POD-Galerkin reduced-order modeling with adaptive finite element snapshots[☆]

Sebastian Ullmann[a,b], Marko Rotkvic[a], Jens Lang[a,b]

[a] *Department of Mathematics, Technische Universität Darmstadt, Dolivostr. 15, D-64293 Darmstadt, Germany*

[b] *Graduate School Computational Engineering, Technische Universität Darmstadt, Dolivostr. 15, D-64293 Darmstadt, Germany*

Abstract

We consider model order reduction by proper orthogonal decomposition (POD) for parametrized partial differential equations, where the underlying snapshots are computed with adaptive finite elements. We address computational and theoretical issues arising from the fact that the snapshots are members of different finite element spaces. We propose a method to create a POD-Galerkin model without interpolating the snapshots onto their common finite element mesh. The error of the reduced-order solution is not necessarily Galerkin orthogonal to the reduced space created from space-adapted snapshot. We analyze how this influences the error assessment for POD-Galerkin models of linear elliptic boundary value problems. As a numerical example we consider a two-dimensional convection-diffusion equation with a parametrized convective direction. To illustrate the applicability of our techniques to non-linear time-dependent problems, we present a test case of a two-dimensional viscous Burgers equation with parametrized initial data.

Keywords: proper orthogonal decomposition, adaptive finite elements, model order reduction, reduced basis method

1. Introduction

Model order reduction is a tool to decrease the computational cost for applications where a parametrized PDE problem needs to be solved multiple times for different parameter values. Therefore model order reductions is often studied in the context of optimal control [7, 14, 19] or uncertainty quantification [5, 8, 20]. Snapshot-based model order reduction requires a set of representative samples of the solution, which need to be computed in advance. The solution of the reduced-order model is then represented as a linear combination of these snapshots. The respective coefficients are determined by means of a Galerkin

[☆]This work was supported by the Graduate School Computational Engineering, Technische Universität Darmstadt.

projection, based on a weak form of the governing equations. In this way, the reduced-order model inherits both the spatial structure of typical solutions as well as the underlying physics. Introductions to snapshot-based model order reduction are provided by the textbooks [9, 16].

Standard techniques for model order reduction assume that all snapshots use one and the same spatial mesh. Combining adaptive meshing with model order reduction, however, can be an advantage from two points of view: Firstly, introducing spatial adaptivity to the set-up phase of a reduced-order model can decrease the total computation time if the solution contains local features depending on the parameter. By adapting the mesh to the features at a given parameter value, less degrees of freedom are required to obtain a certain accuracy. Secondly, introducing model order reduction to space-adaptive simulations promises computational speed-up in cases where the solution is susceptible to an approximation in a low-dimensional linear space and needs to be evaluated for multiple different parameter values.

Model order reduction with spatial adaptivity has been studied in [1, 2] for snapshot computations with adaptive wavelets and in [21] for snapshot computations with adaptive mixed finite elements. The main issue addressed in these publications is the assessment of the error between the reduced-order solution and the infinite-dimensional true solution. In the case of static snapshot computations, this problem can be circumvented by assuming a sufficiently fine snapshot discretization. Then the error between the reduced-order solution and a corresponding discrete solution can be estimated with the help of the discrete residual. For the case of adaptive snapshot computations, [1, 2] use wavelet techniques to estimate the required dual norm of the continuous residual. In contrast, [21] derives a bound for the dual norm of the continuous residual from a special mixed finite element and reduced basis formulation.

The references [1, 2, 21] focus on model order reduction by the greedy reduced basis method [15]. In this paper, however, we consider an alternative approach, namely proper orthogonal decomposition (POD) [10, 17]. The major difference between both methods lies in the construction of the reduced space used as a test and trial space in a Galerkin procedure. Both methods require a fixed set of training parameters to be chosen in advance. From these, the greedy reduced basis method selects a set of parameter values in an iterative way using an error estimator and uses the span of the corresponding snapshots as a reduced space. This can save computation time by avoiding the computation of snapshots which have not been selected. In contrast, POD forms a reduced basis by linearly combining the snapshots corresponding to all training parameter values. The linear combination is done in a way which minimizes the approximation error between the snapshots and their reduced-order approximations. If the dimension of the reduced spaces are increased, both the greedy and the POD space eventually become equal to the span of the snapshots corresponding to all training parameter values. A POD of snapshots resulting from a static spatial discretization can implemented in terms of a truncated singular value decomposition [11]. Therefore, an error estimator is not necessary for creating a POD reduced basis. Because snapshots have to be computed for all training parameters, POD is often applied to time-dependent problems, where snapshot data arise as a by-product of the numerical time stepping scheme. We note that in this context, POD with time-adaptive snapshots has been studied in [3]. POD with one-dimensional space-adaptive snapshots has already been addressed in [13], where the POD computation relies on a polynomial approximation of the snapshots. In contrast, we focus on the two-dimensional case and present a method which does not require an intermediate approximation of the snapshots.

A major difference between greedy and POD reduced basis methods for space adaptive snapshots is caused by the relation between the snapshots and the reduced basis functions: In the greedy reduced basis method, the reduced space is formed by linear combinations of snapshots, while the snapshots are themselves elements of the reduced space. In the POD reduced basis method, the reduced space is also formed by linear combinations of snapshots, but the snapshots are not elements of the reduced space, in general. The difference between the snapshots and their closest approximation in the POD space can be measured in terms of the truncated singular values. This has consequences for the error assessment of POD-Galerkin schemes in presence of space-adapted snapshots. While the main difficulties in the greedy reduced basis method arise from the fact that the error of the reduced-order solution is not necessarily orthogonal to the reduced space anymore, the POD reduced basis method is additionally subject to a truncation error.

This paper is structured as follows: In section 2, we introduce proper orthogonal decomposition for adaptive finite element snapshots. We propose methods to efficiently compute POD bases for adaptive finite element discretizations with nested refinement. A POD-Galerkin reduced-order model based on adaptive snapshots is formulated in section 3 for an elliptic boundary value problem. We prove error statements for the reduced-order solution in presence of adaptivity and compare the results to the static case. The methods and analytic results are illustrated in section 4 with a numerical test case involving a linear convection-diffusion equation with parametrized convective direction. The applicability to non-linear time-dependent problems is suggested by the results of section 5, which features a Burgers problem with parametrized initial condition.

2. Proper orthogonal decomposition

We consider snapshot-based model order reduction, where the solution of a PDE problem is represented in the space spanned by a set of reduced basis functions obtained by linearly combining a set of snapshots. Such reduced-basis functions typically have a global support and contain information about expected spatial structures of the solution.

One method to compute reduced basis functions from snapshots is the proper orthogonal decomposition [10, 17]. If the snapshots correspond to coefficient vectors of a finite element discretization on a fixed grid, a POD can be given in terms of the snapshot coefficient matrix [11]. In the following, we introduce a method that computes a POD of snapshots stemming from an adaptive finite element simulation, where a snapshot matrix can not be created in straightforward manner.

2.1. Method of snapshots

We consider a PDE problem defined over some bounded open spatial domain Ω and some time and/or parameter domain S. In particular, we are interested in parametrized elliptic boundary value problems, where S is a parameter domain, and in parametrized parabolic initial boundary value problems, where S is a tensor product of a time interval and a parameter domain.

Let V be the infinite-dimensional Hilbert space used to characterize the solution as a function of space. A typical example is $V = H_0^1(\Omega)$, the Sobolev space of $L^2(\Omega)$ functions with weak first derivatives in $L^2(\Omega)$ and boundary values vanishing in the sense of traces. We denote the V-scalar product by $(\cdot,\cdot)_V$ and the V-norm by $\|\cdot\|_V$.

