



SUPG reduced order models for convection-dominated convection–diffusion–reaction equations

Swetlana Giere^{a,*}, Traian Iliescu^b, Volker John^{a,c}, David Wells^b

^a Weierstrass Institute for Applied Analysis and Stochastics, Leibniz Institute in Forschungsverbund Berlin e. V. (WIAS), Mohrenstr. 39, 10117 Berlin, Germany

^b Department of Mathematics, Virginia Tech, 456 McBryde Hall, Blacksburg, VA 24061-0123, USA

^c Free University of Berlin, Department of Mathematics and Computer Science, Arnimallee 6, 14195 Berlin, Germany

Received 29 August 2014; received in revised form 20 November 2014; accepted 6 January 2015

Available online 18 February 2015

Abstract

This paper presents a Streamline-Upwind Petrov–Galerkin (SUPG) reduced order model (ROM) based on proper orthogonal decomposition (POD). This ROM is investigated theoretically and numerically for convection-dominated convection–diffusion–reaction problems. The SUPG finite element method was used on realistic meshes for computing the snapshots, leading to some noise in the POD data. Numerical analysis is used to propose the scaling of the stabilization parameter for the SUPG-ROM. Two approaches are used: One based on the underlying finite element discretization and the other one based on the POD truncation. The resulting SUPG-ROMs and the standard Galerkin ROM (G-ROM) are studied numerically. For many settings, the results obtained with the SUPG-ROMs are more accurate. Finally, one of the choices for the stabilization parameter is recommended.

© 2015 Elsevier B.V. All rights reserved.

Keywords: Reduced order models (ROMs); Convection-dominated equations; Streamline-Upwind Petrov–Galerkin (SUPG) method; Proper Orthogonal Decomposition (POD); Stabilization parameter

1. Introduction

The numerical simulation of many complex problems requires repeatedly solving subproblems. For instance, in the iterative solution of optimal control problems with partial differential equations, one has to solve the same type of partial differential equation, with (slightly) changing data, over and over again. With standard discretizations for partial differential equations, like finite element methods, the solution of the subproblems is often the most time-consuming part of the simulations. Reduced order modeling aims at finding low-dimensional spaces that allow solution of partial differential equations orders of magnitude more efficiently than a finite element method, with only an acceptable loss of accuracy. This paper considers reduced order models (ROMs) for scalar convection-dominated convection–diffusion–reaction equations and studies suitable choices for the parameter of a stabilized discretization.

* Corresponding author.

E-mail addresses: swetlana.schyschlowa@wias-berlin.de (S. Giere), iliescu@vt.edu (T. Iliescu), volker.john@wias-berlin.de (V. John), drwells@vt.edu (D. Wells).

Solutions of convection-dominated problems usually possess layers which cannot be resolved with the underlying mesh, particularly in higher dimensions. The Galerkin finite element method is called unstable because it cannot cope with this situation and its usage leads to numerical solutions which are globally polluted with spurious oscillations. One has to use a so-called stabilized discretization. A variety of such discretizations have been proposed over the last few decades, e.g., see [1–3] for reviews and numerical comparisons. However, the question of finding a perfect discretization, i.e., a discretization which gives solutions with steep layers and without spurious oscillations, is still open. One of the most popular stabilized finite element methods is the Streamline-Upwind Petrov–Galerkin (SUPG) method proposed in [4,5]. Solutions computed with this method usually possess steep layers but also exhibit some spurious oscillations in a vicinity of the layers. The SUPG method contains a stabilization parameter whose asymptotic value for steady-state problems is well known from finite element error analysis (e.g., [6]); it depends on the local mesh width. The situation is not completely clear for time-dependent problems. For general problems, optimal estimates can only (to the best of the authors' knowledge) be derived for parameters dependent on the length of the time step. For a simplified situation, estimates can also be proven for parameters dependent on the mesh width; see [7] for details. From the practical point of view, the latter choice seems to be more appropriate since the difficulty of not being able to resolve the layers vanishes on sufficiently fine meshes but not for sufficiently small time steps.

ROMs are already used for many complex systems. One of the most popular ROM approaches is Proper Orthogonal Decomposition (POD). POD extracts the most pertinent features of a data set and naturally leads to a Galerkin formulation of a ROM, which will be denoted as the Galerkin ROM (G-ROM). This paper exclusively considers ROMs based on POD. There are situations where G-ROMs are efficient and relatively accurate (see [8–10]). However, in other situations, a G-ROM might produce inaccurate results [11]. One of the main reasons for these inaccurate results is that the underlying G-ROM can be numerically unstable, e.g., the G-ROM solution can blow up in a nonphysical way, see [12–14] for the compressible Navier–Stokes equations and [15,16] for the incompressible Navier–Stokes equations. Various stabilized ROMs have been proposed, see [12,17,18,15,19–21,16,22]; see also [23–25] for similar work in reduced basis methods. This paper focuses on ROMs for convection-dominated convection–diffusion–reaction equations and these ROMs' potential numerical instability due to the unresolved layers.

After computing the POD modes, the question is how to design a numerical method that gives a solution that is as accurate as possible in the space spanned by the first r POD modes. In the numerical studies presented below, the solution of the continuous problem is known, so accuracy always refers to this solution. When the solution of the continuous problem is not available, one could compare the ROM results with the solution of the simulation used to compute the snapshots, even if this solution is usually not the best approximation of the continuous solution in the finite element space. Since the numerical method providing the most accurate solution in the r -dimensional space is not necessarily the method which was used for computing the snapshots, other methods can be investigated as well.

The main class of methods which will be studied in this paper is SUPG stabilized ROMs (SUPG-ROMs), see, e.g., [26] for alternative approaches. To the best of the authors' knowledge the SUPG-ROM was first used in [21] and later on in [19], in both papers for the Navier–Stokes equations. In addition to the SUPG-ROMs, the G-ROM will also be investigated.

As in the finite element SUPG method, the question of appropriate stabilization parameters for SUPG-ROMs arises. SUPG-ROM parameters depending only on the spatial resolution are preferable for the same reasons as in the finite element method. A ROM based on finite element data has two parts to its spatial resolution: The spatial resolution from the finite element space and the spatial resolution from the space of POD modes used in the ROM, which is a subspace of the finite element space. One can ask upon which spatial resolution the stabilization parameter for the SUPG-ROM should depend. This is the main question studied in this paper.

The question of appropriate stabilization parameters for the SUPG-ROM is addressed by means of a numerical analysis of this problem. To the best of the authors' knowledge, the use of numerical analysis to propose the SUPG-ROM stabilization parameter is new. In the literature so far, simply the stabilization parameter from the finite element method was used, like in [21], or an optimization problem for the determination of the parameter was solved, as in [19]. Motivations for these approaches with numerical analysis were not provided. In the authors' opinion, it is important to have some support for the choice of stabilization parameters coming from numerical analysis, since parameters determined with considerations from numerical analysis should be valid for a wide range of settings (e.g., diffusion coefficients and the convection vector). Two stabilization parameters will be proposed as a result of analytical considerations. One of them is based on the finite element resolution and the other one is based on the POD spatial resolution. The resulting ROMs will be denoted as FE-SUPG-ROM and POD-SUPG-ROM, respectively.

This paper is organized as follows. Section 2 gives the continuous and the finite element formulations of the time-dependent convection–diffusion–reaction equation. Moreover, a short review of POD is presented and the G-ROM and the SUPG-ROM are defined. The core of Section 3 is the proposal of two stabilization parameters for the SUPG-ROM, using a numerical analysis of the SUPG-ROM. The two versions of the SUPG-ROM, together with the G-ROM, are studied numerically in Section 4 on two convection-dominated convection–diffusion–reaction problems that display sharp internal layers. The paper concludes with a summary and outlook in Section 5.

2. The continuous, the finite element, and the ROM problem

Throughout this paper, standard notations for Lebesgue and Sobolev spaces and their corresponding norms are used. With C , a generic constant is denoted which does not depend on the mesh width nor the diffusion coefficient.

A time-dependent linear convection–diffusion–reaction equation is given by

$$\begin{aligned} \partial_t u - \varepsilon \Delta u + \mathbf{b} \cdot \nabla u + cu &= f \quad \text{in } (0, T] \times \Omega, \\ u &= 0 \quad \text{on } [0, T] \times \partial\Omega, \\ u(0, \mathbf{x}) &= u_0(\mathbf{x}) \quad \text{in } \Omega, \end{aligned} \quad (1)$$

where Ω is a bounded domain in \mathbb{R}^d , $d \in \{1, 2, 3\}$, with the boundary $\partial\Omega$, $\mathbf{b}(t, \mathbf{x})$ and $c(t, \mathbf{x})$ denote convection and reaction fields, respectively, $\varepsilon > 0$ is a constant diffusion coefficient, $u_0(\mathbf{x})$ is a given initial condition, and T is the length of the considered time interval.

Let $X = H_0^1(\Omega)$. Eq. (1) in the weak form reads as follows: Find $u : [0, T] \rightarrow X$ such that

$$(\partial_t u, v) + (\varepsilon \nabla u, \nabla v) + (\mathbf{b} \cdot \nabla u, v) + (cu, v) = (f, v) \quad \forall v \in X, \quad (2)$$

where (\cdot, \cdot) denotes the inner product in $L^2(\Omega)^d$. To guarantee the coercivity of (2), it is common to assume that there is a constant $c_0 > 0$ such that

$$0 < c_0 \leq \left(c - \frac{1}{2} \nabla \cdot \mathbf{b} \right) (t, \mathbf{x}) \quad \forall (t, \mathbf{x}) \in [0, T] \times \Omega. \quad (3)$$

Moreover, it will be assumed that \mathbf{b} , c and $\nabla \cdot \mathbf{b}$ belong to $L^2(0, T; L^\infty)$.

Let $X_h \subset X$ denote a conforming finite element space spanned by piecewise polynomials of order m . The paper considers only the case of uniform families of triangulations $\{\mathcal{T}_h\}$ of Ω , where h is the mesh size for all mesh cells $K \in \mathcal{T}_h$. By replacing the space X in (2) by X_h , one obtains the Galerkin finite element method which has to be equipped with an appropriate finite element approximation $u_h(0, \mathbf{x})$ of $u_0(\mathbf{x})$.

It is well-known that, in the case of small diffusion ε compared with the convection field \mathbf{b} , the Galerkin method is unstable and leads generally to solutions which are globally polluted with strong spurious oscillations. One of the most popular stabilized finite element methods is the SUPG method, which will be considered in this paper.

Let Δt denote a fixed time step and let u_h^n be the finite element solution at $t^n = n \Delta t$. The fully discretized backward Euler/SUPG method reads as follows: For $n = 1, 2, \dots$ find $u_h^n \in X_h$ such that $\forall v_h \in X_h$

$$\begin{aligned} (u_h^n - u_h^{n-1}, v_h) + \Delta t a_{\text{SUPG},h}(u_h^n, v_h) &= \Delta t (f^n, v_h) + \Delta t \delta_h (f^n, \mathbf{b}^n \cdot \nabla v_h) \\ &\quad - \delta_h (u_h^n - u_h^{n-1}, \mathbf{b}^n \cdot \nabla v_h), \end{aligned} \quad (4)$$

where $u_h^0(\mathbf{x}) = u_h(0, \mathbf{x})$, and δ_h is the SUPG stabilization parameter that has to be chosen. Note that δ_h is in general a local parameter defined on each mesh cell $K \in \mathcal{T}_h$. However, for uniform triangulations it is possible to use the same parameter for all mesh cells. The bilinear form $a_{\text{SUPG},h}(\cdot, \cdot)$ is defined by

$$\begin{aligned} a_{\text{SUPG},h}(u_h, v_h) &= (\varepsilon \nabla u_h, \nabla v_h) + (\mathbf{b} \cdot \nabla u_h, v_h) + (cu_h, v_h) \\ &\quad + \delta_h \sum_{K \in \mathcal{T}_h} (-\varepsilon \Delta u_h + \mathbf{b} \cdot \nabla u_h + cu_h, \mathbf{b} \cdot \nabla v_h)_K \end{aligned} \quad (5)$$

for all $u_h, v_h \in X_h$, where $(\cdot, \cdot)_K$ denotes the inner product in $L^2(K)^d$, $K \in \mathcal{T}_h$.

