interpolation for parametric model order reduction

3.3 A note on embedded geometry of $\mathcal{S}_+(k,n)$

In the introduction, we mentioned both embedded and quotient geometries. I just want to explain here why we don't use the embedded one for interpolation. Please give opinion on the necessity of this subsection.

Might be good to know: In an unpublished work, I and Tatjana have developed all necessary tools for interpolation using this geometry. What we did is the simple scheme: interpolating all in one tangent space. For widespread data, we encountered the problem that retraction and/or the lifting map are not defined. This is the result of high curvature of the model where we roughly showed in one lemma that it is the smallest singular value. For this reason, we did not invest more time for it. For the interpolation scheme presented in section 3, we use quotient geometry whose curvature, if I understand correctly, is the square root of the smallest singular value which is usually larger than itself in our setting. In addition, the interpolation method is more complicated and designed to avoid this problem. As a result, we don't face any problem. However, we don't know this good behavior coming from either both or one of the mentioned facts unless we try this interpolation for embedded geometry.

Worth to invest more time or better to ignore?

4 Numerical examples

We consider in this section two numerical examples. Before going into detail, we would like to discuss some details and report the general setting. First, the choice for positive weight coefficients used in algebraic approach. Options are weights based on distance from the test point to training points and linear splines. Our tests revealed that the latter delivers smaller error. Moreover, since we have to gather all data to make a big matrix in this method, see (8), too much data may result in inefficiency. Therefore, we only use linear spline in the numerical test. That is, instead of q matrix blocks in each factor of (8), there are only two (resp., four) of them for model with one (resp., two) parameter(s) regardless of the number of training points. One advantage of this fact is that if we want more accuracy by increasing the number of training points, more computation will be required in the offline stage but this makes no changes in the online stage; with the same argument, this local interpolation is much less affected by the so-called curse of dimensionality when the number of parameters increases compared to the conventional approach. (Geometric method may have this advantage; I will check it carefully and add details.)

For viewing the numerical accuracy, we compute an approximate $\mathcal{H}_\infty$-norm of the absolute errors in the frequency response defined as

$$\begin{aligned} \|H(\cdot,\mu)-\tilde{H}(\cdot,\mu)\|_{\mathcal{H}_\infty} &= \sup_{\omega\in\mathbb{R}} \|H(i\omega,\mu)-\tilde{H}(i\omega,\mu)\|_2 \\ &\approx \sup_{\omega_j\in[\omega_{\min},\omega_{\max}]} \|H(i\omega_j,\mu)-\tilde{H}(i\omega_j,\mu)\|_2, \end{aligned} \tag{15}$$

where $H(s,\mu)=C\big(sE-A(\mu)\big)^{-1}B$ and $\tilde{H}(s,\mu)=\tilde{C}\big(s\tilde{E}-\tilde{A}(\mu)\big)^{-1}\tilde{B}$ are the transfer functions of the original and the reduced-order system.

For the reference of efficiency, all computations are performed with MATLAB R2018a on a standard desktop using 64-bit OS Windows 10, equipped with 3.20 GHz 16 GB Intel Core i7-8700U CPU.

4.1 A model for heat conduction in solid material

This model is adapted from the one used in [10]. Consider the heat equation

$$\begin{aligned} \frac{\partial\vartheta}{\partial t}-\nabla(\sigma(\xi)\nabla\vartheta) &= f \quad \text{in} \quad \Omega\times(0,T), \\ \vartheta &= 0 \quad \text{on } \partial\Omega\times(0,T), \end{aligned} \tag{16}$$

with the heat conductivity coefficient

$$\sigma(\xi)=\begin{cases} 1+\mu_i & \text{for } \xi\in D_i,\ i=1,\dots,2, \\ 1 & \text{for } \xi\in\Omega\backslash(\cup_{i=1}^2 D_i), \end{cases} \tag{17}$$