A proper orthogonal decomposition of snapshots $u_1, \dots, u_N \in V$ can be defined in terms of a system of minimization problems [12]: Find functions $\phi_1, \dots, \phi_N \in V$ which solve the minimization problems

$$\min_{\phi_1,\dots,\phi_R \in V} \sum_{n=1}^{N} \Big\| u_n - \sum_{k=1}^{R} (u_n, \phi_k)_V \phi_k \Big\|_V^2, \quad (\phi_i, \phi_j)_V = \delta_{ij}, \quad i, j = 1, \dots, R \tag{1}$$

for all $R = 1, \dots, N$. The solutions can be computed by an eigenvalue decomposition of the matrix containing the mutual V-inner products of $u_1, \dots, u_N$. This approach is often called the *method of snapshots* [17]. The eigenvalue problem can be written in components as follows: For given $u_1, \dots, u_N \in V$, find $\lambda \in \mathbb{R}$ and $\vec{a} = (a_1, \dots, a_N)^T \in \mathbb{R}^N$, such that

$$\sum_{j=1}^{N} (u_i, u_j)_V a_j = \lambda a_i, \qquad i = 1, \dots, N.$$

The matrix form is written as $\mathcal{G}\vec{a} = \lambda\vec{a}$, where $\mathcal{G}$ is called the snapshot Gramian. The eigenvalue decomposition of the snapshot Gramian results in eigenvalues $\lambda_1, \dots, \lambda_N \in \mathbb{R}$ and eigenvectors $\vec{a}^1, \dots, \vec{a}^N \in \mathbb{R}^N$. We order the eigenvalues such that $\lambda_1 \geq \cdots \geq \lambda_D > 0 = \lambda_{D+1} = \cdots = \lambda_N$ and write the eigenvectors in components as $\vec{a}^r = (a_1^r, \dots, a_N^r)^T$ for $r = 1, \dots, N$. Then the first POD basis functions are given as linear combinations of snapshots,

$$\phi_r = \sum_{n=1}^{N} u_n \frac{a_n^r}{\sqrt{\lambda_r}}, \qquad r = 1, \dots, D. \tag{2}$$

The space $V^R = \mathrm{span}(\phi_1, \dots, \phi_R)$ is called a POD space of dimension R for any $1 \leq R \leq D$.

POD reduced-order modeling tries to approximate a solution $u : S \to V$ with a function $u^R : S \to V^R$ defined as

$$u^R = \sum_{r=1}^{R} \phi_r b^r, \tag{3}$$

where $\vec{b} = (b^1, \dots, b^R)^T : S \to \mathbb{R}^R$ is a POD coefficient vector. Combining (2) and (3) gives a POD approximation in terms of the snapshots,

$$u^R = \sum_{n=1}^{N} u_n \sum_{r=1}^{R} \frac{a_n^r b^r}{\sqrt{\lambda_r}}, \tag{4}$$

One particular choice of POD coefficients is implied by the POD minimization problem (1) and given by a V-orthogonal projection of u onto V^R:

$$P^R u := \sum_{r=1}^{R} \phi_r (\phi_r, u)_V \quad \forall u \in V, \quad R = 1, \dots, D,$$

which means $b^r = (\phi_r, u)$ for $r = 1, \dots, R$. The orthogonal projection is used as a reference solution later on, because it gives a POD representation with optimal coefficients:

$$\|u - P^R u\|_V = \inf_{v \in V^R} \|u - v\|_V \quad \forall u \in V.$$

The error of the POD projection of the snapshots can be computed from the POD eigenvalues,

$$\sum_{n=1}^{N} \|u_n - P^R u_n\|_V^2 = \sum_{n=R+1}^{D} \lambda_n, \tag{5}$$

which implies that the POD projection error of the snapshots decreases monotonically with the POD dimension and that $u_n = P^D u_n$ for $n = 1, \dots, N$. Together with (2) this means

$$\operatorname{span}(u_1, \dots, u_N) = \operatorname{span}(\phi_1, \dots, \phi_D). \tag{6}$$

2.2. Adaptive snapshot spaces

In order to compute a set of snapshots, we discretize our PDE problem of interest with adaptive finite elements in space. Let $V_1, \dots, V_N \subset V$ be adapted finite element spaces, so that $u_1 \in V_1, \dots, u_N \in V_N$. Let $M_1, \dots, M_N$ be the dimensions of the respective spaces. We focus on h-adaptive Lagrangian finite elements with a fixed polynomial degree, so that each snapshot finite element space is defined by a triangulation.

For discretizations on a fixed triangulation, one can represent linear combinations of snapshots by linear combinations of finite element coefficient vectors. In order to do this for adaptive spatial discretizations, however, one must first express the snapshots in terms of a suitable common finite element basis. Therefore, we introduce a space $V_+ \subset V$ with finite dimension M_+, on which we impose two properties:

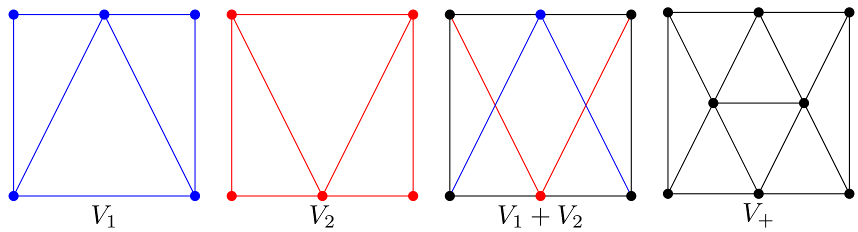

Figure 1: Illustration of meshes corresponding to general finite element spaces V_1 and V_2, their vector sum $V_1 + V_2$ and a common finite element space V_+ obtained by adding nodes and edges.

1. V_+ is a finite element space of the same type as $V_1, \dots, V_N$,
2. $V_1 + \dots + V_N \subset V_+$ in terms of a vector sum.

A consequence of the first property is that after interpolating all snapshots onto V_+, we can work with them in the same way as if they were computed on a fixed triangulation. The second property ensures that the error between any snapshot and its representation in V_+ is zero.

In general, setting $V_+ = V_1 + \dots + V_N$ would be too restrictive in the sense that it does not necessarily fulfill the first property. Consider, for example, the case where $V_1, \dots, V_N$ are linear Lagrangian finite element spaces defined over different triangulations of a common spatial domain, like in Figure 1. While the functions in $V_1 + \dots + V_N$ are still piecewise linear, they do not always correspond to a finite element discretization on a triangulation. Still, by adding degrees of freedom one can find a triangulation and a respective linear Lagrangian finite element space V_+ containing $V_1 + \dots + V_N$.

A more convenient situation is encountered if the snapshots are adapted with the newest vertex bisection algorithm starting from a common initial triangulation. It is known that the smallest common refinement of two such meshes is their overlay [4, 18], which implies $V_1 + \dots + V_N = V_+$. A sketch is given in Figure 2. Moreover, the mesh of V_+ can be found by repeated local refinements of any snapshot mesh. Consequently, to interpolate a function from V_n to V_+ for any $n = 1, \dots, N$, one only needs to interpolate the function between successive refinement steps. Because of this favorable property, we focus on refinement by newest vertex bisection in our numerical examples. However, the theory does not depend on this decision.

Besides a common finite element space of all snapshots, it will be useful to have common finite element spaces of subsets of snapshots available. To this end, we extend our notation in the following way: Let $V_{n_1}, \dots, V_{n_K}$ for $K > 1$ define a K-tuple of snapshot finite element spaces with $1 \leq n_k \leq N$ for all $k = 1, \dots, K$. We define $V_{n_1 \dots n_K}$ as a finite element space of the same type as $V_{n_1}, \dots, V_{n_K}$, with $V_{n_1} + \dots + V_{n_K} \subset V_{n_1 \dots n_K}$. As a special case we denote $V_+ = V_{1 \dots N}$.

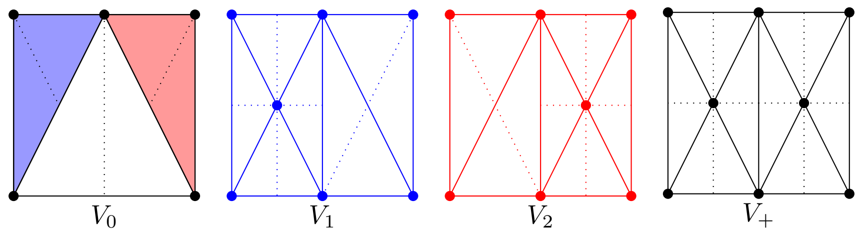

Figure 2: Illustration of meshes resulting from refinement by newest vertex refinement based on a common initial triangulation corresponding to a finite element space V_0. Dotted lines indicate the next possible refinement. Refining the upper left triangle of the initial mesh results in V_1. Refining the upper right triangle of the initial mesh results in V_2. The common finite element space V_+ equals the overlay of both refined meshes, and therefore $V_+ = V_1 + V_2$.

2.3. Gramian of adapted snapshots

The first step in the computation of a POD with the method of snapshots is the creation of the snapshot Gramian, see section 2.1. For the case of space adapted snapshots, we consider two options: The first option is to represent all snapshots as members of the common finite element space of all snapshots. The second option is to represent pairs of snapshots as members of common finite element spaces of these pairs.