Finite element error analysis for the method (4) can be found in [7]. For the general case of time-dependent coefficients of the problem, an optimal error estimate was proved for $\delta_h = \mathcal{O}(\Delta t)$. However, this choice of the stabilization parameter does not reflect the fact that the reason for needing a stabilized discretization is the appearance of layers in the solution, which are spatial features. Thus, stabilization parameters depending on the mesh width seem to be more appropriate. For the situation of steady-state convection and reaction, an optimal error estimate for $\delta_h = \mathcal{O}(h)$ was derived in [7]. Numerical studies in [7] reveal that the choice of $\delta_h = \mathcal{O}(h)$ is more appropriate also in the general case. For this reason, all simulations with the finite element SUPG method presented in this paper were performed with $\delta_h = h$.

It is assumed that the space X_h satisfies the following local approximation property: for each $u \in X \cap H^{m+1}(\Omega)$ there exists a function $\hat{u}_h \in X_h$ such that

$$\|u - \hat{u}_h\|_{0,K} + h\|\nabla(u - \hat{u}_h)\|_{0,K} + h^2\|\Delta(u - \hat{u}_h)\|_{0,K} \leq Ch^{m+1}\|u\|_{m+1,K} \tag{6}$$

for all $K \in \mathcal{T}_h$. This property is given, for example, for Lagrange finite elements on mesh cells which allow an affine mapping to a reference mesh cell.

One of the most popular techniques for the reduced order modeling of time-dependent problems is POD. For a detailed description of the method, the reader is referred to [27,13,28–31]. The main idea of POD is to find a low-dimensional basis that approximates the snapshots $\{u_h^1, \dots, u_h^N\}$, which here are finite element solutions of (4) at N different times, in the best possible way with respect to a certain norm. In this paper, the $L^2(\Omega)$ norm will be chosen, which is the most common choice found in the literature. The best approximation problem can be expressed by a minimization problem whose solution is equivalent to the solution of the eigenvalue problem

$$YY^T M \underline{\varphi}_j = \lambda_j \underline{\varphi}_j, \quad j = 1, \dots, N, \tag{7}$$

where $\underline{\varphi}_j$ and λ_j denote the vector of the finite element coefficients of the POD basis functions and the POD eigenvalues, respectively, Y denotes the snapshot matrix, whose columns correspond to the finite element coefficients of the snapshots, and M denotes the finite element mass matrix representing the $L^2(\Omega)$ inner product. It turns out that the eigenvalues are real and non-negative such that they can be ordered in the form

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_R > \lambda_{R+1} = \dots = \lambda_N = 0.$$

Then, the basis for the ROM is chosen to consist of the functions $\varphi_i, i = 1, \dots, r$, which correspond to the first $r \leq R$ largest eigenvalues. These functions are normalized such that $\|\varphi_i\|_0 = 1, i = 1, \dots, r$ and r is called the rank of the basis.

Let X_R denote the R -dimensional space of the snapshots with $R \leq N$, and X_r the r -dimensional POD space spanned by the POD basis functions $\varphi_1, \dots, \varphi_r$. As the POD basis functions are, by construction, linear combinations of finite element basis functions, it holds that

$$X_r \subseteq X_R \subset X_h \subset X. \tag{8}$$

Let M_r, S_r , and H_r be the POD mass, stiffness, and Hessian matrices, respectively, with

$$\begin{aligned} (M_r)_{ij} &= \int_{\Omega} \varphi_j \varphi_i \, d\mathbf{x}, \\ (S_r)_{ij} &= \int_{\Omega} \nabla \varphi_j \cdot \nabla \varphi_i \, d\mathbf{x}, \\ (H_r)_{ij} &= \sum_{K \in \mathcal{T}_h} \int_K \Delta \varphi_j \Delta \varphi_i \, d\mathbf{x}. \end{aligned} \tag{9}$$

In the numerical studies, second order finite elements will be used such that H_r does not become the zero matrix.

In this paper, the centered-trajectory method for computing the POD basis functions is considered, i.e., the POD basis functions are computed from the fluctuation of the snapshots $u_h^n - \bar{u}_h, n = 1, \dots, N$, where \bar{u}_h is the arithmetic average value of the snapshots. The centered-trajectory method is the common approach in practice. Among other advantages, the use of the centered-trajectory method facilitates the incorporation of steady-state Dirichlet boundary

conditions into the ROM, since these boundary conditions are taken by \bar{u}_h and the fluctuations obey homogeneous Dirichlet boundary conditions.

Let $u_r(t, \mathbf{x}) + \bar{u}_h(\mathbf{x})$ denote the ROM approximation of the finite element solution and let $u_r^n + \bar{u}_h$ denote the ROM solution evaluated at the time instance t^n . The backward Euler/SUPG reduced order model reads as follows: For $n = 1, 2, \dots$ find $u_r^n \in X_r$ such that $\forall v_r \in X_r$

$$\begin{aligned} & \left(u_r^n - u_r^{n-1}, v_r \right) + \Delta t a_{\text{SUPG},r} (u_r^n, v_r) = \Delta t (f^n, v_r) \\ & + \Delta t \delta_r (f^n, \mathbf{b}^n \cdot \nabla v_r) - \delta_r (u_r^n - u_r^{n-1}, \mathbf{b}^n \cdot \nabla v_r) - \Delta t a_{\text{SUPG},r} (\bar{u}_h, v_r), \end{aligned} \quad (10)$$

where $a_{\text{SUPG},r}(\cdot, \cdot)$ has the same form as the bilinear form (5) with the SUPG-ROM parameter δ_r instead of δ_h . Note that by setting $\delta_r = 0$ in (10), the G-ROM is recovered. The study of appropriate choices of the SUPG-ROM parameter δ_r , using numerical analysis of the method and numerical simulations, is the main goal of this paper. To the best of the authors' knowledge, such investigations do not exist in the literature yet.

3. Proposals for the stabilization parameter δ_r based on numerical analysis

The proposals for choosing the stabilization parameter δ_r of the SUPG-ROM (10) are based on considerations of the error between the solution u of the continuous problem (2) and the solution $\bar{u}_h + u_r$ of (10). As the first step, the error is split in the form

$$u - (\bar{u}_h + u_r) = (u - u_h) + (u_h - P_r(u_h)) + (P_r(u_h) - (\bar{u}_h + u_r)), \quad (11)$$

where

$$P_r(u_h) := \bar{u}_h + \sum_{j=1}^r (u_h - \bar{u}_h, \varphi_j)_X \varphi_j \quad (12)$$

is the X projection of a finite element function into the space X_r .

This section starts with a discussion of inverse estimates in the context of ROMs. A new estimate for functions from X_r is presented. Then, error estimates for the finite element problem are discussed and estimates of the projection error for the projection from X_h to X_r are derived. Next, conditions for the coercivity of the SUPG bilinear form in X_r are given. The main part of this section consists in the derivation of appropriate stabilization parameters for the SUPG-ROM problem (10). Considering the simplified situation of a steady-state problem, two different types of parameters will be proposed, which are based on different types of inverse estimates discussed at the beginning of this section. Numerical examples illustrate several estimates and assumptions made in the derivation of the SUPG-ROM parameters.

3.1. Inverse estimates

Assuming that the family of triangulations is quasi-uniform, the following local inverse inequality for finite element functions holds, e.g., see [32, Thm. 3.2.6],

$$\|v_h\|_{m,K} \leq \mu_{\text{inv}} h_K^{l-m} \|v_h\|_{l,K} \quad \forall v_h \in X_h, \quad (13)$$

for $0 \leq l \leq m$, where h_K is the diameter of the mesh cell $K \in \mathcal{T}_h$. Values of μ_{inv} for different situations can be found in [33]. They are of order one. In the following, the constant in (13) will be included into a generic constant. For families of uniform triangulations, as considered in this paper, one can derive from (13) global inverse estimates of the form

$$\left(\sum_{K \in \mathcal{T}_h} \|v_h\|_{m,K}^2 \right)^{1/2} \leq C h^{l-m} \left(\sum_{K \in \mathcal{T}_h} \|v_h\|_{l,K}^2 \right)^{1/2} \quad \forall v_h \in X_h. \quad (14)$$

Inverse estimates are also known in the context of POD. In [34], the following inverse estimate was proven:

$$\|\nabla v_r\|_0 \leq \sqrt{\|S_r\|_2 \|M_r^{-1}\|_2} \|v_r\|_0, \quad \forall v_r \in X_r, \quad r \in \{1, \dots, R\}, \quad (15)$$

where $\|\cdot\|_2$ denotes the spectral norm of a matrix. When the $L^2(\Omega)$ norm is used to generate the snapshots (as in this paper), M_r in (15) is the identity matrix. Estimate (15) was derived in [34] for the situation that the POD basis is computed from snapshots in an infinite-dimensional Hilbert space.

In the setting of [34], the POD basis functions are known to belong to the infinite-dimensional Hilbert space. In practice, however, the POD basis is usually computed from snapshots of some numerical approximation of the solution of the continuous problem. Here, the situation where snapshots are computed with a finite element method and belong to X^h is considered. Consequently, the POD basis functions $\{\varphi_1, \dots, \varphi_r\}$, $r \leq R$, belong not only to X but also to X_h and X_r as in (8). Hence, two inverse estimates hold for functions in X_r : a POD estimate of form (15) and a finite element estimate of form (14).

The POD inverse estimate (15) can be extended to the piecewise defined Laplacian.

Lemma 3.1 (POD Inverse Estimate). *For all $v_r \in X_r$, $1 \leq r \leq R$, the following estimate holds:*

$$\|\Delta v_r\|_0 \leq \sqrt{\|H_r\|_2 \|S_r^{-1}\|_2} \|\nabla v_r\|_0, \tag{16}$$

where $\|\Delta v_r\|_0$ is defined by a sum over the mesh cells as used in (9).

Proof. The proof follows [34]. Let $v_r = \sum_{j=1}^r x_j \varphi_j$ and $\mathbf{x} = (x_1, \dots, x_r)^T$. Then, one obtains from the definition of H_r , a standard estimate of matrix–vector products, the fact that S_r is symmetric and positive definite, and the definition of S_r

$$\begin{aligned} \|\Delta v_r\|_0^2 &= \mathbf{x}^T H_r \mathbf{x} \leq \|H_r\|_2 \|\mathbf{x}\|_2^2 = \|H_r\|_2 \mathbf{x}^T \mathbf{x} \\ &= \|H_r\|_2 \mathbf{x}^T S_r^{T/2} S_r^{-1} S_r^{1/2} \mathbf{x} \leq \|H_r\|_2 \|S_r^{-1}\|_2 \|S_r^{1/2} \mathbf{x}\|_2^2 \\ &= \|H_r\|_2 \|S_r^{-1}\|_2 \mathbf{x}^T S_r \mathbf{x} = \|H_r\|_2 \|S_r^{-1}\|_2 \|\nabla v_r\|_0^2. \quad \square \end{aligned}$$

If in the snapshot generation the $H^1(\Omega)$ seminorm was used (see, e.g., [35,34]) instead of the $L^2(\Omega)$ norm utilized in this paper, then S_r in (16) is the identity matrix.