where $D_i\subset\Omega=(0,4)^2$, $i=1,\dots,2$, are two discs of radius 0.5 centered at $(1,1)$,and $(3,3)$, respectively, and the parameter $\mu=[\mu_1,\mu_2]^T$ varies in $\mathcal{D}=[1,10]\times[4,10]$. Equation (16) with the source term $f\equiv 1$ is discretized using the finite element method with piecewise linear basis functions resulting in a system (1) of dimension $N=1580$ with the symmetric positive definite mass matrix $E(\mu)\equiv E$ and the stiffness matrix

$$A(\mu)=\mu_1A_1+\mu_2A_2+A_3, \tag{18}$$

where A_i, $i=1,2$, are symmetric negative semi-definite, and A_3 is symmetric negative definite. The input matrix $B(\mu)\equiv B\in\mathbb{R}^N$ originates from the source function f, and the output matrix $C(\mu)\equiv C=1/N[1,\dots,1]\in\mathbb{R}^{1\times N}$. The data were provided by the authors of [10] for which we would like to thank .

First, we fix a uniform grid $\mu_1,\dots,\mu_q\in\mathcal{D}$, which will be specified in the caption of error figures. At those points, we solve (3) and (4) using the low-rank ADI method [11] with a prescribed tolerance $1\mathrm{e}-10$. We end up with local approximate solutions whose rank varies from 25 to 27. In order to apply the presented interpolation on manifold, we slightly truncate them to make them all of rank 25. Then, the working manifold in this case is $\mathcal{S}_+(25,1580)$. Note that for method presented in section 2, local solutions at training points do not necessarily have the same rank.

The pre-computed solutions are then employed as data to interpolate for evaluate the two gramians which in turn are use to construct the ROM at test points. To decide the reduced order, we use the criterion that $\sigma_r(\mu)/\sigma_1(\mu)<1\mathrm{e}-8$ which