At first we provide an implementation for adaptive finite element snapshots in terms of the common finite element space of *all* snapshots, where we choose $u_1, \ldots, u_N \in V_+$. We collect the finite element coefficients of the snapshots with respect to a basis of V_+ in a set of snapshot coefficient vectors $\mathcal{U}_1, \ldots, \mathcal{U}_N \in \mathbb{R}^{M_+}$ and define a snapshot matrix $\mathcal{U} = (\mathcal{U}_1, \ldots, \mathcal{U}_N) \in \mathbb{R}^{M_+ \times N}$. Let $\mathcal{M}_+$ be the matrix associated with the V-inner product of functions in V_+, so that for $u_i, u_j \in V_+$ we have $(u_i, u_j)_V = \mathcal{U}_i^T \mathcal{M}_+ \mathcal{U}_j$. Then the snapshot Gramian matrix is given by $\mathcal{G} = \mathcal{U}^T \mathcal{M}_+ \mathcal{U}$.

Now we reformulate the computation of the snapshot Gramian so that we only need to create common finite element spaces of *pairs* of snapshots. We consider the computation of a single entry of the snapshot Gramian matrix for a pair consisting of $u_i \in V_i$ and $u_j \in V_j$. Let $\mathcal{U}_i^{ij}$ and $\mathcal{U}_j^{ij}$ be the finite element coefficients of u_i and u_j with respect to a basis of their common finite element space V_{ij} and let $\mathcal{M}_{ij}$ be the matrix associated with the V-inner product of functions in V_{ij}, so that $(u_i, u_j)_V = (\mathcal{U}_i^{ij})^T \mathcal{M}_{ij} \mathcal{U}_j^{ij}$. This means that the Gramian matrix can be filled by creating finite element spaces V_{ij} of all pairs of snapshots.

We have presented two ways of creating the snapshot Gramian matrix for the eigenvalue decomposition associated with the POD method of snapshots. In any case, due to the properties of the common finite element spaces we obtain the exact Gramians. The advantage of the first method is that only a single common finite element space has to be created. A possible disadvantage is that the dimension of this space may be very high. In the second method a larger number of lower-dimensional finite element spaces must be created.

2.4. POD basis functions and approximation

The POD basis functions are determined as linear combinations of snapshots by (2). If the snapshots are represented as members of V_+, the POD basis functions are automatically members of V_+ and can be computed by linearly combining the snapshot finite element coefficient vectors corresponding to a basis of V_+. If the snapshots are represented as members of their original adapted finite element spaces $V_1, \dots, V_N$, the POD basis functions can be implicitly defined as linear combinations of snapshots. In this way, forming a basis of V_+ can be avoided, but applying a linear operator to a single POD basis function means applying this operator to all snapshots. Following this idea, there are multiple ways to represent a POD approximation: in terms of V_+, or in terms of a linear combination of POD basis functions by (3), or in terms of snapshots by (4).

The theoretical results regarding the V-orthogonal POD projection in the last paragraph of section 2.1 have been stated in the context of functions in V. Therefore, these results do not depend on whether the snapshots have been computed with a static or an adaptive discretization. Still, the V-orthogonal projection requires knowledge of the function to be projected, and is therefore only valuable as a reference.

A different scenario is the computation of POD coefficients by a reduced-order model obtained via Galerkin projection. Here, knowledge of the solution is not necessary to obtain POD coefficients. As we will see in the following section, however, the snapshot discretization influences the accuracy of the POD approximation.

3. POD Galerkin reduced-order modeling for an elliptic PDE

We highlight the principal differences between POD Galerkin reduced-order modeling for static and adapted snapshots with an example of a parametrized elliptic boundary value problem. For the case $R = D$ we can use results from the greedy reduced basis theory [1], because according to (6) the POD space equals the span of the snapshots. For $R < D$ we need to take the additional POD truncation error into account.

3.1. Weak formulation

Let $\mu \in S$ be a parameter vector in a domain $S \subset \mathbb{R}^K$. We define a parametrized bilinear form $a(\cdot, \cdot; \mu)$ which is uniformly coercive with coercivity constant

$$\alpha(\mu) = \inf_{v \in V \setminus \{0\}} \frac{a(v, v; \mu)}{\|v\|_V^2} \geq \alpha > 0$$

and uniformly continuous with continuity constant

$$\gamma(\mu) = \sup_{v, w \in V \setminus \{0\}} \frac{a(v, w; \mu)}{\|v\|_V \|w\|_V} \leq \gamma < \infty.$$

We also define a linear form $f(v;\mu)$ which is uniformly continuous with continuity constant

$$\delta(\mu) = \sup_{v\in V\setminus\{0\}} \frac{f(v;\mu)}{\|v\|_V} \leq \delta < \infty.$$

The parametrized elliptic PDE problem is now formulated as follows: For $\mu \in S$, find $u(\mu) \in V$ such that

$$a(u(\mu), v; \mu) = f(v; \mu) \quad \forall v \in V. \tag{7}$$

The Lax-Milgram theorem guarantees well-posedness of this problem under the given continuity and coercivity assumptions. Its solution is called *true solution* in the following.

3.2. Snapshot computation

To provide snapshots for the POD computation, we introduce a collection of discretized PDE problems associated with a given discrete training set $S_N = \{\mu_1, \ldots, \mu_N\}$ with $\mu_1, \ldots, \mu_N \in S$. For each $\mu \in S_N$ we solve the respective PDE problem with an adaptive discretization scheme, which leads to the snapshot spaces $V_1, \ldots, V_N \subset V$. The snapshots are solutions of the following discretized version of (7): For each $n = 1, \ldots, N$, find $u_n \in V_n$ such that

$$a(u_n, v; \mu_n) = f(v; \mu_n) \quad \forall v \in V_n.$$

For any $n = 1, \ldots, N$, the subspace property $V_n \subset V$ leads to Galerkin orthogonality between the error $u(\mu_n) - u_n$ and the discrete space V_n, which means

$$a(u(\mu_n) - u_n, v; \mu_n) = 0 \quad \forall v \in V_n.$$

We imply a Céa lemma stating

$$\|u(\mu_n) - u_n\|_V \leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u(\mu_n) - P_n u(\mu_n)\|_V, \quad n = 1, \ldots, N, \tag{8}$$

where P_n denotes the V-orthogonal projection onto the snapshot discretization space V_n, so that

$$\|u(\mu_n) - P_n u(\mu_n)\|_V = \inf_{v\in V_n} \|u(\mu_n) - v\|_V.$$

3.3. Reduced-order model and error assessment

Using the methods from section 2, we create a POD space $V^R \subset V$ from the snapshots. The respective POD-Galerkin reduced-order model of (7) is formulated as follows: For $\mu \in S$, find $u^R(\mu) \in V^R$ such that

$$a(u^R(\mu), v; \mu) = f(v; \mu) \quad \forall v \in V^R.$$

For the error assessment of u^R, we first recall the main results for static discretizations. Then we study to the adaptive case and point out major differences compared to the static case. We restrict our attention to $\mu \in S_N$. For practical applications this means that a sufficiently rich snapshot set is assumed, so that the error from discretizing S is negligible.

3.3.1. Static discretization

Assume $V^R \subset V_n$ for $n = 1, \ldots, N$. This assumption holds if the snapshots have been computed with static finite elements.

At first we study the error between the snapshots and the reduced-order solution evaluated at the corresponding training parameter values. We can derive a Galerkin orthogonality between this error and the POD space,

$$a(u_n - u^R(\mu_n), v; \mu_n) = 0 \quad \forall v \in V^R, \quad n = 1, \ldots, N.$$

The Céa lemma, following from coercivity and continuity, states the relation to the POD approximation error,

$$\|u_n - u^R(\mu_n)\|_V \leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - P^R u_n\|_V, \quad n = 1, \ldots, N.$$

Moreover, (5) implies $u^D(\mu_n) = u_n$ for $n = 1, \ldots, N$. The fact that the snapshots are recovered for large enough R is called asymptotic snapshot reproducibility.

For the error with respect to the true solution we use a triangle inequality and respective Céa lemmas to obtain

$$\begin{aligned} \|u(\mu_n) - u^R(\mu_n)\|_V &\leq \|u(\mu_n) - u_n\|_V + \|u_n - u^R(\mu_n)\|_V \\ &\leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u(\mu_n) - P_n u(\mu_n)\|_V + \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - P^R u_n\|_V \end{aligned}$$

for $n = 1, \ldots, N$. For maximum efficiency, the errors of the finite element discretization and the POD truncation should be balanced. It should be noticed, however, that the first term stems from the off-line discretization, which is only relevant for the setup of the reduced-order model, while the second term stems from the on-line discretization, which is also relevant for the evaluation time of the reduced-order model.