Remark 3.1. The asymptotic behavior of the first factor on the right-hand side of (16) will be discussed in a simplified situation. As in [35], it is assumed that the POD vectors are the Fourier basis in a single dimension with homogeneous Dirichlet boundary conditions, i.e., $\varphi_j(x) = \sin(\pi j x)$. The single dimension case is relevant because the considered convection-dominated problems exhibit motion along a preferred direction. Therefore, H_r is diagonal and has entries

$$\int_0^1 (\pi^2 j^2)^2 \sin(\pi j x)^2 dx = \mathcal{O}(j^4), \quad j = 1, \dots, r.$$

Similarly, S_r^{-1} is a diagonal matrix and it has entries

$$\left(\int_0^1 (\pi j)^2 \cos(\pi j x)^2 dx \right)^{-1} = \mathcal{O}(j^{-2}), \quad j = 1, \dots, r.$$

Hence $\|H_r\|_2 = \mathcal{O}(r^4)$ and $\|S_r^{-1}\|_2 = \mathcal{O}(1)$. Altogether, the first factor on the right-hand side of (16) scales like $\mathcal{O}(r^2)$.

The following numerical example will demonstrate that this scaling can be observed also in more general situations.

Remark 3.2. Consider a two-dimensional test example for the convection–diffusion–reaction equation (1) describing a traveling wave. The specification of the problem setting is given in Example 4.2. In Fig. 1, the dependence of the constants from the inverse estimates (15) and (16) on the dimension of the POD basis r is shown. The asymptotic behavior discussed in Remark 3.1 can be observed also in this two-dimensional case.

3.2. The finite element error

Standard finite element error analysis for scalar convection–diffusion–reaction equations is performed in the energy norm and for terms in appropriate norms which come from the stabilization of the finite element discretization,

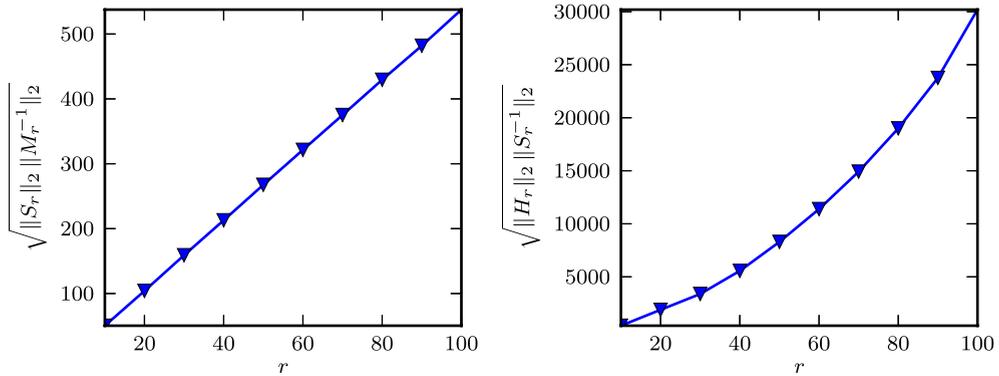


Fig. 1. Dependence of the constants from the inverse estimates (15) (left) and (16) (right) on the dimension of the POD basis r .

e.g., see [7, Thm. 5.3, Thm. 5.4] for results of this form. The errors are bounded by constants times the sum of the mesh width and the time step, both to some powers. The constants depend on norms of the solution of the continuous problem and the data of the problem, but not on ε and h . Thus, one typically obtains finite element error estimates of the form

$$(\Delta t)^{1/2} \left(\sum_{n=1}^N \|u^n - u_h^n\|_0 + \varepsilon^{1/2} \sum_{n=1}^N \|\nabla(u^n - u_h^n)\|_0 + \delta_h^{1/2} \sum_{n=1}^N \|\mathbf{b} \cdot \nabla(u^n - u_h^n)\|_0 \right) \leq C h^{m+1/2} + (\Delta t)^k, \quad (17)$$

where k is the order of the temporal discretization.

Using the local approximation property (6), the inverse estimate (14), and estimate (17), one also obtains an estimate for the Laplacian of the error

$$\begin{aligned} & (\Delta t)^{1/2} \sum_{n=1}^N \sum_{K \in \mathcal{T}_h} \|\Delta(u^n - u_h^n)\|_{0,K} \\ & \leq (\Delta t)^{1/2} \sum_{n=1}^N \left(\sum_{K \in \mathcal{T}_h} \|\Delta(u^n - \hat{u}_h^n)\|_{0,K} + \|\Delta(\hat{u}_h^n - u_h^n)\|_{0,K} \right) \\ & \leq C (\Delta t)^{1/2} \sum_{n=1}^N \left(\sum_{K \in \mathcal{T}_h} h^{m-1} \|u^n\|_{m+1,K} + h^{-1} \|\nabla(\hat{u}_h^n - u_h^n)\|_{0,K} \right) \\ & \leq C (\Delta t)^{1/2} \sum_{n=1}^N \left(h^{m-1} + h^{-1} \|\nabla(u^n - \hat{u}_h^n)\|_0 + h^{-1} \|\nabla(u^n - u_h^n)\|_0 \right) \\ & \leq C (\Delta t)^{1/2} \left(h^{m-1} + \varepsilon^{-1/2} h^{m-1/2} + \varepsilon^{-1/2} h^{-1} (\Delta t)^k \right). \end{aligned} \quad (18)$$

3.3. The projection error

This subsection presents an estimate for the error between the snapshots and their projection into the POD space X_r . This error is the second term on the right-hand side of the error decomposition (11).

Lemma 3.2. Let $\bar{u}_h = 0$ and let $(\cdot, \cdot)_s$, $s \in \{0, 1, 2\}$, be a semi-inner product [36] with induced seminorm $|\cdot|_s$ of $H^s(\Omega)$. Then the POD projection error in the s -seminorm satisfies

$$\sum_{n=1}^N \left| u_h^n - \sum_{j=1}^r (u_h^n, \varphi_j)_X \varphi_j \right|_s^2 = \sum_{j=r+1}^R \lambda_j |\varphi_j|_s^2.$$

For $s = 2$, $(\cdot, \cdot)_s$ and $|\cdot|_s$ have to be understood as a sum over the mesh cells.

Proof. Taking the s -seminorm of the POD truncation error, using the definition of the eigenvalues and eigenfunctions of POD, e.g., see [27, Eq. (6)] or (7) for the matrix–vector representation, and applying the orthogonality of the POD basis functions yields directly

$$\begin{aligned} \sum_{n=1}^N \left| u_h^n - \sum_{j=1}^r (u_h^n, \varphi_j)_X \varphi_j \right|_s^2 &= \sum_{n=1}^N \left| \sum_{j=r+1}^R (u_h^n, \varphi_j)_X \varphi_j \right|_s^2 \\ &= \sum_{n=1}^N \left(\sum_{j=r+1}^R (u_h^n, \varphi_j)_X \varphi_j, \sum_{k=r+1}^R (u_h^n, \varphi_k)_X \varphi_k \right)_s \\ &= \sum_{n=1}^N \sum_{j=r+1}^R \sum_{k=r+1}^R (u_h^n, \varphi_j)_X (u_h^n, \varphi_k)_X (\varphi_j, \varphi_k)_s \\ &= \sum_{j=r+1}^R \sum_{k=r+1}^R \left(\sum_{n=1}^N (u_h^n, \varphi_j)_X (u_h^n, \varphi_k)_X \right) (\varphi_j, \varphi_k)_s \\ &= \sum_{j=r+1}^R \sum_{k=r+1}^R (\lambda_j \varphi_j, \varphi_k)_X (\varphi_j, \varphi_k)_s \\ &= \sum_{j=r+1}^R \lambda_j |\varphi_j|_s^2. \quad \square \end{aligned}$$

The result of Lemma 3.2 is similar to results obtained in [37,38].

Corollary 3.1. *It holds that*

$$\sum_{n=1}^N |u_h^n - P_r(u_h^n)|_s^2 = \sum_{j=r+1}^R \lambda_j |\varphi_j|_s^2. \tag{19}$$

Proof. By the definition (12) of the projection, it follows that

$$\sum_{n=1}^N |u_h^n - P_r(u_h^n)|_s^2 = \sum_{n=1}^N \left| (u_h^n - \bar{u}_h) - \sum_{j=1}^r (u_h^n - \bar{u}_h, \varphi_j)_X \varphi_j \right|_s^2.$$

Since $\overline{u_h^n - P_r(u_h^n)}$ is zero, the application of Lemma 3.2 proves the statement. \square

Corollary 3.2. *It holds that*

$$\sum_{n=1}^N |u_h^n - P_r(u_h^n)|_s^2 \leq Ch^{-2s} \sum_{j=r+1}^R \lambda_j. \tag{20}$$

Proof. Since $X_r \subset X_h$, one can apply the inverse estimate (14) to the right-hand side of (19)

$$|\varphi_j|_s^2 \leq Ch^{-2s} \|\varphi_j\|_0^2.$$

The statement of the corollary follows by utilizing the fact that the POD basis functions are normalized. \square

Altogether, there are two ways to bound the projection error: With data from the POD only, as in (19), or with data from the POD and the finite element method, as in (20). Note that the sums on the right-hand side of (19) and (20) can be computed.

3.4. The coercivity of the SUPG-ROM bilinear form in X_r

The coercivity of the SUPG bilinear form (5) in X_r is essential for the well-posedness of the SUPG-ROM problem in X_r . This property gives some restrictions on the stabilization parameters.

Lemma 3.3. *Let either*

$$0 \leq \delta_r \leq \frac{1}{2} \min \left\{ \frac{c_0}{\|c\|_\infty^2}, \frac{h^2}{\varepsilon \mu_{\text{inv}}^2} \right\}, \quad (21)$$

or

$$0 \leq \delta_r \leq \frac{1}{2} \min \left\{ \frac{c_0}{\|c\|_\infty^2}, \frac{1}{\varepsilon \|H_r\|_2 \|S_r^{-1}\|_2} \right\}. \quad (22)$$

Then

$$a_{\text{SUPG},r}(v_r, v_r) \geq \frac{1}{2} \|v_r\|_{\text{SUPG},r}^2 \quad \forall v_r \in X_r, \quad (23)$$

where

$$\|v_r\|_{\text{SUPG},r} = \left(\varepsilon |v_r|_1^2 + c_0 \|v_r\|_0^2 + \delta_r \|\mathbf{b} \cdot \nabla v_r\|_0^2 \right)^{1/2}.$$

Proof. The proof follows the standard techniques found, e.g., in [6]. One gets, using integration by parts, (3), the Cauchy–Schwarz inequality, and Young’s inequality

$$a_{\text{SUPG},r}(v_r, v_r) \geq \varepsilon |v_r|_1^2 + c_0 \|v_r\|_0^2 + \delta_r \|\mathbf{b} \cdot \nabla v_r\|_0^2 - \delta_r \sum_{K \in \mathcal{T}_h} \varepsilon^2 \|\Delta v_r\|_{0,K}^2 - \delta_r \|c\|_\infty^2 \|v_r\|_0^2 - \frac{\delta_r}{2} \|\mathbf{b} \cdot \nabla v_r\|_0^2.$$

For estimating the term with the Laplacian, one can either use the finite element inverse estimate (14) or the POD inverse estimate (16). Inserting in either case the restrictions on the stabilization parameter, proves the statement of the lemma. \square

Note that the second restrictions in (21) and (22) are only necessary if finite elements are used where the restriction of the Laplacian to a mesh cell does not vanish.