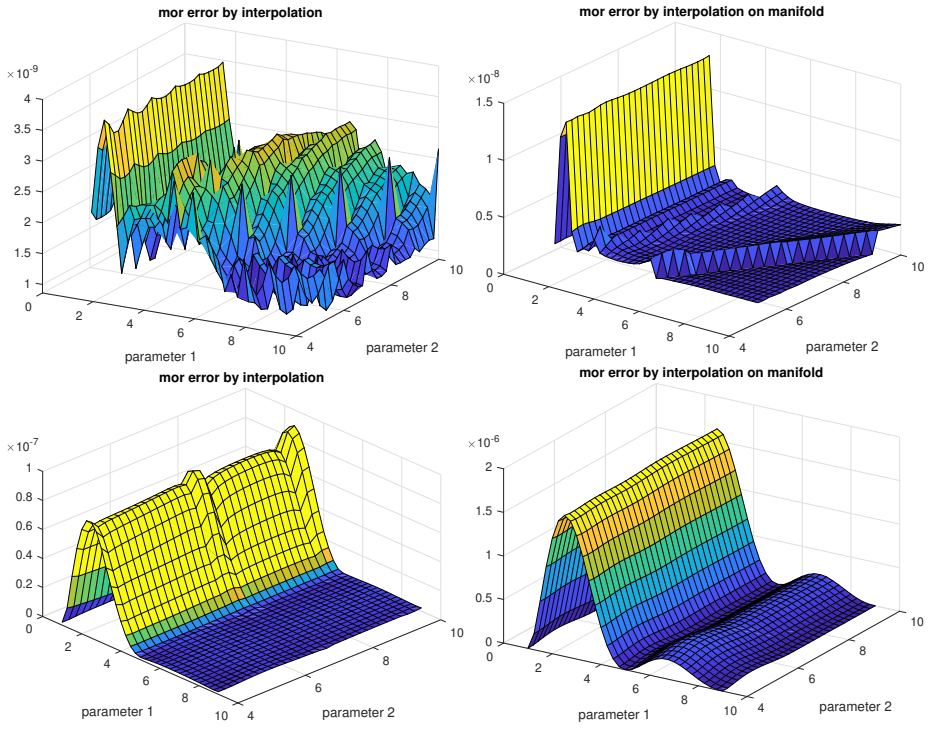

Fig. 1 The heat conduction model equation: errors of ROMs at test points. Data for top figures: training grid $\mu_1 = 1 : 1 : 10, \mu_2 = 5 : 1 : 10$ and test grid $\mu_1 = 1 : 0.25 : 10, \mu_2 = 5 : 0.2 : 10$; data for bottom figures: training grid $\mu_1 = 1 : 4 : 9, \mu_2 = 4 : 3 : 10$ and test grid $\mu_1 = 1 : 0.25 : 9, \mu_2 = 4 : 0.2 : 10$. The left figures present the error of the ROMs produced by the algebraic method and the right figures present that by the geometric method.

varies r between 12 to 15 at different test points. Finally, we plot the approximate absolute errors with respect to $\mathcal{H}_\infty$-norm as defined in (15). For ease of reading numerical results as figures, we simply choose the set o test points as a finer grid of the training grid which will be specified later. Two situations are set and the corresponding results are showed in Figure 1. It can be observed that, in the same setting, the algebraic method delivers a slightly smaller error than the geometric one. This is most probably due to the truncation step to make all the local gramians have the same rank. Moreover, the figures show that the error corresponding to small μ_1 tends to be larger. This suggests that we should use more interpolation data in this area. To this end, we try an adaptively finer grid for the algebraic method and derived the result as showed in Figure 2 left. Finally, to give the reader a view on the relative errors of the method, we plot the $\mathcal{H}_\infty$-norm of the full-order transfer function in Figure 2 right.

We now report the time consumed by the two proposed methods. We will use the first setting that produced the errors as showed in Figure 2 top. First, solving two Lyapunov equations at 60 training points needs ..... sec. Then, interpolation of low-rank solutions of these two equations using the geometric approach at 926 test points costs .... sec. From the difference in time consumed, clearly this method can

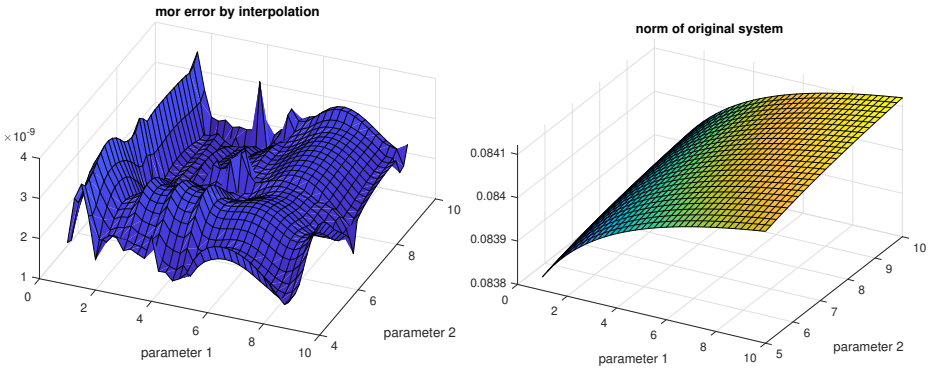

Fig. 2 The heat conduction model equation: errors of ROMs at test points. The left figure presents the error with adaptive grid $\mu_1 = [1\ 2\ 3\ 4\ 5\ 9], \mu_2 = 4 : 3 : 10$. The right figure indicates the norm of the full-order transfer function on the parameter domain.

be a good candidate for quickly computing the solutions of parametric Lyapunov equations.