3.3.2. Adaptive discretization

We derive error inequalities similar to the ones in section 3.3.1, but for the more general case, where the snapshots are members of different finite element spaces. Here we do not have a Galerkin orthogonality between the reduced-order error and the reduced space available.

We start with the error between the solution of the reduced-order model and the true solution. Due to $V^R \subset V$, for any $\mu \in S$ one can derive a Galerkin orthogonality

$$a(u(\mu) - u^R(\mu), v; \mu) = 0 \quad \forall v \in V^R$$

and a corresponding Céa lemma

$$\|u(\mu) - u^R(\mu)\|_V \leq \frac{\gamma(\mu)}{\alpha(\mu)} \|u(\mu) - P^R u(\mu)\|_V.$$

We split the right-hand side of the Céa lemma into contributions from the snapshot computation and from the POD truncation. To exclude the error associated with the discretization of the parameter domain, we consider only $\mu \in S_N$. The derivation starts with adding a zero to the right-hand side of the Céa lemma for the reduced-order model and subsequently uses triangle inequalities and the properties of orthogonal projections,

$$
\begin{aligned}
\|u(\mu_n) - u^R(\mu_n)\|_V &\leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u(\mu_n) - u_n + u_n - P^R u_n + P^R u_n - P^R u(\mu_n)\|_V \\
&\leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - P^R u_n\|_V + \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|(I - P^R)(u(\mu_n) - u_n)\|_V \\
&\leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - P^R u_n\|_V + \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u(\mu_n) - u_n\|_V
\end{aligned}
$$

for $n = 1, \ldots, N$. Using (8) we obtain

$$
\|u(\mu_n) - u^R(\mu_n)\|_V \leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - P^R u_n\|_V + \frac{\gamma(\mu_n)^2}{\alpha(\mu_n)^2} \|u(\mu_n) - P_n u(\mu_n)\|_V
$$

for $n = 1, \ldots, N$. This means that for parameter values in S_N, the error between the true and the reduced-order solution can be split into contributions from the projection of the respective snapshot onto the POD space and from the projection of the true solution onto the respective snapshot finite element space. In absence of POD truncation, i.e. for $R = D$, the POD projection error vanishes and we obtain a variant of known results from greedy reduced basis theory [1].

Because in general $V^R \not\subset V_n$ for $n = 1, \ldots, N$, we are not able to derive a Céa lemma for $u_n - u^R(\mu_n)$. A straight-forward approach is using the results from above to obtain

$$
\begin{aligned}
&\|u_n - u^R(\mu_n)\|_V \\
&\qquad \leq \|u_n - u(\mu_n)\|_V + \|u(\mu_n) - u^R(\mu_n)\|_V \\
&\qquad \leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \left(1 + \frac{\gamma(\mu_n)}{\alpha(\mu_n)}\right) \|u(\mu_n) - P_n u(\mu_n)\|_V + \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - P^R u_n\|_V.
\end{aligned}
$$

Alternatively, we can use another result adapted from the literature [1]: Due to coercivity, continuity, $u_n \in V^D$ and Galerkin orthogonality between V^D and the error between the true solution and the solution of the reduced-order model we have

$$
\begin{aligned}
\alpha(\mu_n)\|u_n - u^D(\mu_n)\|_V^2 &\leq a(u_n - u^D(\mu_n), u_n - u^D(\mu_n); \mu_n) \\
&= a(u_n - u(\mu_n), u_n - u^D(\mu_n); \mu) \\
&\leq \gamma(\mu_n)\|u_n - u(\mu_n)\|_V \|u_n - u^D(\mu_n)\|_V, \quad n = 1, \ldots, N,
\end{aligned}
$$

so that

$$
\|u_n - u^D(\mu_n)\|_V \leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)} \|u_n - u(\mu_n)\|_V, \quad n = 1, \ldots, N.
$$

Because $V^R \subset V^D$ for $R \leq D$ we obtain

$$\begin{aligned}
&\|u_n - u^R(\mu_n)\|_V \\
&\qquad \leq \|u_n - u^D(\mu_n)\|_V + \|u^D(\mu_n) - u^R(\mu_n)\|_V \\
&\qquad \leq \frac{\gamma(\mu_n)}{\alpha(\mu_n)}\|u_n - u(\mu_n)\|_V + \frac{\gamma(\mu_n)}{\alpha(\mu_n)}\|u^D(\mu_n) - P^R u^D(\mu_n)\|_V \\
&\qquad \leq \frac{\gamma(\mu_n)^2}{\alpha(\mu_n)^2}\|u(\mu_n) - P_n u(\mu_n)\|_V + \frac{\gamma(\mu_n)}{\alpha(\mu_n)}\|u^D(\mu_n) - P^R u^D(\mu_n)\|_V,
\end{aligned}$$

with the help of the Céa lemma of the snapshot computation. In any case, the error between a snapshot and the solution of the reduced-order model at the corresponding parameter value contains a component from the finite element computation, which was not present for static snapshots.

4. Numerical example of a convection-diffusion equation

We apply POD model order reduction to a two-dimensional convection-diffusion problem, where the transport direction serves as a parameter and adaptive finite element snapshots are taken over the parameter interval. The test case illustrates the computational methods introduced in section 2 and the theoretical results derived in section 3.

4.1. Problem setting

We consider a parametrized boundary value problem based on a convection-diffusion equation in two dimensions,

$$v_x(\mu)\partial_x u + v_y(\mu)\partial_y u - \nu\partial_{xx} u - \nu\partial_{yy} u = 1 \quad \forall(\vec{x}, \mu) \in \Omega \times S,$$

with solution $u(\vec{x}, \mu) : \Omega \times S \to \mathbb{R}$ for some spatial domain $\Omega = (0, X) \times (0, Y)$ with $X, Y < \infty$, and some parameter interval $S \subset \mathbb{R}$. For the diffusivity constant we assume $\nu > 0$. The dependence of the convective velocity components $v_x(\mu)$ and $v_y(\mu)$ on the parameter μ is given by $v_x = \cos(0.25\pi\mu)$ and $v_y = \sin(0.25\pi\mu)$, which means that only the direction of the velocity vector is varied. At the boundary $\partial\Omega$ of the spatial domain Ω, we specify homogeneous Dirichlet conditions, so that

$$u(\vec{x}, \mu) = 0 \quad \forall(\vec{x}, \mu) \in \partial\Omega \times S.$$

We fix $V = H^1_0(\Omega)$ for this example problem. A spatial weak form of the parametrized convection-diffusion problem is given as follows: Find $u(\mu) : S \to V$, such that

$$a(u, \phi; \mu) = f(\phi) \quad \forall \phi \in V, \quad \mu \in S, \tag{9}$$

where the bilinear and linear forms are defined as

$$\begin{aligned}
a(w, \psi; \mu) &:= \int_\Omega v_x(\mu)\partial_x w\, \psi + v_y(\mu)\partial_y w\, \psi + \nu\partial_x w\, \partial_x \psi + \nu\partial_y w\, \partial_y \psi \, \mathrm{d}\vec{x}, \\
f(\psi) &:= \int_\Omega \psi \, \mathrm{d}\vec{x}.
\end{aligned}$$

4.2. Discretization

The parameter domain is discretized into a training set $S_N := \{\mu_1, \ldots, \mu_N\}$ with $\mu_1, \ldots, \mu_N \in S$. For each training parameter value, an adaptive finite element algorithm chooses an individual finite element space with respect to some fixed error tolerance. The respective discrete solutions and finite element spaces are denoted by $u_n \in V_n$ for $n = 1, \ldots, N$, so that $u_n \approx u(\mu_n)$. We assume adaptive finite discretizations for the training parameter values are given as follows: For $n = 1, \ldots, N$, find $u_n \in V_n$ such that

$$a(u_n, \phi; \mu_n) = f(\phi) \quad \forall \phi \in V_n. \tag{10}$$

The finite element discretization leads to a set of linear algebraic equations which can be solved in parallel for $n = 1, \ldots, N$.

4.3. Reduced-order modeling

Using the methods from section 2, we create a POD space $V^R \subset V$ with basis functions $\phi_1, \ldots, \phi_R$ from the discrete solutions $u_1 \in V_1, \ldots, u_N \in V_N$. Substituting V by V^R in (9) gives rise to the following reduced-order model: Find $u^R(\mu) : S \to V^R$ such that

$$a(u^R, \phi; \mu) = f(\phi) \quad \forall \phi \in V^R. \tag{11}$$

We can rewrite the model as an equation for the POD coefficient vector: Find $\vec{b}(\mu) : S \to \mathbb{R}^R$ such that

$$(A_x^R v_x(\mu) + A_y^R v_y(\mu) + A_\nu^R)\vec{b} = \vec{F}^R.$$

Expressions for the constant model coefficient matrices A_x^R, A_y^R, A_ν^R and the right-hand side vector $\vec{F}^R$ follow from substitution of the POD expansion (3) into (11) and testing against the POD basis functions.