3.5. On error estimates involving the SUPG-ROM solution

The first approach considers the error between u^n and the SUPG-ROM solution directly, using the splitting

$$u^n - (\bar{u}_h + u_r^n) = (u^n - P_r(u_h^n)) + (P_r(u_h^n) - (\bar{u}_h + u_r^n)) = \eta^n - \psi_r^n.$$

Subtracting the SUPG-ROM problem (10) from the continuous equation (2) and using the fact that the residual of the solution of (2) vanishes almost everywhere leads to

$$\begin{aligned} a_{\text{SUPG},r}(\psi_r^n, \psi_r^n) &= a_{\text{SUPG},r}(\eta^n, \psi_r^n) + \left(\partial_t u^n - \frac{u_r^n - u_r^{n-1}}{\Delta t}, \psi_r^n \right) \\ &\quad + \delta_r \left(\partial_t u^n - \frac{u_r^n - u_r^{n-1}}{\Delta t}, \mathbf{b} \cdot \nabla \psi_r^n \right). \end{aligned} \quad (24)$$

The second approach consists of considering the error between u_h^n and the SUPG-ROM solution and using the decomposition

$$u_h^n - (\bar{u}_h + u_r^n) = (u_h^n - P_r(u_h^n)) + (P_r(u_h^n) - (\bar{u}_h + u_r^n)) = \eta_h^n - \psi_r^n.$$

Since $X_r \subset X_h$, POD functions can be used as test functions in the finite element problem (4). An error equation is obtained by subtracting the SUPG-ROM problem (10) from (4), which is possible for all test functions in X_r . Next, the test function $\psi_r^n \in X_r$ is used and the right-hand side f is replaced by the solution of the continuous problem (1).

Noting that $(\eta_h^n, \psi_r^n) = 0$ since $P_r(u_h^n)$ is the L^2 projection of u_h^n into X_r and similarly $(\eta_h^{n-1}, \psi_r^n) = 0$, these steps lead to the error equation at time t^n

$$\begin{aligned} & \|\psi_r^n\|_0^2 + \Delta t a_{\text{SUPG},r}(\psi_r^n, \psi_r^n) = (\psi_r^{n-1}, \psi_r^n) + \Delta t a_{\text{SUPG},r}(\eta_h^n, \psi_r^n) \\ & + \Delta t (\delta_h - \delta_r) \sum_{K \in \mathcal{T}_h} (-\varepsilon \Delta(u_h^n - u^n) + \mathbf{b} \cdot \nabla(u_h^n - u^n) + c(u_h^n - u^n), \mathbf{b} \cdot \nabla \psi_r^n)_K \\ & + \delta_h (u_h^n - u_h^{n-1} - \Delta t \partial_t u^n, \mathbf{b} \cdot \nabla \psi_r^n) - \delta_r (u_r^n - u_r^{n-1} - \Delta t \partial_t u^n, \mathbf{b} \cdot \nabla \psi_r^n). \end{aligned} \tag{25}$$

Ideally, one would obtain optimal choices for δ_r by deriving an error estimate from (24) or (25). However, it is not known if such a derivation is possible. Even if it is possible, one has to expect that in the general case δ_r depends on the length of the time step like the finite element error estimate in [7]. Numerical evidence for simulations of the finite element problem shows that the stabilization parameter should not depend on the length of the time step; see the discussion of this topic in Section 2. In [7], an error estimate with the stabilization parameter depending on the mesh width as in the steady-state case was proven in a simplified case. Here, also a simplified situation will be considered, which also ensures that the stabilization parameter does not depend on the length of the time step: the steady-state situation will be studied. To this end, all dependencies of the previous results on the length of the time step will be neglected in the following. Thus, the error equation (24) simplifies to

$$a_{\text{SUPG},r}(\psi_r^n, \psi_r^n) = a_{\text{SUPG},r}(\eta^n, \psi_r^n) \tag{26}$$

and the error equation (25) to

$$\begin{aligned} a_{\text{SUPG},r}(\psi_r^n, \psi_r^n) &= a_{\text{SUPG},r}(\eta_h^n, \psi_r^n) \\ &+ (\delta_h - \delta_r) \sum_{K \in \mathcal{T}_h} (-\varepsilon \Delta(u_h^n - u^n) + \mathbf{b} \cdot \nabla(u_h^n - u^n) + c(u_h^n - u^n), \mathbf{b} \cdot \nabla \psi_r^n)_K. \end{aligned} \tag{27}$$

3.5.1. Stabilization parameters obtained with (26)

The estimate of the right-hand side of (26) is obtained in the same way as it is found in the literature, e.g., see [6]. With the Cauchy–Schwarz inequality, one obtains

$$\varepsilon (\nabla \eta^n, \nabla \psi_r^n) \leq \varepsilon \|\nabla \eta^n\|_0 \|\nabla \psi_r^n\|_0 \leq \varepsilon^{1/2} \|\nabla \eta^n\|_0 \|\psi_r^n\|_{\text{SUPG},r}.$$

Using integration by parts for the convective term gives the estimate

$$(\mathbf{b} \cdot \nabla \eta^n + c \eta^n, \psi_r^n) \leq \left(\frac{\|\nabla \cdot \mathbf{b}\|_\infty + \|c\|_\infty}{c_0^{1/2}} + \delta_r^{-1/2} \right) \|\eta^n\|_0 \|\psi_r^n\|_{\text{SUPG},r}.$$

Straightforward estimates lead to

$$\delta_r (\mathbf{b} \cdot \nabla \eta^n, \mathbf{b} \cdot \nabla \psi_r^n) \leq \delta_r^{1/2} \|\mathbf{b} \cdot \nabla \eta^n\|_0 \|\psi_r^n\|_{\text{SUPG},r}, \tag{28}$$

$$\delta_r (c \eta^n, \mathbf{b} \cdot \nabla \psi_r^n) \leq \delta_r^{1/2} \|c\|_\infty \|\eta^n\|_0 \|\psi_r^n\|_{\text{SUPG},r},$$

$$\delta_r \sum_{K \in \mathcal{T}_h} \varepsilon (-\Delta \eta^n, \mathbf{b} \cdot \nabla \psi_r^n)_K \leq \delta_r^{1/2} \varepsilon \left(\sum_{K \in \mathcal{T}_h} \|\Delta \eta^n\|_{0,K}^2 \right)^{1/2} \|\psi_r^n\|_{\text{SUPG},r}. \tag{29}$$

Using the coercivity (23) of the SUPG-ROM bilinear form, inserting all terms into (26), and including all data from convection and reaction into the constant yields

$$\begin{aligned} \|\psi_r^n\|_{\text{SUPG},r} &\leq C \left[\left(1 + \delta_r^{-1/2} + \delta_r^{1/2} \right) \|\eta^n\|_0 + \varepsilon^{1/2} \|\nabla \eta^n\|_0 \right. \\ &\quad \left. + \delta_r^{1/2} \|\mathbf{b} \cdot \nabla \eta^n\|_0 + \delta_r^{1/2} \varepsilon \left(\sum_{K \in \mathcal{T}_h} \|\Delta \eta^n\|_{0,K}^2 \right)^{1/2} \right]. \end{aligned} \tag{30}$$

The second factor on the right-hand side of (30) shall be minimized, thereby providing information about an appropriate choice of the stabilization parameter.

3.5.1.1. The finite element option. The straightforward approach consists in decomposing

$$\eta^n = u^n - P_r(u_h^n) = (u^n - u_h^n) + (u_h^n - P_r(u_h^n)) \quad (31)$$

and using the error estimates (17) and (18) for the first part and the estimate (20) for the second part. The term to minimize becomes

$$\begin{aligned} & \left(1 + \delta_r^{-1/2} + \delta_r^{1/2}\right) \left(h^{m+1/2} + \Lambda_0\right) + h^{m+1/2} + \varepsilon^{1/2} h^{-1} \Lambda_0 + \delta_r^{1/2} \delta_h^{-1/2} h^{m+1/2} \\ & + \delta_r^{1/2} h^{-1} \Lambda_0 + \delta_r^{1/2} \varepsilon h^{m-1} + \delta_r^{1/2} \varepsilon^{1/2} h^{m-1/2} + \delta_r^{1/2} \varepsilon h^{-2} \Lambda_0 \end{aligned}$$

with

$$\Lambda_0 = \left(\sum_{j=r+1}^R \lambda_j \right)^{1/2}.$$

Standard calculus, using $\delta_h = h$, gives the minimum

$$\delta_r = \frac{h^{m+1/2} + \Lambda_0}{h^{m-1}(h^{3/2} + h + \varepsilon + \varepsilon^{1/2} h^{1/2}) + \Lambda_0(1 + h^{-1} + \varepsilon h^{-2})}.$$

Concentrating on the most important terms in the convection-dominated case $\varepsilon \ll h$ and for grid widths $h < 1$, one gets

$$\delta_r = h \frac{h^{m+1/2} + \Lambda_0}{h^{m+1} + \Lambda_0}.$$

Numerical evidence, e.g., Remark 3.4, shows that Λ_0 dominates the finite element errors, such that in this case the stabilization parameter becomes

$$\delta_r^{\text{FE}} = h. \quad (32)$$

The SUPG-ROM using δ_r^{FE} is denoted by FE-SUPG-ROM.

Remark 3.3. Some remarks on (32) are as follows:

- In the convection-dominated case, condition (21) for the coercivity of the SUPG-ROM bilinear form will be satisfied for δ_r^{FE} .
- There is an explicit impact of the setup for simulating the snapshots onto the stabilization used in the SUPG-ROM. It is not clear, if this situation is always desirable, e.g., if the snapshots were computed on a very fine mesh, there would be only a weak stabilization in the SUPG-ROM.
- For using δ_r^{FE} , one has to know the mesh width. If even the mesh itself is known, then it is possible to use also the local mesh width in assembling the terms for the stabilization, like usually done in the finite element method.
- There is no impact of the number of used snapshots or POD modes on δ_r^{FE} .

3.5.1.2. The POD option. Instead of using (20) for the second part of the decomposition (31), one can apply (19).

Hypothesis 3.1. Let the finite element simulation be sufficiently accurate or let sufficiently few POD basis functions be used in the ROM, such that the norms on the left-hand sides of (17) and (18) can be estimated by a constant times the respective terms of the right-hand side of (19).