I will add later!

tried algebraic approach with compression/projection

4.2 An anmometer model

In this second example, we want to verify the numerical behavior of the proposed methods when applied to fairly large problem. To this end, we consider now a model for a thermal based flow sensor; see [14] and references therein. Simulation of this device requires solving a convection-diffusion partial differential equation of the form

$$\rho c \frac{\partial \vartheta}{\partial t} = \nabla(\kappa \nabla \vartheta) - \rho c v \nabla \vartheta + \dot{q}, \tag{19}$$

where ρ denotes the mass density, c is the specific heat, κ is the thermal conductivity, v is the fluid velocity, ϑ is the temperature, and $\dot{q}$ is the heat flow into the system caused by the heater. The considered model is restricted to the case $\rho = 1, c = 1, \kappa = 1$ and $v \in [0, 1]$ which corresponds to the 1-parameter model. The finite element discretization of (19) leads to system (1) of order $N = 29008$ with the symmetric positive definite mass matrix $E(\mu) = E$ and the stiffness matrix $A(\mu) = A_1 + vA_2$, where A_1 is symmetric negative definite, A_2 is non-symmetric but negative semi-definite. The input matrix $B \in \mathbb{R}^N$ and the output matrix $C \in \mathbb{R}^{1\times N}$ are parameter-independent. The reader is referred to [20] and references therein for more detailed descriptions and the numerical data.

For this model, we use the training grid as $\mu = v = 0 : 0.1 : 1$ while the test grid is randomly generated within the range of the domain. The tolerance for the low-rank ADI solver and that for balancing truncation are chosen as $1e-8$. This setting results in ROMs with different reduced orders at test points: the one produced by the geometric approach has rank between 11 and 18 while that by algebraic approach is from 18-32. Error is visualized in Fig. 3. A closer look at

Solving two Lyapunov equations at 11 grid points: 181	
Geometric approach	Algebraic approach
	Offline: 21
Interpolation on $\mathcal{S}(k,n)$: 11.5 Compute ROMs: 0.3	Online: 0.03

Table 1 The 1-parameter anemometer model: Time consumed by different tasks (sec.)

the figure may imply, that the error of the geometric approach tend to be smaller when the test point is far from the training points and vice versa.

The time consumed by different tasks are summarized in Table 1.

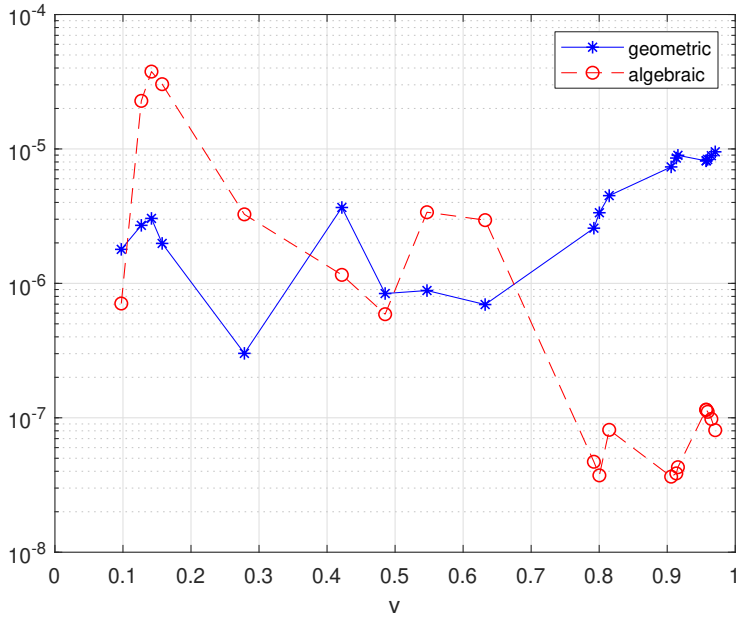

Fig. 3 The 1-parameter anemometer model: errors of ROMs at test points.