4.4. Results

For our computations, we choose a domain with $X = Y = 1$, a diffusivity of $\nu = 0.01$ and a parameter interval of $S = [0, 1]$. For the numerical discretization, we use a custom implementation of piecewise linear Lagrangian finite elements on triangular meshes with refinement by newest vertex bisection. As an error indicator we use the Matlab PDE toolbox with a spatial tolerance of 0.008 for the error based on the viscous term. Numerical solutions for varying parameter values are presented in Figure 3. Respective adapted spatial meshes are plotted in Figure 4.

In order to generate snapshots for subsequent reduced-order modeling, we solve the problem for 33 parameter values distributed equidistantly over the parameter interval and store the respective solutions. We compute a POD of these snapshots and create Galerkin reduced-order models of varying dimension. In order to test the accuracy of the reduced-order models, we solve them for the parameter values in the training set.

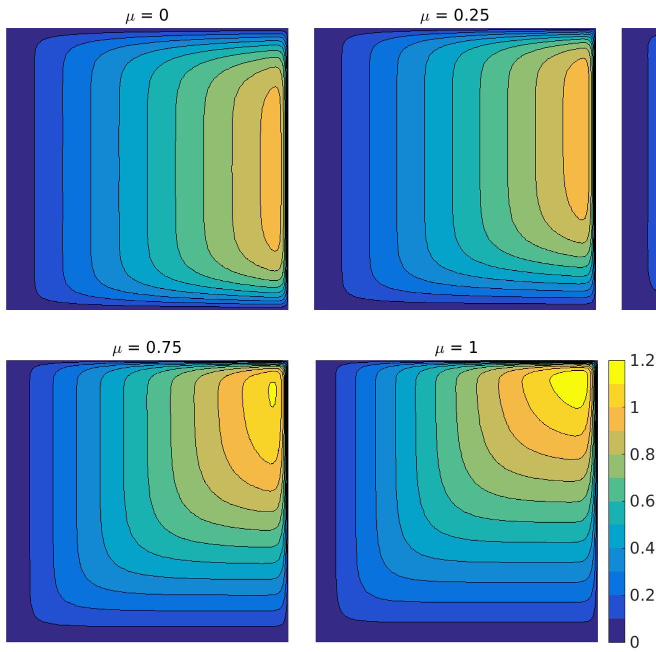

Figure 3: Solution of the convection-diffusion problem for different parameter values.

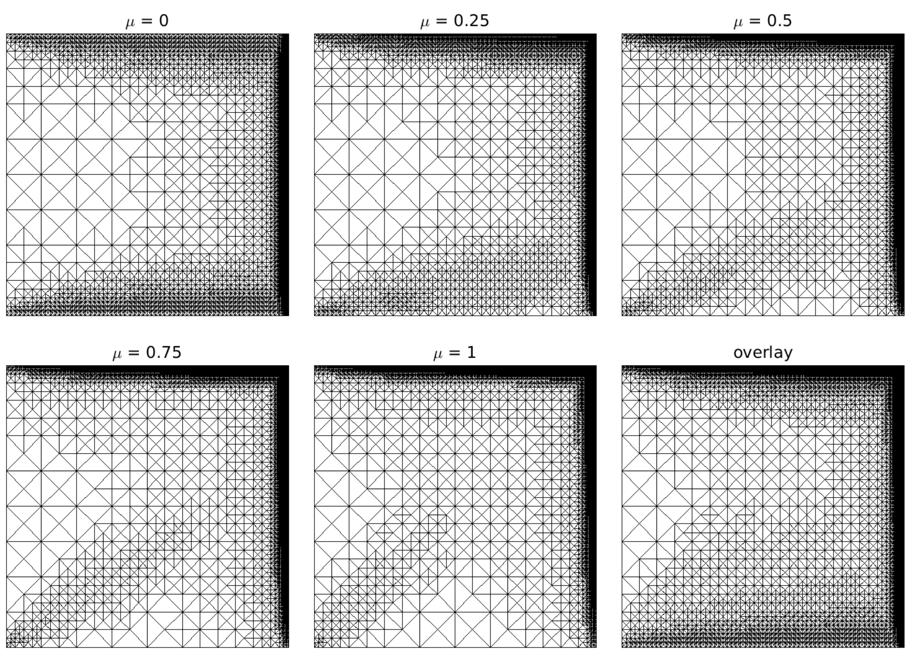

Figure 4: Adapted triangulations corresponding to Figure 3, and their overlay.

The error between the snapshots and the solution of the POD-Galerkin model is named ϵ_{ROM}. It can be viewed as a benchmark for the Galerkin reduced-order model under the assumption that sufficient snapshot data has been used to create the POD. It is measured in the relative norm induced by the POD, so that

$$\epsilon_{\mathrm{ROM}} = \sqrt{\sum_{n=1}^{N} \|u_n - u^R(\mu_n)\|_V^2} \Big/ \sqrt{\sum_{n=1}^{N} \|u_n\|_V^2}. \tag{12}$$

The error between the snapshots and their orthogonal projection on the POD is named ϵ_{POD}. It can be viewed as a benchmark for the POD in the sense that it characterizes the ability of the snapshot data to be represented in a low-dimensional space. It is measured in the relative norm induced by the POD, which enables an alternative expression in terms of the POD eigenvalues via (5):

$$\epsilon_{\mathrm{POD}} = \sqrt{\sum_{n=1}^{N} \|u_n - P^R u_n\|_V^2} \Big/ \sqrt{\sum_{n=1}^{N} \|u_n\|_V^2} = \sqrt{\sum_{n=R+1}^{D} \lambda_n} \Big/ \sqrt{\sum_{n=1}^{D} \lambda_n}. \tag{13}$$

To compare the approximation errors resulting from the model order reduction with the finite element discretization error, we computed reference snapshots $u_1^{\mathrm{ref}}, \ldots, u_N^{\mathrm{ref}}$ with a stricter spatial tolerance of 0.002. The resulting estimated finite element discretization error of the original snapshots is measured in the same norm as above, so that

$$\epsilon_{\mathrm{FEM}} = \sqrt{\sum_{n=1}^{N} \|u_n^{\mathrm{ref}} - u_n\|_V^2} \Big/ \sqrt{\sum_{n=1}^{N} \|u_n\|_V^2}. \tag{14}$$

The errors defined in (12)–(14) are plotted in Figure 5 (left). We observe that the POD projection error ϵ_{POD} decreases monotonically, which is implied by its representation in terms of the POD eigenvalues. The convergence of the error of the reduced-order model ϵ_{ROM}, however, stagnates at a level related to the finite element error of the snapshots, as suggested by the results of section 3.3.2. Still, both model order reduction errors ϵ_{POD} and ϵ_{ROM} are well below the finite element error for reasonable dimensions of the reduced space.

The same types of plots are shown in Figure 5 (right) for simulations on a fixed grid given by the overlay of all adapted snapshots. Here both the error of the POD projection and the solution of the reduced-order model converge. This can be explained by the Galerkin orthogonality property detailed in section 3.3.1. The convergence rate is higher in the case of a static finite element discretization. Our interpretation of this behavior is that in the adaptive case the POD basis functions of higher index start approximating spatial artifacts resulting from the different discretizations.

To underline our interpretation of the differences in the POD approximation error ϵ_{POD} between adaptive and static snapshots simulations, we plot the first

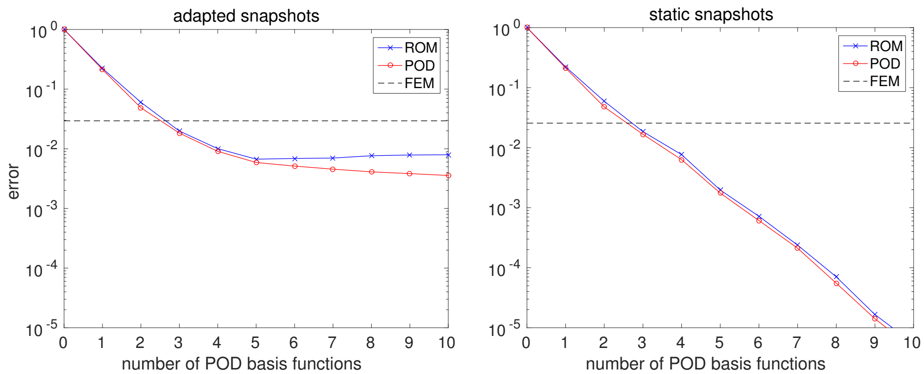

Figure 5: Error ϵ_{POD} of the POD projection, error ϵ_{ROM} of the solution of the reduced-order model, and finite element discretization error ϵ_{FEM}, corresponding to (12)–(14). Left: adapted snapshots, right: static snapshots computed on the common grid of the space-adapted snapshots.