This hypothesis implies that there is a constant C_A such that

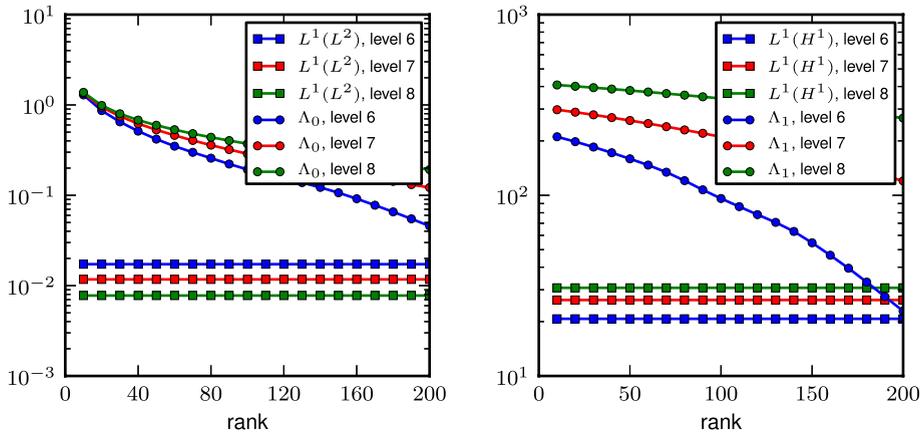


Fig. 2. Remark 3.4: Numerical verification of Hypothesis 3.1 with the help of the test example in Remark 3.2. Here, $L^1(L^2)$ and $L^1(H^1)$ are abbreviations for $\frac{1}{N} \sum_{n=1}^N \|u^n - u_h^n\|_0$ and $\frac{1}{N} \sum_{n=1}^N \|\nabla(u^n - u_h^n)\|_0$, respectively.

$$\begin{aligned} \|\eta^n\|_0 &\leq C_A \Lambda_0, \\ \|\nabla \eta^n\|_0 &\leq C_A \left(\sum_{j=r+1}^R \lambda_j |\varphi_j|_1^2 \right)^{1/2} = C_A \Lambda_1, \\ \left(\sum_{K \in \mathcal{T}_h} \|\Delta \eta^n\|_{0,K}^2 \right)^{1/2} &\leq C_A \left(\sum_{j=r+1}^R \lambda_j |\varphi_j|_2^2 \right)^{1/2} = C_A \Lambda_2. \end{aligned}$$

Note that the constant C_A will cancel in the further calculations such that its actual value does not influence the final result.

Remark 3.4. Considering the same problem as in Remark 3.2, one obtains the curves depicted in Fig. 2 for the errors of the finite element solution on different meshes (levels 6, 7, and 8 are introduced in the beginning of Section 4) and for the corresponding values Λ_0 and Λ_1 for different dimensions of the ROM basis. One can observe that in these cases Hypothesis 3.1 is satisfied.

Using the estimate $\|\mathbf{b} \cdot \nabla \eta^n\|_0 \leq C \|\nabla \eta^n\|_0$, the factor to be minimized in (30) has the form

$$\left(1 + \delta_r^{-1/2} + \delta_r^{1/2}\right) C_A \Lambda_0 + \left(\varepsilon^{1/2} + \delta_r^{1/2}\right) C_A \Lambda_1 + \delta_r^{1/2} \varepsilon C_A \Lambda_2,$$

which leads to the minimum

$$\delta_r = \frac{\Lambda_0}{\Lambda_0 + \Lambda_1 + \varepsilon \Lambda_2}. \tag{33}$$

In the convection-dominated regime, the last term in the denominator will be small. From estimate (20), it can be expected that $\Lambda_0 \ll \Lambda_1$, at least for small mesh widths. Thus, an appropriate choice of the stabilization parameter is

$$\delta_r^{\text{POD}} = \frac{\Lambda_0}{\Lambda_1}. \tag{34}$$

The SUPG-ROM using δ_r^{POD} is denoted by POD-SUPG-ROM.

Remark 3.5. For the problem described in Remark 3.2, one gets for the values in the denominator of (33) the curves presented in Fig. 3. Hence, for this example, the assumptions made for reducing (33) to (34) are satisfied. The same behavior was observed also for other test cases.

Remark 3.6. Some remarks on (34) are as follows:

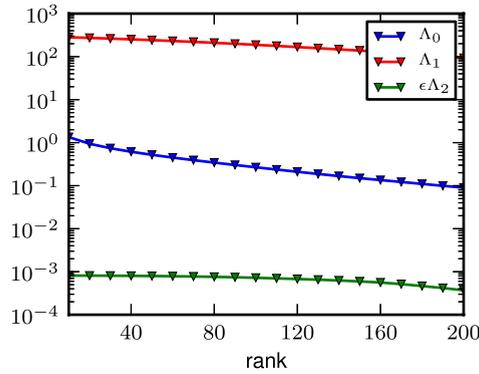


Fig. 3. Remark 3.5: Curves for Λ_0 , Λ_1 , and $\varepsilon\Lambda_2$ for the problem from Remark 3.2.

- It is not clear a priori that either (21) or (22) are satisfied by δ_r^{POD} . In the numerical simulations, where the grids for computing the snapshots were not extremely fine, it was found that generally $\delta_r^{\text{POD}} \leq \delta_r^{\text{FE}}$, which implies the satisfaction of (21).
- The parameter δ_r^{POD} is influenced by the number r of used POD modes and also by the simulation for computing the snapshots, since this simulation determines the eigenvalues and eigenfunctions.
- For computing δ_r^{POD} , one has to consider in the offline step of the ROM simulation all POD modes, because they are necessary for computing Λ_1 .
- There is no possibility to localize δ_r^{POD} .

3.5.2. Stabilization parameters obtained with (27)

Most terms of the first term on the right-hand side of (27) are estimated in the same way as in Section 3.5.1. Only, instead of (28),

$$\delta_r (\mathbf{b} \cdot \nabla \eta^n, \mathbf{b} \cdot \nabla \psi_r^n) \leq C \delta_r^{1/2} \|\nabla \eta^n\|_0 \|\psi_r^n\|_{\text{SUPG},r}$$

is used, giving the upper bound

$$C \left[\left(1 + \delta_r^{-1/2} + \delta_r^{1/2}\right) \|\eta_h^n\|_0 + \left(\varepsilon^{1/2} + \delta_r^{1/2}\right) \|\nabla \eta_h^n\|_0 + \delta_r^{1/2} \varepsilon \left(\sum_{K \in \mathcal{T}_h} \|\Delta \eta_h^n\|_{0,K}^2 \right)^{1/2} \right] \|\psi_r^n\|_{\text{SUPG},r}.$$

For estimating the second term of (27), the finite element error estimates (17) and (18) are utilized, leading to the upper bound

$$C |\delta_h - \delta_r| \delta_r^{-1/2} \left(\varepsilon h^{m-1} + \varepsilon^{1/2} h^{m-1/2} + \delta_h^{-1/2} h^{m+1/2} + h^{m+1/2} \right) \|\psi_r^n\|_{\text{SUPG},r}.$$

By inserting these bounds into (27) and using $\delta_h = h$, one gets the estimate

$$\begin{aligned} \|\psi_r^n\|_{\text{SUPG},r} \leq & C \left[\left(1 + \delta_r^{-1/2} + \delta_r^{1/2}\right) \|\eta_h^n\|_0 + \left(\varepsilon^{1/2} + \delta_r^{1/2}\right) \|\nabla \eta_h^n\|_0 \right. \\ & + \delta_r^{1/2} \varepsilon \left(\sum_{K \in \mathcal{T}_h} \|\Delta \eta_h^n\|_{0,K}^2 \right)^{1/2} + |\delta_h - \delta_r| \delta_r^{-1/2} \\ & \left. \times \left(\varepsilon h^{m-1} + \varepsilon^{1/2} h^{m-1/2} + h^m + h^{m+1/2} \right) \right]. \end{aligned}$$

From this estimate, one can determine the parameter δ_r which minimizes the right-hand side again with standard calculus. Restricting the considerations to the convection-dominated case and to the dominant terms, one finds the same parameters δ_r^{FE} and δ_r^{POD} as in (32) and (34), respectively, depending on using the estimates (20) or (19). For the sake of brevity, details of the calculations leading to these results will be omitted here.

4. Numerical studies

The numerical investigations aim at answering the following questions:

- *Question 1:* Does the SUPG-ROM yield more accurate results than the G-ROM?
- *Question 2:* Which of the two stabilization parameters of the SUPG-ROM (derived in Sections 3.5.1 and 3.5.2) yields more accurate results?

To answer Question 1, a variety of two-dimensional tests were considered. For the sake of brevity, only results for the case that the finite element mesh was significantly coarser than the width of the internal layers will be presented below, since this is the common situation in applications. If the mesh width $h \in [10^{-3}, 10^{-2}]$ and the layer width $\mathcal{O}(\sqrt{\varepsilon})$ were comparable, the SUPG-ROM and the G-ROM performed similarly in the numerical tests.

For all the ROMs investigated, the SUPG method (4) with P_2 finite elements and the backward Euler time integrator with small time steps were used to generate the snapshots. All test problems were defined in the unit square. For the coarsest grid (level 0), the unit square was divided by the diagonal from bottom left to top right into two triangles. Uniform grid refinement was applied for constructing the finer grids. Snapshots were computed on levels 6, 7, and 8, with 16 641, 66 049, 263 169 degrees of freedom (including Dirichlet nodes), respectively. For the sake of brevity, only results for two test problems are presented: A hump changing its height (Example 4.1) and a traveling wave (Example 4.2). The results from these two tests are representative for the other tests that were investigated. In all cases, the POD modes were computed with respect to the $L^2(\Omega)$ inner product and with the centered-trajectory method, i.e., from the fluctuating parts of the snapshots.

Three types of ROMs were studied. The first one is the G-ROM, i.e., the ROM without any stabilization. It is known that solutions of convection-dominated problems can be approximated accurately with the Galerkin finite element method if the finite element space possesses sufficient information about the considered problem and its solution. This information can be used to construct suitable grids, e.g., layer-adapted grids as in [39], or to design appropriate finite element functions, e.g., exponentially fitted functions as in [40]. For the ROM, the POD modes already contain important features of the solution. The questions are if the information contained in these basis functions suffices to stabilize the G-ROM and if yes, how the accuracy of the computational results compares with the accuracy of the results obtained with the SUPG-ROMs. The same questions were posed in [25], where stabilized ROMs were used for convection-dominated convection–diffusion–reaction equations. In the “Offline-only” approach in [25], the SUPG method was used in the offline stage, but not in the online stage. Thus, the “Offline-only” approach is similar to the G-ROM considered in this paper. In the “Offline–Online” approach in [25], the SUPG method was used both in the offline and online stages. Thus, the “Offline–Online” approach is similar to the SUPG-ROM considered in this paper. The other two methods studied in this section are ROMs with SUPG stabilization (10), where the FE-SUPG-ROM uses the stabilization parameter δ_r^{FE} from (32) and the POD-SUPG-ROM the stabilization parameter δ_r^{POD} from (34). From the practical point of view, the computation of the stabilization parameter δ_r^{FE} is much easier than the computation of δ_r^{POD} . The former parameter is equal to the mesh width h ; no additional information is needed. The latter one requires storage of all R POD modes and eigenvalues. Moreover, Λ_0 and Λ_1 have to be calculated, which can be time-consuming for problems with a high-dimensional snapshot space X_R . However, these values have to be computed only once in the offline step.

Analytical solutions are known for the presented examples. Thus, besides plots of the computed ROM solutions, errors can be used to evaluate the results. In the following, the discrete analog of the $L^1(0, T; L^2(\Omega))$ error, given, e.g., for the ROM solution by

$$\frac{1}{N+1} \sum_{n=0}^N \|u^n - (\bar{u}_h + u_r^n)\|_0,$$

will be considered, where u^n denotes the solution of the continuous problem at time t^n .