5 Conclusion

References

1. Allasia, G.: Simultaneous interpolation and approximation by a class of multivariate positive operators. NumerAlg **34**, 147–158 (2003)
2. Amsallem, D., Farhat, C.: Interpolation method for adapting reduced-order models and application to aeroelasticity. AIAAJ **46**(7), 1803–1813 (2008)
3. Amsallem, D., Farhat, C.: An online method for interpolating linear reduced-order models. SIAM J. Sci. Comput. **33**(5), 2169–2198 (2011)
4. Antoulas, A.: Approximation of Large-Scale Dynamical Systems. SIAM, Philadelphia, PA (2005). DOI 10.1137/1.9780898718713

5. Antoulas, A., Sorensen, D., Zhou, Y.: On the decay rate of the Hankel singular values and related issues. Systems Control Lett. **46**(5), 323–342 (2002)
6. Baur, U., Benner, P.: Modellreduktion für parametrisierte Systeme durch balanciertes Abschneiden und Interpolation. at-Automatisierungstechnik **57**(8), 411–422 (2009). DOI 10.1524/auto.2009.0787
7. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Review **57**(4), 483–531 (2015)
8. Degroote, J., Vierendeels, J., Willcox, K.: Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis. Int. J. Numer. Meth. Fl. **63**, 207–230 (2010)
9. Hesthaven, J., Rozza, G., Stamm, B.: Certified Reduced Basis Methods for Parametrized Partial Differential Equations. SpringerBriefs in Mathematics. Springer, Cham (2016). DOI 10.1007/978-3-319-22470-1
10. Kressner, D., Plešinger, M., Tobler, C.: A preconditioned low-rank CG method for parameter-dependent Lyapunov matrix equations. Numer. Linear Algebra Appl. **21**(5), 666–684 (2014). DOI https://doi.org/10.1002/nla.1919
11. Li, J.R., White, J.: Low rank solution of Lyapunov equations. SIAM J. Matrix Anal. Appl. **24**(1), 260–280 (2002). DOI 10.1137/S0895479801384937
12. Massart, E., Absil, P.A.: Quotient geometry with simple geodesics for the manifold of fixed-rank positive-semidefinite matrices. Technical Report UCL-INMA-2018.06, Université catholique de Louvain, Louvain-la-Neuve, Belgium, (2018)
13. Moore, B.: Principal component analysis in linear systems: controllability, observability, and model reduction. IEEE Trans. Automat. Control **AC-26**(1), 17–32 (1981)
14. Moosmann, C., Rudnyi, E.B., Greiner, A., Korvink, J.G., Hornung, M.: Parameter preserving model order reduction of a flow meter. In: Technical Proceedings of the 2005 Nanotechnology Conference and Trade Show (Nanotech 2005, Anaheim, California, USA), vol. 3, pp. 684–687. NSTINanotech (2005)
15. Panzer, H., Mohring, J., Eid, R., Lohmann, B.: Parametric model order reduction by matrix interpolation. at – Automatisierungstechnik **58**(4), 475–484 (2010)
16. Patera, A.T., Rozza, G.: Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations. MIT Pappalardo Graduate Monographs in Mechanical Engineering. MIT, MA (2007)
17. Penzl, T.: Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case. Systems Control Lett. **40**(2), 139–144 (2000)
18. Son, N.: Interpolation based parametric model order reduction. Ph.D. thesis, Universität Bremen, Germany (2012)
19. Son, N., Stykel, T.: Model order reduction of parameterized circuit equations based on interpolation. Adv. Comput. Math. **41**(5), 1321–1342 (2015)
20. The MORwiki Community: Anemometer. MORwiki – Model Order Reduction Wiki (2018). URL http://modelreduction.org/index.php/Anemometer
21. Tombs, M.S., Postlethweite, I.: Truncated balanced realization of a stable non-minimal state-space system. Internat. J. Control **46**(4), 1319–1330 (1987). DOI https://doi.org/10.1080/00207178708933971
22. Vandereycken, B., Absil, P.A., Vandewalle, S.: Embedded geometry of the set of symmetric positive semidefinite matrices of fixed rank. In: Proceedings of the IEEE 15th Workshop on Statistical Signal Processing *(Washington, DC)*, pp. 389–392. IEEE (2009). DOI https://doi.org/10.1109/SSP.2009.5278558