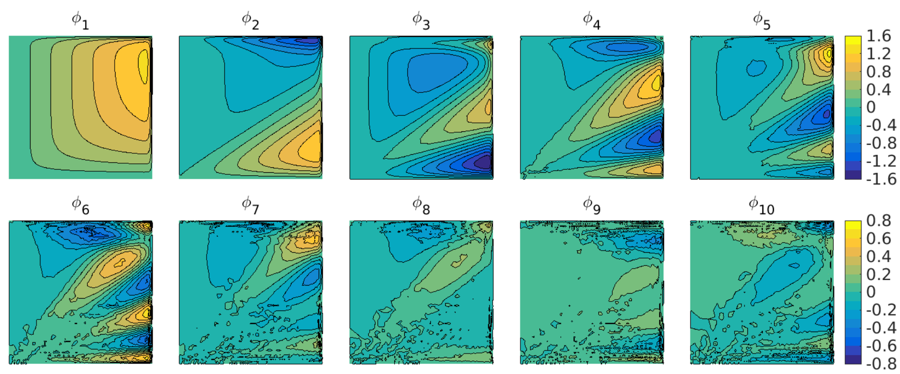

Figure 6: First 10 POD basis functions resulting from 33 adapted snapshots.

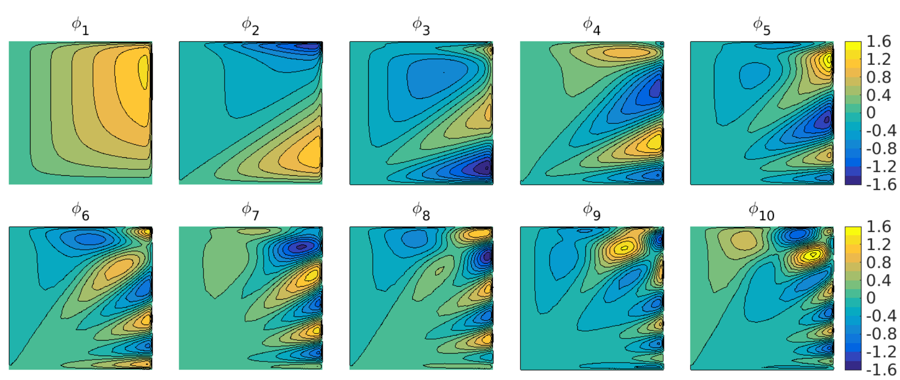

Figure 7: First 10 POD basis functions resulting from 33 static snapshots.

10 POD basis functions resulting from the adapted snapshots in Figure 6. We observe that the POD basis functions corresponding to the adaptive simulation start exhibiting local variations in the size of typical mesh cells when the index is increased. For comparison, we present the first 10 POD basis functions resulting from the static snapshots in Figure 7. It is reasonable that the oscillations visible in these plots are necessary to approximate the parameter-dependent physical structures of the solution with increasing accuracy, see Figure 3.

Despite this qualitative differences in the appearance of the POD basis functions of higher index, we stress that in this example including more than 4 basis functions does not decrease the total error of the reduced-order solution. This is because the POD errors are dominated by the finite element approximation error for $R \geq 4$ in both the adaptive and the static case, as obvious from Figure 5.

5. Numerical example of a Burgers equation

To illustrate the potential of POD-Galerkin modeling with adaptive snapshots for non-linear time-dependent problems, we apply our techniques to a Burgers equation. Adaptive finite element snapshots are taken over the time and parameter domain. The test case suggests that the theoretical results derived in section 3 for a linear elliptic setting can be transferred to a parabolic setting.

5.1. Problem setting

We consider a parametrized initial-boundary value problem based on a scalar viscous Burgers equation in two dimensions,

$$\partial_t u + u\partial_x u - \nu\partial_{xx} u - \nu\partial_{yy} u = 0 \quad \forall (t, \vec{x}, \mu) \in I \times \Omega \times S,$$

with solution $u(t, \vec{x}, \mu) : I \times \Omega \times S \to \mathbb{R}$ for some time interval $I = (0, T]$ with $T < \infty$, some spatial domain $\Omega = (0, X) \times (0, Y)$ with $X, Y < \infty$, and some parameter interval $S \subset \mathbb{R}$. A parametrized initial condition is given by

$$u(0, \vec{x}, \mu) - u_0(\vec{x})\mu = 0 \quad \forall (\vec{x}, \mu) \in \Omega \times S$$

with spatial data $u_0(x) : \Omega \to \mathbb{R}$. At the boundary we choose a combination of homogeneous Neumann and periodic conditions, so that

$$\begin{aligned} u(t, (0, y)^T, \mu) - u(t, (X, y)^T, \mu) = 0 \quad &\forall (t, y, \mu) \in I \times (0, Y) \times S, \\ \partial_y u(t, (x, 0)^T, \mu) = \partial_y u(t, (x, Y)^T, \mu) = 0 \quad &\forall (t, x, \mu) \in I \times (0, X) \times S. \end{aligned}$$

We assume that $\nu > 0$. Note that for the inviscid Burgers equation with $\nu = 0$ we would have $u(t, \vec{x}, \mu) = \mu u(t\mu, \vec{x}, 1)$.

In the context of section 5, we define $V \subset H^1(\Omega)$ as the Sobolev space of $L^2(\Omega)$ functions which have weak first derivatives in $L^2(\Omega)$ and which satisfy the periodic boundary conditions in a weak sense. A spatial weak form of the parametrized Burgers problem is then given as follows: For $u_0 \in L^2(\Omega)$, find $u(t,\mu) : I \times S \to V$, such that

$$\begin{aligned} (u(0,\mu) - u_0\mu, \phi) &= 0 \quad \forall \phi \in V, \quad \mu \in S, \\ (\partial_t u, \phi) + a(u,\phi) + b(u,u,\phi) &= 0 \quad \forall \phi \in V, \quad (t,\mu) \in I \times S, \end{aligned}$$

where $(\cdot,\cdot)$ denotes the $L^2(\Omega)$-inner product and the bilinear and trilinear forms are defined as

$$a(v,\psi) := \nu \int_\Omega \partial_x v \, \partial_x \psi + \partial_y v \, \partial_y \psi \, \mathrm{d}\vec{x}, \qquad b(v,w,\psi) := \int_\Omega v \partial_x w \, \psi \, \mathrm{d}\vec{x}.$$

5.2. Discretization

We apply Rothe's method to the problem, i.e. we employ a time discretization followed by space discretizations of the sequence of resulting boundary value problems. We use a constant time step size $\tau = T/(K-1)$ with according discrete time instances $t_k = (k-1)\tau$ and semi-discretized solutions $\tilde{u}_k \approx u(t_k,\cdot)$ for $k = 1,\dots,K$. An implicit Euler time discretization is given as follows: For $u_0 \in L^2(\Omega)$, find $\tilde{u}_1(\mu),\dots,\tilde{u}_K(\mu) : S \to V$ such that

$$(\tilde{u}_k - u_0\mu, \phi) = 0 \quad \forall \phi \in V, \quad k = 1, \tag{15}$$

$$(\tilde{u}_k - \tilde{u}_{k-1}, \phi) + \tau a(\tilde{u}_k,\phi) + \tau b(\tilde{u}_k,\tilde{u}_k,\phi) = 0 \quad \forall \phi \in V, \quad k = 2,\dots,K. \tag{16}$$

This semi-discretization is the starting point for further adaptive space discretization and reduced-order modeling.