The code MoonNMD [41] was used to run the numerical experiments.

Example 4.1 (Hump Changing its Height). This example is taken from [3]. It is defined in $\Omega = (0, 1)^2$ and $(0, T) = (0, 2)$. The coefficients of (1) were chosen to be $\varepsilon = 10^{-6}$, $\mathbf{b} = (2, 3)^T$ and $c = 1$. There is a prescribed

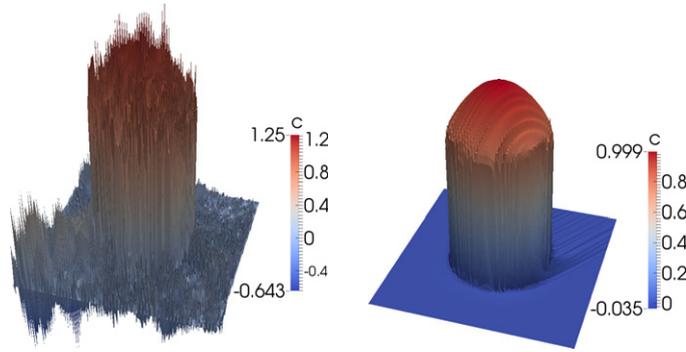


Fig. 4. Example 4.1: Galerkin finite element method (left), SUPG finite element method (right) at $t = 0.5$.

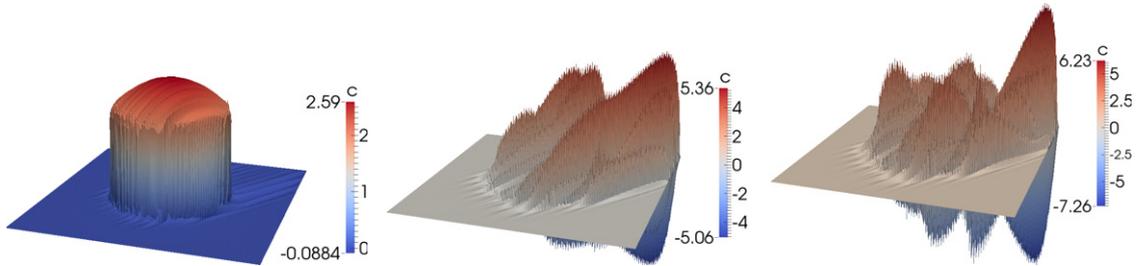


Fig. 5. Example 4.1: POD modes φ_1 , φ_2 , and φ_3 (from left to right).

solution of the form

$$u(t, x, y) = 16 \sin(\pi t) x(1 - x)y(1 - y) \times \left[\frac{1}{2} + \frac{\arctan(2\varepsilon^{-1/2} (0.25^2 - (x - 0.5)^2 - (y - 0.5)^2))}{\pi} \right]. \tag{35}$$

The forcing term f , the initial condition u_0 , and the boundary conditions were set such that (35) satisfies the boundary value problem. The solution (35) possesses an internal layer of size $\mathcal{O}(\sqrt{\varepsilon})$.

The finite element problem for computing the snapshots was solved on level 7, such that $h = 1.1 \cdot 10^{-2}$, and the backward Euler scheme was applied with the time step $\Delta t = 10^{-3}$. Since the problem is convection-dominated and the solution has a layer, the use of a stabilized discretization is necessary; see Fig. 4 for a comparison of snapshots from the Galerkin finite element method and the SUPG method (4). Whereas the solution of the Galerkin finite element method is globally polluted with spurious oscillations, there are only few oscillations, mainly in the right upper part of the domain, in the solution computed with the SUPG method.

For computing the POD basis, every fifth solution was stored such that 401 snapshots were used. If the finite element method accurately resolved all layers then the POD basis would consist just of one mode representing the time-independent parts of (35). However, such a method is not known so far and in the simulations of convection-dominated problems one has always to expect numerical artifacts. Because of the spurious oscillations of the SUPG method, one obtains 14 POD modes; see Fig. 5 for the first POD modes and Fig. 6 for the corresponding eigenvalues, where the POD modes for $r > 1$ come from the spurious oscillations of the finite element solution.

Figs. 7 and 8 present results for the three considered ROMs. In Fig. 7, the temporal evolution of the error in the $L^2(\Omega)$ norm for $r = 9$ and the errors in the discrete $L^1(0, T; L^2(\Omega))$ norm are shown. Corresponding numerical solutions for $r = 9$ are depicted in Fig. 8.

Before answering Questions 1 and 2, the following important observation should be made: The $L^1(0, T; L^2(\Omega))$ error in the right panel of Fig. 7 shows that adding more POD modes that represent oscillations (i.e., POD modes $\varphi_2, \varphi_3, \dots, \varphi_{14}$) results in a continuous increase of the error. This behavior is in clear contrast with the standard POD-ROM experience, where adding more POD modes generally reduces the error. The reason for this behavior is that the POD uses noisy data, resulting in POD modes which contain mostly information on the numerical artifacts of the finite

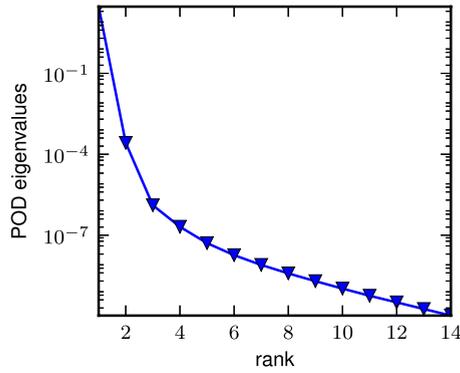


Fig. 6. Example 4.1: POD eigenvalues.

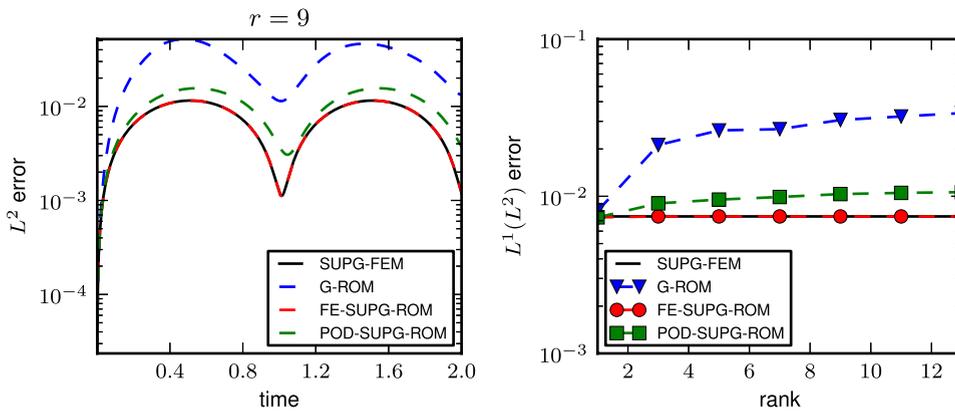


Fig. 7. Example 4.1: Errors for different ROMs, $L^2(\Omega)$ error (left), $L^1(0, T; L^2(\Omega))$ error (right).

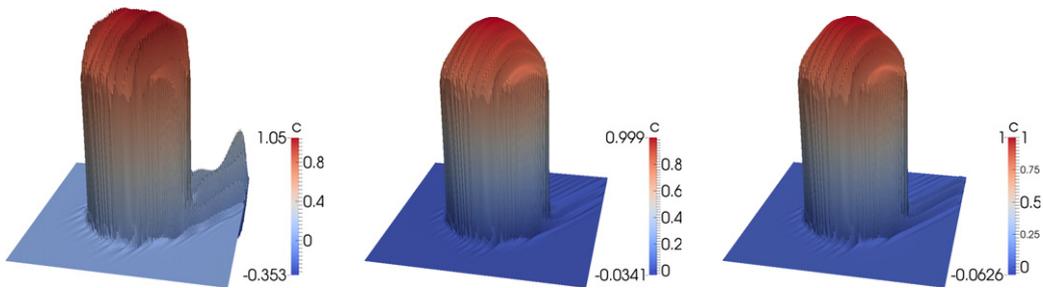


Fig. 8. Example 4.1: Solution at $t = 0.5$ for G-ROM, FE-SUPG-ROM, POD-SUPG-ROM (from left to right) for $r = 9$.

element solution. As already discussed above, with most of the finite element methods, the appearance of such modes is inevitable. Unlike the present example, in general it is not known which modes are strongly influenced by the noise. In adding more and more POD modes, it is important that the ROM can suppress the influence of such noisy modes.

The answer to Question 1 is given by Figs. 7 and 8: Both the FE-SUPG-ROM and the POD-SUPG-ROM yield more accurate results than the G-ROM for $r \geq 2$. The stabilized ROMs can compute good solutions even if POD modes strongly influenced by spurious oscillations are used. Question 2 is answered in Fig. 7: The FE-SUPG-ROM performs better than the POD-SUPG-ROM, which seems to be due to the larger stabilization parameters, see Fig. 9.

Example 4.2 (Traveling Wave). This example is similar to the one used in [42]. It is given in $\Omega = (0, 1)^2$ and $(0, T) = (0, 1)$ with coefficients of (1) chosen as $\varepsilon = 10^{-8}$, $\mathbf{b} = (\cos(\pi/3), \sin(\pi/3))$, and $c = 1$. The analytical

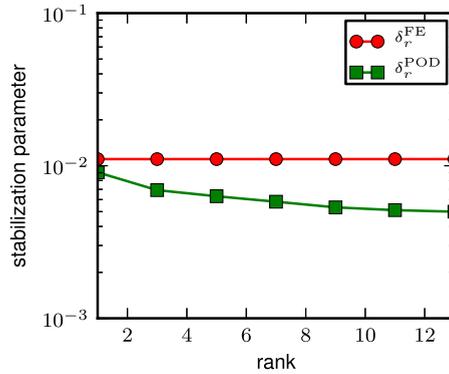


Fig. 9. Example 4.1: Stabilization parameters for the SUPG-ROMs.

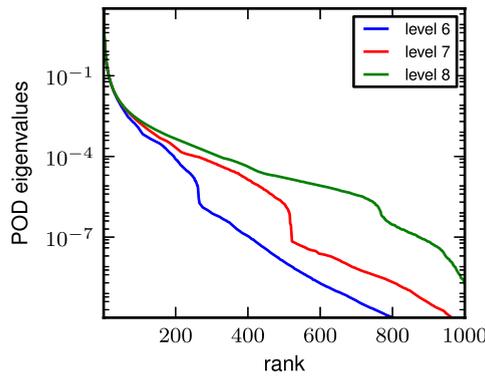


Fig. 10. Example 4.2: POD eigenvalues.

solution is defined by

$$u(t, x, y) = 0.5 \sin(\pi x) \sin(\pi y) \left[\tanh \left(\frac{x + y - t - 0.5}{\sqrt{\varepsilon}} \right) + 1 \right]. \tag{36}$$

The right-hand side f , the initial condition u_0 , and the boundary condition were chosen such that (36) satisfies the boundary value problem. Solution (36) possesses a moving internal layer of width $\mathcal{O}(\sqrt{\varepsilon})$.