In order to obtain snapshots for a reduced-order model, we discretize the parameter domain into a training set $S_P := \{\mu_1,\dots,\mu_P\}$ with $\mu_1,\dots,\mu_P \in S$. We denote the respective semi-discrete solutions by $u_{k,p} \in V$ for time indices $k = 1,\dots,K$ and parameter indices $p = 1,\dots,P$, so that $u_{k,p} \approx u(t_k,\mu_p)$. We introduce a discretization in space with adaptive finite elements, so that each snapshot belongs to an individual finite element space chosen with respect to some global error tolerance ϵ. We denote the respective fully discrete solutions and finite element spaces by $u^\epsilon_{k,p} \in V^\epsilon_{k,p}$ for $k = 1,\dots,K$ and $p = 1,\dots,P$, so that $u^\epsilon_{k,p} \approx u_{k,p}$. A discretization of the problem with an implicit Euler scheme in time and adaptive finite elements in space for the training parameter values is given as follows: For $u_0 \in L^2(\Omega)$ and $p = 1,\dots,P$, find $u^\epsilon_{1,p} \in V^\epsilon_{1,p},\dots,u^\epsilon_{K,p} \in V^\epsilon_{K,p}$ such that

$$\begin{aligned} (u^\epsilon_{k,p} - u_0\mu_p, \phi) &= 0 \quad \forall \phi \in V^\epsilon_{k,p},\ k = 1, \\ (u^\epsilon_{k,p} - u^\epsilon_{k-1,p}, \phi) + \tau a(u^\epsilon_{k,p},\phi) + \tau b(u^\epsilon_{k,p},u^\epsilon_{k,p},\phi) &= 0 \quad \forall \phi \in V^\epsilon_{k,p},\ k = 2,\dots,K. \end{aligned}$$

The finite element discretization leads to a set of non-linear algebraic equations which can be solved in parallel for $p = 1,\dots,P$ and in sequence for $k = 1,\dots,K$. The non-linear equations are solved with a standard Newton method using a direct solver for the resulting sequence of linear algebraic equations in each time step.

5.3. Reduced-order modeling

We compute a POD of all available discrete solutions. To have a consistent notation, we fix ϵ and denote the snapshots as $u_1 \in V_1, \dots, u_N \in V_N$ with $N = KP$. We set $u_n = u^\epsilon_{k,p}$ and $V_N = V^\epsilon_{k,p}$ with $N = k + (p-1)P$ for $k = 1, \dots, K$ and $p = 1, \dots, P$. Using the methods from section 2, we create a POD space $V^R \subset V$ with basis functions $\phi_1, \dots, \phi_R$. Substituting V by V^R in the semi-discretization (15)–(16) gives rise to the following discretized reduced-order model: Find $u_1^R(\mu), \dots, u_K^R(\mu) : S \to V^R$ such that

$$(u_k^R - u_0\mu, \phi) = 0 \quad \forall \phi \in V^R, \quad k = 1, \tag{17}$$

$$(u_k^R - u_{k-1}^R, \phi) + \tau a(u_k^R, \phi) + \tau b(u_k^R, u_k^R, \phi) = 0 \quad \forall \phi \in V^R, \quad k = 2, \dots, K. \tag{18}$$

We can rewrite the model as a set of equations for the POD coefficient vectors: Find $\vec{b}_1(\mu), \dots, \vec{b}_K(\mu) : S \to \mathbb{R}^R$ such that

$$M^R \vec{b}_k - \vec{b}_0 \mu = \vec{0}, \quad k = 1,$$

$$M^R(\vec{b}_k - \vec{b}_{k-1}) + \tau A^R \vec{b}_k + \tau B^R(\vec{b}_k)\vec{b}_k = \vec{0}, \quad k = 2, \dots, K.$$

Expressions for the constant model coefficient matrices M^R, A^R, B^R and the initial data $\vec{b}_0$ follow directly from substitution of the POD expansion (3) into (17)–(18) and testing against the POD basis functions. An evaluation of the term involving B amounts to a multiplication with a tensor containing R^3 constant model coefficients.

5.4. Results

We consider an example with a domain width $X = 1.0$, a domain height $Y = 0.5$, a final time $T = 1.2$ and a parameter interval $S = [0, 1]$. As initial data we specify a two-dimensional sinusoidal profile

$$u_0 = 0.5 + 0.5 \sin\big((x - y - 0.75)\pi\big) \sin\big((x + y + 0.25)\pi\big).$$

We choose a viscosity constant of $\nu = 0.001$. The problem is a modified version of a numerical example in [6].

The time discretization uses a step size of 0.005. For the spatial refinement, we employ the error indicator of the Matlab PDE toolbox with a spatial tolerance of 0.02 for the error based on the viscous term. The adapted meshes and corresponding numerical solutions at various time instances are presented in Figure 8 for a fixed parameter choice of $\mu = 1$.

In order to generate snapshots for subsequent reduced-order modeling, we solve the problem for 33 parameter values distributed equidistantly over the parameter interval and store the spatial solutions at all time steps. We compute a POD of these snapshots and create Galerkin reduced-order models of varying dimension. In order to test the accuracy of the reduced-order model, we solve it for the parameter values in the training set.

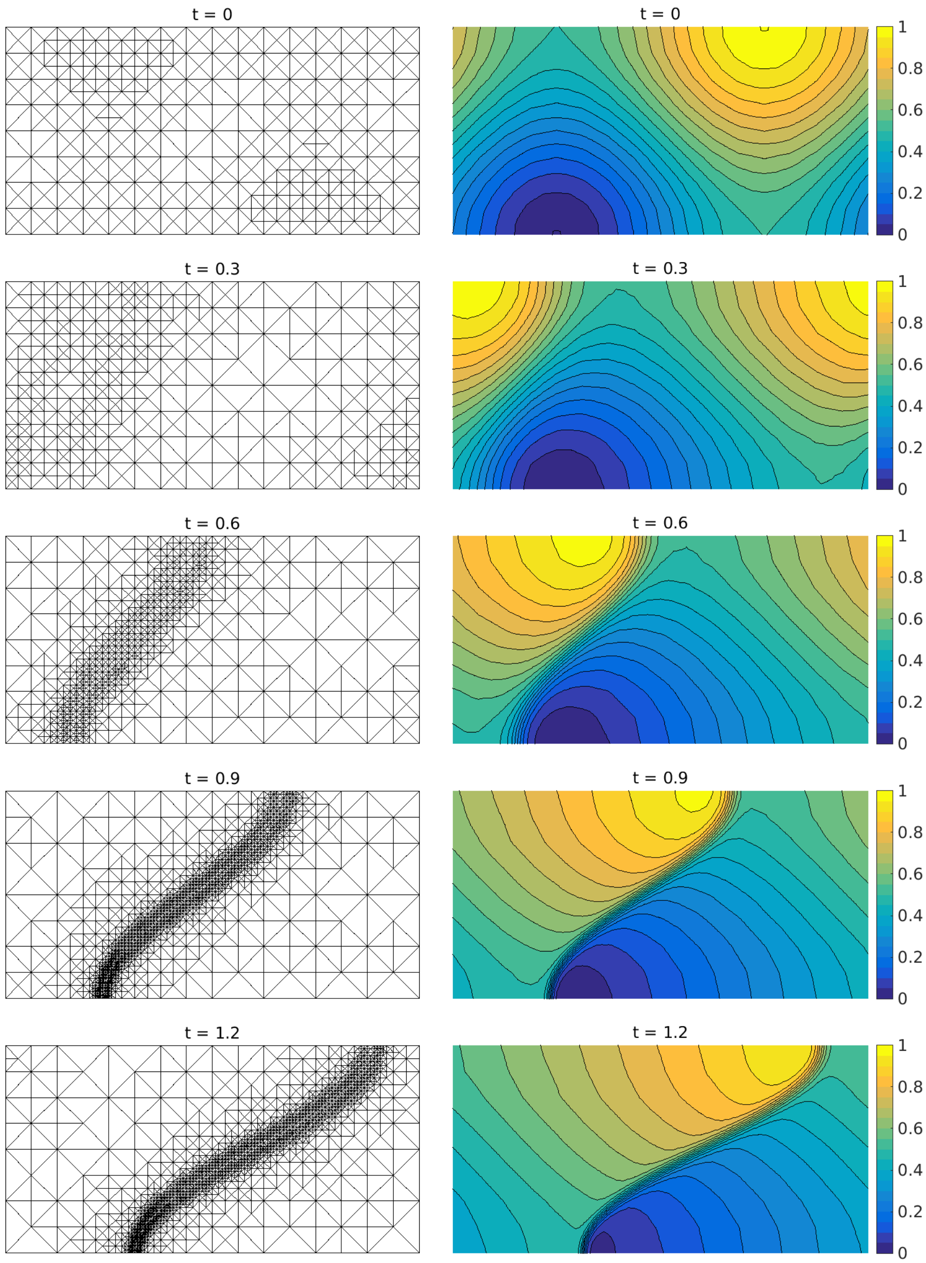

Figure 8: Solution of the Burgers problem with $\mu = 1$ at different times. Left: Adapted finite element meshes, right: numerical solutions

The errors are measured like in the elliptic case presented in section 4. The error between the snapshots and the solution of the POD-Galerkin model is defined by

$$\epsilon_{\text{ROM}} = \sqrt{\sum_{k=1}^{K}\sum_{p=1}^{P} \|u_{k,p}^{\epsilon} - u_k^R(\mu_p)\|_V^2} \Bigg/ \sqrt{\sum_{k=1}^{K}\sum_{p=1}^{P} \|u_{k,p}^{\epsilon}\|_V^2}. \tag{19}$$

The error between the snapshots and their orthogonal projection on the POD is defined by

$$\begin{aligned} \epsilon_{\text{POD}} &= \sqrt{\sum_{k=1}^{K}\sum_{p=1}^{P} \|u_{k,p}^{\epsilon} - P^R u_{k,p}^{\epsilon}\|_V^2} \Bigg/ \sqrt{\sum_{k=1}^{K}\sum_{p=1}^{P} \|u_{k,p}^{\epsilon}\|_V^2}, \\ &= \sqrt{\sum_{n=R+1}^{D} \lambda_n} \Bigg/ \sqrt{\sum_{n=1}^{D} \lambda_n}. \end{aligned} \tag{20}$$

To estimate the finite element discretization error of the snapshots, we compute reference snapshots $u_{k,p}^{\text{ref}}$ for $k = 1, \ldots, K$ and $p = 1, \ldots, P$ with an adaptive discretization using a stricter spatial tolerance of 0.005. The finite element discretization error is given by

$$\epsilon_{\text{FEM}} = \sqrt{\sum_{k=1}^{K}\sum_{p=1}^{P} \|u_{k,p}^{\epsilon} - u_{k,p}^{\text{ref}}\|_V^2} \Bigg/ \sqrt{\sum_{k=1}^{K}\sum_{p=1}^{P} \|u_{k,p}^{\epsilon}\|_V^2}. \tag{21}$$

The results are shown in Figure 9. We observe that the number of POD basis functions needed for a given relative error is increased in comparison with the convection-diffusion problem presented in the previous section. Otherwise the results are similar. In particular, the POD projection error with respect to the snapshots monotonically decreases, but the convergence of the solution of the reduced-order model stagnates at some point. Also the results for a fixed grid are qualitatively similar to the corresponding results of the convection-diffusion problem, so the same conclusions regarding the use of adaptive snapshots can be drawn in the case of a Burgers problem.

6. Conclusions

We have extended the framework of POD-Galerkin reduced-order modeling to snapshot data obtained with adaptive finite elements, where each snapshot may be represented in a different finite element space. These reduced-order models rely on a representation in a common finite element space of all snapshots. Because creating such a space may be computationally demanding, we have proposed a method to create the reduced-order model without actually building the common finite element space.

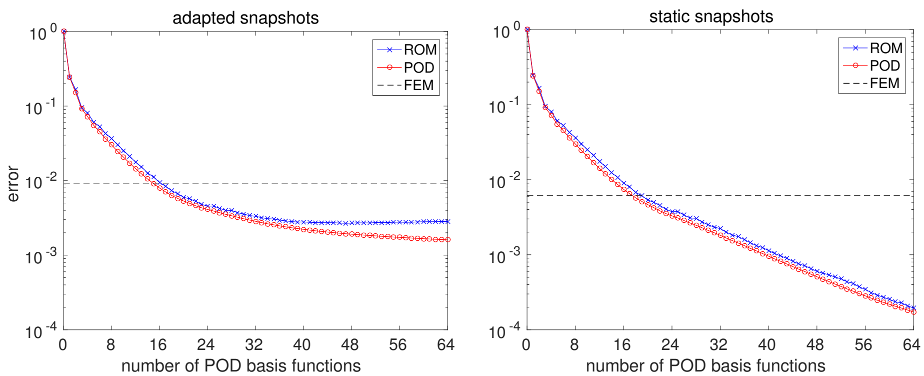

Figure 9: Error ϵ_{POD} of the POD projection, error ϵ_{ROM} of the solution of the reduced-order model, and finite element discretization error ϵ_{FEM}, corresponding to (19)–(21). Left: adapted snapshots, right: static snapshots computed on the common grid of the space-adapted snapshots.

The POD projection error of our method converges when the dimension is increased. The error between the POD Galerkin solution and the snapshots, however, contains a contribution from the spatial finite element discretization, which does not vanish when the POD dimension is increased. We have shown this effect for the case of linear elliptic boundary value problems.

Our findings are underlined with a numerical example of a convection-diffusion problem. Here we could observe that the error caused by the spatial adaptivity is dominated by the finite element error of the snapshots. Computational results for a Burgers problem suggest that the principal statements can be carried over to a broader class of problems, including non-linear parametrized parabolic PDEs.

References

[1] M. Ali, K. Steih, and K. Urban, *Reduced basis methods based upon adaptive snapshot computations*, submitted.

[2] M. Ali and K. Urban, *Reduced basis exact error estimates with wavelets*, to appear in Proceedings of ENUMATH 2015.

[3] A. Alla, C. Grässle, and M. Hinze, *A residual based snapshot location strategy for POD in distributed optimal control of linear parabolic equations*, tech. rep., Hamburger Beiträge zur Angewandten Mathematik 2015-42, 2015.

[4] M. Cascon, C. Kreuzer, R. Nochetto, and K. Siebert, *Quasi-optimal convergence rate for an adaptive finite element method*, SIAM J. Numer. Anal., 46 (2008), pp. 2524–2550.

[5] P. Chen and A. Quarteroni, *A new algorithm for high-dimensional uncertainty quantification based on dimension-adaptive sparse grid approximation and reduced basis methods*, J. Comp. Phys., 298 (201), pp. 176 – 193.

[6] M. Drohmann, B. Haasdonk, and M. Ohlberger, *Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation*, SIAM J. Sci. Comp., 34 (2012), pp. A937–A969.

[7] J. Ghiglieri and S. Ulbrich, *Optimal flow control based on POD and MPC and an application to the cancellation of Tollmien-Schlichting waves*, Optim. Meth. Software, 29 (2014), pp. 1042–1074.

[8] B. Haasdonk, K. Urban, and B. Wieland, *Reduced basis methods for parametrized partial differential equations with stochastic influences using the Karhunen Loeve expansion*, SIAM/ASA J. Uncertainty Quantification, 1 (2013), pp. 79–105.

[9] J. Hesthaven, G. Rozza, and B. Stamm, *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, Springer, 2016.

[10] P. Holmes, J. Lumley, and G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge University Press, 1996.

[11] K. Kunisch and S. Volkwein, *Control of the Burgers equation by a reduced-order approach using proper orthogonal decomposition*, J. Optimiz. Theory App., 102 (1999), pp. 345–371.

[12] ———, *Galerkin proper orthogonal decomposition methods for parabolic problems*, Numer. Math., 90 (2001), pp. 117–148.

[13] O. Lass, *Reduced order modeling and parameter identification for coupled nonlinear PDE systems*, PhD thesis, Universiät Konstanz, 2014.

[14] F. Negri, G. Rozza, A. Manzoni, and A. Quarteroni, *Reduced basis method for parametrized elliptic optimal control problems*, SIAM J. Sci. Comp., 35 (2013), pp. A2316–A2340.

[15] C. Prud'homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici, *Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods*, J. Fluid. Eng.–T. ASME, 124 (2002), pp. 70–80.

[16] A. Quarteroni, A. Manzoni, and F. Negri, *Reduced Basis Methods for Partial Differential Equations*, Springer, 2016.

[17] L. Sirovich, *Turbulence and the dynamics of coherent structures. parts I, II and III*, Quart. Appl. Math., 45 (1987), pp. 561–590.

[18] R. Stevenson, *Optimality of a standard adaptive finite element method*, Found. Comput. Math., (2007), pp. 245–269.

[19] F. Tröltzsch and S. Volkwein, *POD a-posteriori error estimates for linear-quadratic optimal control problems*, Comput. Optim. Appl., 44 (2009), pp. 83–115.

[20] S. Ullmann and J. Lang, *POD-Galerkin modeling and sparse-grid collocation for a natural convection problem with stochastic boundary conditions*, in Sparse Grids and Applications – Munich 2012, J. Garke and D. Pflüger, eds., Lecture Notes in Computational Science and Engineering, Springer, 2014.

[21] M. Yano, *A minimum-residual mixed reduced basis method: Exact residual certification and simultaneous finite-element reduced-basis refinement*, ESAIM: M2AN, 50 (2016), pp. 163–185.