To investigate the sensitivity of the numerical results with respect to the mesh width h , the finite element problem for computing the snapshots was solved on levels 6, 7, and 8. The backward Euler scheme was applied with a time step $\Delta t = 10^{-4}$. Example 4.2 is a convection-dominated problem and therefore the SUPG stabilization was applied to the finite element method, just as in Example 4.1. For computing the POD basis, every tenth solution was stored such that 1001 snapshots were used. Note that Example 4.2 is more complex than Example 4.1 since the position of the layer depends on time. The complexity is reflected by the dimension of the snapshot space X_R , which is 14 for Example 4.1 and between 800 and 1000, depending on the underlying spatial level, for Example 4.2 (see Fig. 10 with the corresponding POD eigenvalues). The plot in Fig. 10 also shows that the snapshot space X_R changes with the mesh width h , its dimension increases with decreasing h . Also in this example, the studied ROMs use noisy POD data, which, as explained, is inevitable for convection-dominated problems on realistic meshes.

In Figs. 11–13, the errors in the discrete $L^1(0, T; L^2(\Omega))$ norm are shown for the three considered ROMs and for spatial levels 6, 7, and 8. On every spatial level, a similar behavior to that in Example 4.1 is observed: Increasing the number of POD modes eventually results in an increase of the G-ROM error. This time, however, this error increase is observed later than in Example 4.1 (around $r = 70$ for level 6, $r = 130$ for level 7, and $r = 270$ for level 8). Like in Example 4.1, the G-ROM error grows due to the fact that the impact of noise becomes more and more dominant for modes with higher indices. Note that the finer the mesh, the higher the threshold rank is. Furthermore, below this threshold the discrete $L^1(0, T; L^2(\Omega))$ errors of the G-ROM and the SUPG-ROMs are similar. The answer to

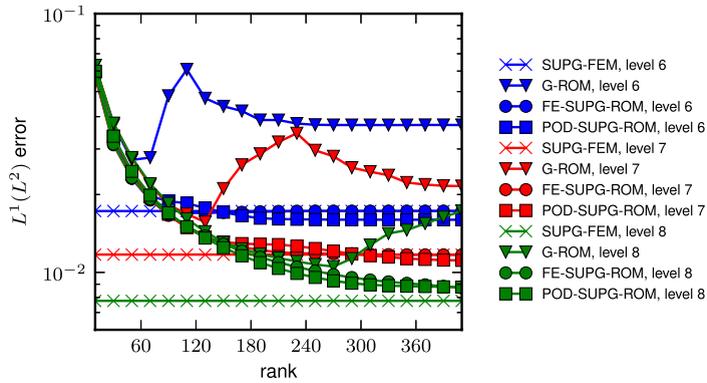


Fig. 11. Example 4.2: $L^1(0, T; L^2(\Omega))$ error for different ROMs and three different spatial levels.

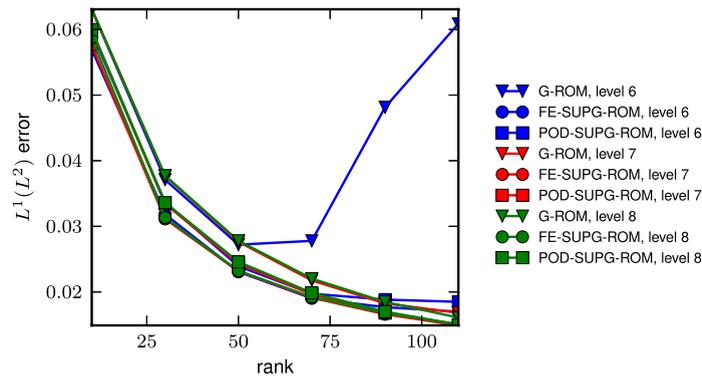


Fig. 12. Example 4.2: $L^1(0, T; L^2(\Omega))$ error for different ROMs, three different spatial levels, and small r .

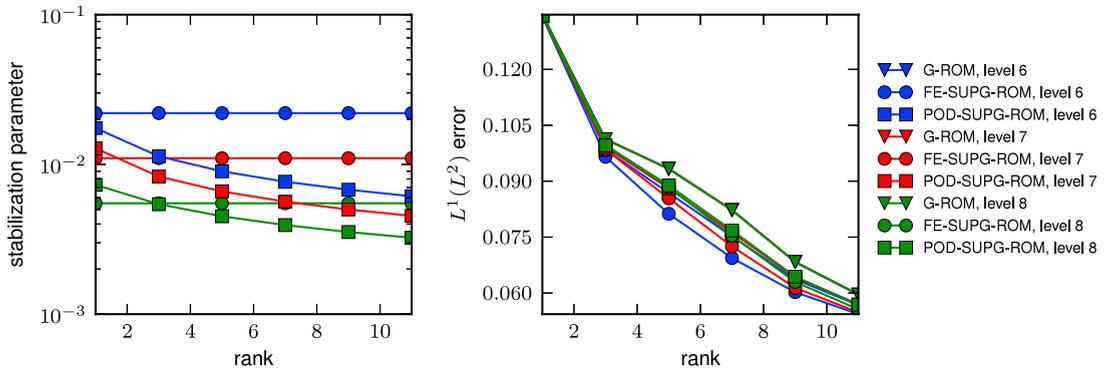


Fig. 13. Example 4.2: Different ROMs at three different spatial levels and for very small r : Stabilization parameters (left) and $L^1(0, T; L^2(\Omega))$ error (right). The G-ROM error curves for all spatial levels are the same.

Question 1 is given by Figs. 11–14. For small numbers of POD modes, all ROMs have a similar error in the discrete $L^1(0, T; L^2(\Omega))$ norm and for large numbers of POD modes, both the FE-SUPG-ROM and POD-SUPG-ROM yield more accurate results than the G-ROM. With respect to the size of the spurious oscillations, the SUPG-ROMs are always better than the G-ROM. In Figs. 13 and 15, the used stabilization parameters δ_r^{FE} and δ_r^{POD} for three spatial levels are shown. Even if the representation of δ_r^{POD} in (34) has no explicit dependence on the mesh width h , the stabilization parameter seems to depend implicitly on the underlying grid. This behavior is expected as \mathcal{A}_0 and \mathcal{A}_1 in (34) are computed using the POD modes and eigenvalues resulting from the finite element simulation. In particular, the value of δ_r^{POD} decreases for finer grids. The difference between the parameters δ_r^{FE} and δ_r^{POD} becomes more

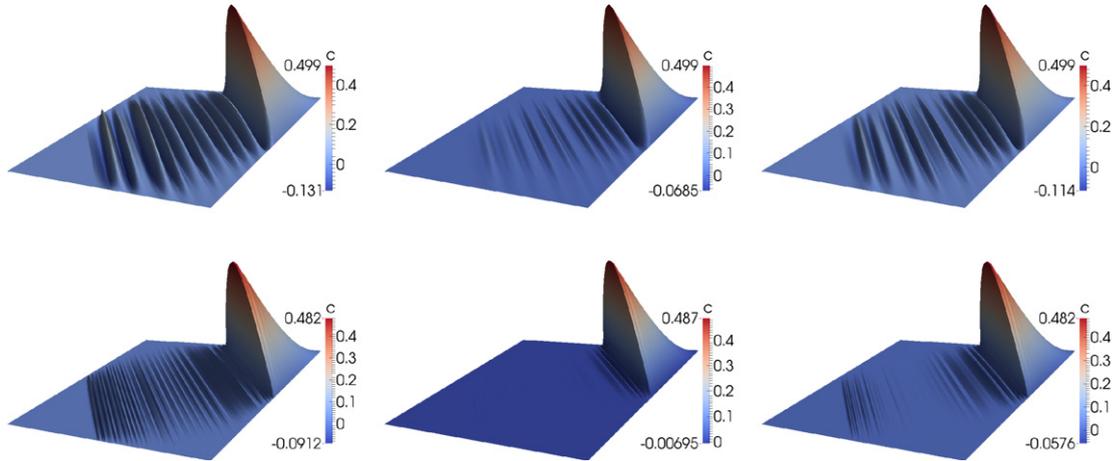


Fig. 14. Example 4.2 for level 7: Solution at $t = 1.0$ for G-ROM, FE-SUPG-ROM, POD-SUPG-ROM (from left to right) for $r = 30, 130$ (from top to bottom).

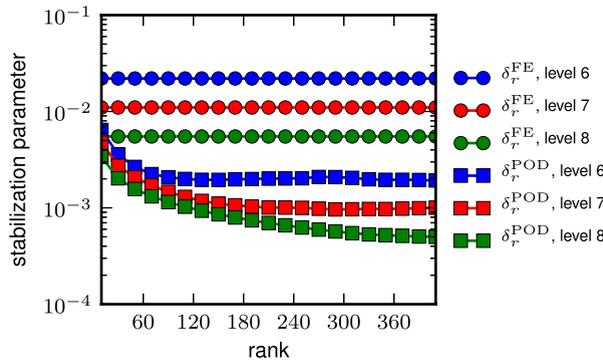


Fig. 15. Example 4.2: Stabilization parameters for the SUPG-ROMs.

pronounced for large r . Question 2 can be studied with the results presented in Figs. 11–14. Concerning the spurious oscillations, the results obtained with FE-SUPG-ROM are always better. With respect to the error in the discrete $L^1(0, T; L^2(\Omega))$ norm, the POD-SUPG-ROM performs slightly better than FE-SUPG-ROM for larger values of r . The explanation of both observations comes again from the different sizes of the stabilization parameters, see Fig. 15. Since δ_r^{FE} is larger than δ_r^{POD} , the spurious oscillations in the FE-SUPG-ROM are damped more efficiently than in the POD-SUPG-ROM. To investigate why the POD-SUPG-ROM for larger r yielded slightly more accurate results than the SUPG finite element method in Fig. 11, additional SUPG finite element simulations with $\delta_h = \delta_r^{POD} = 0.2h$ were carried out. The resulting discrete $L^1(0, T; L^2(\Omega))$ error was comparable to that of the POD-SUPG-ROM for larger r in Fig. 11. The plots in Figs. 11–13 also show that as the mesh width h changes, the ROM results change as well, but the qualitative behavior of the ROMs (and in particular the answers to Questions 1 and 2) remains unchanged. The plots in Fig. 13, however, show that for very low r as the mesh width decreases the difference between the stabilization parameters and the difference between the errors of the two SUPG-ROMs decrease as well.

5. Summary and outlook

This paper presented a theoretical and numerical investigation of an SUPG-ROM for convection-dominated convection/diffusion/reaction equations. At a theoretical level, numerical analysis was used to suggest the stabilization parameter applied in the SUPG-ROM. Two scalings for the stabilization parameter were proposed: One based on the underlying finite element discretization (which yields the FE-SUPG-ROM) and one based on the POD truncation (which yields the POD-SUPG-ROM). At a numerical level, the G-ROM, the FE-SUPG-ROM, and the POD-SUPG-ROM were tested on a variety of convection-dominated convection/diffusion/reaction problems. For clarity, results

were presented for two representative test problems that display sharp internal layers. The numerical investigations yielded the following conclusions:

- If the finite element discretization was fine enough to capture the internal layer (and thus no numerical stabilization was needed), i.e., $h \approx \mathcal{O}(\sqrt{\varepsilon}) \in [10^{-3}, 10^{-2}]$, then the standard G-ROM yielded accurate results that the SUPG-ROM could not improve upon.
- On relatively coarse meshes, which is the usual situation encountered in practice, a SUPG finite element discretization was used to generate the snapshots. This approach led to POD modes containing numerical artifacts (spurious oscillations). Thus, the considered ROMs used noisy POD data.
- The standard G-ROM was stable, i.e., the solutions were not globally polluted with huge spurious oscillations. It yielded comparable results to the SUPG-ROMs with respect to the discrete $L^1(0, T; L^2(\Omega))$ error only if sufficiently few POD modes were used. Once the number of POD modes was increased above a certain limit, the noise of these modes was reflected strongly in the results of the G-ROM.
- Both the FE-SUPG-ROM and the POD-SUPG-ROM yielded results that were significantly more accurate than those for the G-ROM for larger numbers of used POD modes. The noise of the POD modes was suppressed by both SUPG-ROMs much better than by the G-ROM.
- The exact meanings of “sufficiently few” and “large” numbers of POD modes in the previous points depend on the example and the mesh width. In practice, corresponding values for the number of POD modes will generally not be known.
- In the numerical studies, it was observed that δ_r^{POD} was generally smaller than δ_r^{FE} . Based on this experience, it is recommended that one uses δ_r^{FE} , i.e., FE-SUPG-ROM. This choice of the SUPG-ROM stabilization parameter suppressed the spurious oscillations in the ROM somewhat better than δ_r^{POD} and it was never observed that the results obtained with FE-SUPG-ROM were notably worse in any respect than the results of POD-SUPG-ROM. Moreover, the computation of δ_r^{FE} is easier than that of δ_r^{POD} .
- The sensitivity study with respect to the mesh width showed that, although the ROMs yield different results, their qualitative behavior remains unchanged.

This paper considered the SUPG method for both computing the snapshots and performing the ROM simulations, since this method is analytically well understood and the main purpose was to propose SUPG-ROM stabilization parameters based on analytical considerations. The numerical tests showed that the ROM solutions exhibit spurious oscillations. The next step consists in studying if it is possible to obtain snapshots without (or with fewer) spurious oscillations and to investigate if the availability of such snapshots is sufficient to obtain more accurate SUPG-ROM results than with the approach studied in this paper. A sensitivity study of the ROM results with respect to parameters, e.g., the diffusion coefficient, would also be an interesting research topic. After achieving a clear understanding of the issues and possible remedies for SUPG-ROMs of convection-dominated convection–diffusion–reaction equations, one could carry out a similar theoretical and numerical investigation of the significantly more challenging Navier–Stokes equations.

Acknowledgment

The second and fourth authors were partially supported by the National Science Foundation (DMS-1025314).

References

- [1] Matthias Augustin, Alfonso Caiazzo, André Fiebach, Jürgen Fuhrmann, Volker John, Alexander Linke, Rudolf Umla, An assessment of discretizations for convection-dominated convection–diffusion equations, *Comput. Methods Appl. Mech. Engrg.* 200 (47–48) (2011) 3395–3409.
- [2] Volker John, Petr Knobloch, On spurious oscillations at layers diminishing (SOLD) methods for convection–diffusion equations. I. A review, *Comput. Methods Appl. Mech. Engrg.* 196 (17–20) (2007) 2197–2215.
- [3] Volker John, Ellen Schmeyer, Finite element methods for time-dependent convection–diffusion–reaction equations with small diffusion, *Comput. Methods Appl. Mech. Engrg.* 198 (3–4) (2008) 475–494.
- [4] Alexander N. Brooks, Thomas J.R. Hughes, Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Comput. Methods Appl. Mech. Engrg.* 32 (1–3) (1982) 199–259. FENOMECH '81, Part I (Stuttgart, 1981).
- [5] T.J.R. Hughes, A. Brooks, A multidimensional upwind scheme with no crosswind diffusion, in: *Finite Element Methods For Convection Dominated Flows* (Papers, Winter Ann. Meeting Amer. Soc. Mech. Engrs., New York, 1979), in: AMD, vol. 34, Amer. Soc. Mech. Engrs. (ASME), New York, 1979, pp. 19–35.

- [6] H.G. Roos, M. Stynes, L. Tobiska, *Robust Numerical Methods for Singularly Perturbed Differential Equations: Convection-Diffusion-Reaction and Flow Problems*, second ed., in: Springer Series in Computational Mathematics, vol. 24, Springer, 2008.
- [7] V. John, J. Novo, Error analysis of the SUPG finite element discretization of evolutionary convection–diffusion–reaction equations, *SIAM J. Numer. Anal.* 49 (3) (2011) 1149–1176.
- [8] P. Holmes, J.L. Lumley, G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge, 1996.
- [9] T. Lassila, A. Manzoni, A. Quarteroni, G. Rozza, Model order reduction in fluid dynamics: challenges and perspectives, in: A. Quarteroni, G. Rozza (Eds.), *Reduced Order Methods For Modeling And Computational Reduction*, in: Modeling, Simulation and Applications, vol. 9, Springer, 2014, pp. 235–274.
- [10] B.R. Noack, M. Morzynski, G. Tadmor, *Reduced-Order Modelling for Flow Control*, Vol. 528, Springer-Verlag, 2011.
- [11] N. Aubry, W.Y. Lian, E.S. Titi, Preserving symmetries in the proper orthogonal decomposition, *SIAM J. Sci. Comput.* 14 (1993) 483–505.
- [12] D. Amsallem, C. Farhat, Stabilization of projection-based reduced-order models, *Internat. J. Numer. Methods Engrg.* 91 (4) (2012) 358–377.
- [13] I. Kalashnikova, S. van Bloemen Waanders, B. Arunajatesan, M. Barone, Stabilization of projection-based reduced order models for linear time-invariant systems via optimization-based eigenvalue reassignment, *Comput. Methods Appl. Mech. Engrg.* 272 (15) (2014) 251–270.
- [14] M.F. Barone, I. Kalashnikova, D.J. Segalman, H.K. Thornquist, Stable Galerkin reduced order models for linearized compressible flow, *J. Comput. Phys.* 228 (6) (2009) 1932–1946.
- [15] Maciej Balajewicz, Earl H. Dowell, Stabilization of projection-based reduced order models of the Navier-Stokes, *Nonlinear Dynam.* 70 (2) (2012) 1619–1632.
- [16] S. Sirisup, G.E. Karniadakis, A spectral viscosity method for correcting the long-term behavior of POD models, *J. Comput. Phys.* 194 (1) (2004) 92–116.
- [17] N. Aubry, P. Holmes, J.L. Lumley, E. Stone, The dynamics of coherent structures in the wall region of a turbulent boundary layer, *J. Fluid Mech.* 192 (1988) 115–173.
- [18] Joan Baiges, Ramon Codina, Sergio Idelsohn, Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier–Stokes equations, *Internat. J. Numer. Methods Fluids* 72 (12) (2013) 1219–1243.
- [19] M. Bergmann, C.H. Bruneau, A. Iollo, Enablers for robust POD models, *J. Comput. Phys.* 228 (2) (2009) 516–538.
- [20] A. Iollo, A. Dervieux, J.A. Désidéri, S. Lanteri, Two stable POD-based approximations to the Navier–Stokes equations, *Comput. Vis. Sci.* 3 (1) (2000) 61–66.
- [21] B. Kragel, *Streamline diffusion POD models in optimization*, (Ph.D. thesis) University of Trier, 2005.
- [22] Z. Wang, I. Akhtar, J. Borggaard, T. Iliescu, Proper orthogonal decomposition closure models for turbulent flows: A numerical comparison, *Comput. Methods Appl. Mech. Engrg.* 237–240 (2012) 10–26.
- [23] W. Dahmen, C. Plesken, G. Welper, Double greedy algorithms: reduced basis methods for transport dominated problems, *ESAIM: Math. Model. Numer. Anal.* 48 (3) (2014) 623–663.
- [24] Luca Dede, Reduced basis method for parametrized advection–reaction problems, *J. Comput. Math* 28 (1) (2010) 122–148.
- [25] P. Pacciarini, G. Rozza, Stabilized reduced basis method for parametrized advection–diffusion PDEs, *Comput. Methods Appl. Mech. Engrg.* 274 (2014) 1–18.
- [26] W. Aquino, J.C. Brigham, C.J. Earls, N. Sukumar, Generalized finite element method using proper orthogonal decomposition, *Internat. J. Numer. Methods Engrg.* 79 (7) (2009) 887–906.
- [27] Alfonso Caiazzo, Traian Iliescu, Volker John, Svetlana Schyschlowa, A numerical investigation of velocity–pressure reduced order models for incompressible flows, *J. Comput. Phys.* 259 (2014) 598–616.
- [28] K. Kunisch, S. Volkwein, Control of the Burgers equation by a reduced-order approach using proper orthogonal decomposition, *J. Optim. Theory Appl.* 102 (2) (1999) 345–371.
- [29] L. Sirovich, Turbulence and the dynamics of coherent structures. Parts I–III, *Quart. Appl. Math.* 45 (3) (1987) 561–590.
- [30] S. Volkwein, *Model reduction using proper orthogonal decomposition*, in: Lecture Notes, Faculty of Mathematics and Statistics, University of Konstanz, 2011.
- [31] S. Volkwein, *Proper orthogonal decomposition: Theory and reduced-order modelling*, in: Lecture Notes, Department of Mathematics and Statistics, University of Konstanz, 2013.
- [32] Philippe G. Ciarlet, *The finite element method for elliptic problems*, in: Studies in Mathematics and its Applications, vol. 4, North-Holland Publishing Co., Amsterdam–New York–Oxford, 1978.
- [33] Isaac Harari, Thomas J.R. Hughes, What are C and $h^?$: inequalities for the analysis and design of finite element methods, *Comput. Methods Appl. Mech. Engrg.* 97 (2) (1992) 157–192.
- [34] K. Kunisch, S. Volkwein, Galerkin proper orthogonal decomposition methods for parabolic problems, *Numer. Math.* 90 (1) (2001) 117–148.
- [35] T. Iliescu, Z. Wang, Are the snapshot difference quotients needed in the proper orthogonal decomposition?, *SIAM J. Sci. Comput.* 36 (3) (2014) A1221–A1250.
- [36] J.B. Conway, *A Course in Functional Analysis*, Second ed., Springer-Verlag, New York, 1990.
- [37] T. Iliescu, Z. Wang, Variational multiscale proper orthogonal decomposition: Navier-Stokes equations, *Num. Meth. P.D.E.s* 30 (2) (2014) 641–663.
- [38] J.R. Singler, New POD error expressions, error bounds, and asymptotic results for reduced order models of parabolic PDEs, *SIAM J. Numer. Anal.* 52 (2) (2014) 852–876.
- [39] Torsten Linß, Martin Stynes, Numerical methods on Shishkin meshes for linear convection–diffusion problems, *Comput. Methods Appl. Mech. Engrg.* 190 (28) (2001) 3527–3542.
- [40] Yintzer Shih, Jun-Yong Cheng, Kuen-Tsann Chen, An exponential-fitting finite element method for convection–diffusion problems, *Appl. Math. Comput.* 217 (12) (2011) 5798–5809.
- [41] Volker John, Gunar Matthies, MoonMMD—a program package based on mapped finite element methods, *Comput. Vis. Sci.* 6 (2-3) (2004) 163–169.
- [42] J.-L. Guermond, Stabilization of Galerkin approximations of transport equations by subgrid modeling, *M2AN Math. Model. Numer. Anal.* 33 (6) (1999) 1293–1316.