Weierstraß-Institut
für Angewandte Analysis und Stochastik
Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 2198-5855

Numerical study of SUPG and LPS methods combined with higher order variational time discretization schemes applied to time-dependent convection-diffusion-reaction equations

Naveed Ahmed[1], Gunar Matthies[2]

submitted: May 7, 2014

[1] Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
email: Naveed.Ahmed@wias-berlin.de

[2] Universität Kassel, Institut für Mathematik
Heinrich-Plett-Str. 40
34132 Kassel
Germany
email: matthies@mathematik.uni-kassel.de

No. 1948
Berlin 2014

2010 *Mathematics Subject Classification.* 65M12, 65M15, 65M60.

Key words and phrases. stabilized finite elements, discontinuous Galerkin, continuous Galerkin–Petrov, transient convection-diffusion-reaction equations.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 20372-303
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract This paper considers the numerical solution of time-dependent convection-diffusion-reaction equations. We shall employ combinations of streamline-upwind Petrov–Galerkin (SUPG) and local projection stabilization (LPS) meth-ods in space with the higher order variational time discretization schemes. In particular, we consider time discretizations by discontinuous Galerkin (dG) meth-ods and continuous Galerkin–Petrov (cGP) methods. Several numerical tests have been performed to assess the accuracy of combinations of spatial and temporal discretization schemes. Furthermore, the dependence of the results on the stabilization parameters of the spatial discretizations are discussed. In addition, the long-time behavior of overshoots and undershoots is studied. The efficient solution of the obtained systems of linear equations by GMRES methods with multigrid preconditioners will be investigated.

1 Introduction

Let Ω be a polygonal or polyhedral domain in $\mathbb{R}^d$, $d = 2, 3$, with Lipschitz bound-ary $\partial\Omega$. We consider the following time-dependent convection-diffusion-reaction equation:

Find $u : \Omega \times [0,T] \to \mathbb{R}$ such that

$$\begin{aligned} u' - \varepsilon \Delta u + \boldsymbol{b} \cdot \nabla u + \sigma u &= f && \text{in } \Omega \times (0,T), \\ u &= 0 && \text{on } \partial\Omega \times (0,T), \\ u(\cdot,0) &= u_0 && \text{in } \Omega, \end{aligned} \tag{1}$$

with a small positive constant $0 < \varepsilon \ll 1$, $\boldsymbol{b}(x,t)$, $\sigma(x,t)$, and $f(x,t)$ are given functions, u_0 the initial data, and $T > 0$ a given final time. In the following we assume that there exists a positive constant σ_0 such that

$$\sigma(x,t) - \frac{1}{2} \operatorname{div} \boldsymbol{b}(x,t) \geq \sigma_0 > 0 \qquad \forall (x,t) \in \overline{\Omega} \times [0,T] \tag{2}$$

which guarantees the unique solvability of problem (1). Note that the assumption (2) is no restriction for time-dependent convection-diffusion problems. Indeed, if condition (2) is not fulfilled we consider the problem

Find $v : \Omega \times [0,T] \to \mathbb{R}$ such that

$$\begin{aligned} v' - \varepsilon \Delta v + \boldsymbol{b} \cdot \nabla v + (\sigma + M) v &= \mathrm{e}^{-Mt} f && \text{in } \Omega \times (0,T), \\ v &= 0 && \text{on } \partial\Omega \times (0,T), \\ v(\cdot,0) &= u_0 && \text{in } \Omega, \end{aligned}$$

for the function v defined by

$$v(x,t) := \mathrm{e}^{-Mt} u(x,t).$$

If M is chosen sufficiently large then condition (2) is fulfilled for the modified v-problem. Note that the above transformation causes only small changes in all upcoming error estimate since only an additional factor $(M+1)^n$ will appear on the right-hand side where n is the highest order time derivative inside norms of the exact solution.

It is well-known for convection-dominated problems that numerical solutions obtained by standard Galerkin finite element methods are polluted by spurious oscillations which spread over the whole spatial domain unless the mesh width becomes unpractically small. To overcome this instability while ensuring high accuracy, several stabilization techniques have been established in the last decades. One of the famous remedies is the streamline-upwind Petrov–Galerkin (SUPG) method developed by Hughes and Brooks [15]. In [16], a space-time formulation of (1) is studied where the spatial discretization is stabilized by SUPG and a discontinuous Galerkin method is used as temporal discretization. Note that space-time formulations lead to larger systems of linear equations since $(d+1)$-dimensional problems are discretized. In order to handle only d-dimensional problems, we will separate spatial and temporal discretization in this paper.

The success of stabilization methods depends very often crucially on the choice of stabilization parameters. Applying standard techniques to prove stability and error estimates for the residual based stabilization methods for time-dependent convection-diffusion-reaction problems leads to lower bounds on the time step length, see [23]. The SUPG method in space combined with time discretizations by the backward Euler scheme, the Crank-Nicolson method, and the second order BDF scheme for transient transport problems was investigated by Burman [7].

He proved error bounds in the L^2-norm and in a norm of the material derivative provided the problem data are sufficiently smooth. In that study the stabilization parameters depend only on the mesh size in space since the temporal discretization is performed after the choice of spatial stabilization parameters. Semi-discretization in space of (1) without assumption (2) have been considered by Harari and Hauke [13] as well as by Burman and Smith [8].

John and Novo [23] have analyzed for solving (1) the combination of SUPG in space with time stepping by the backward Euler method and the Crank-Nicolson scheme. If the stabilization parameters are defined for the fully discrete scheme then the stabilization parameters depend on the time step length in such a way that they tend to zero as the time step length approaches zero. Two different choices for stabilization parameters which depend on the time step length have been discussed in [23]. For the case that convection and reaction are independent of time, a new technique has been considered in [23] which allows that the stabilization parameters could be chosen similar to the steady-state case. This means, they are independent of the time step length.

Comparisons of the SUPG method with other stabilization techniques can be found in [10, 27]. The stability of consistent stabilization methods for convection-diffusion and flow problems in the small time step limit has been investigated in [5, 14].

A stabilization technique which became very popular during the last decade is the local projection stabilization (LPS) scheme [4, 6, 29]. The local projection stabilization method provides an additional control on the fluctuation of the gradient or parts of its. Although the method is weakly consistent only, the consistency error can be bounded such that the optimal order of convergence is maintained. In contrast to the fully consistent SUPG method, neither time derivatives nor second order derivatives have to be assembled for the stabilization term of LPS. Local projection methods were successfully applied to Stokes problems [3], Oseen problems [6, 29], and convection-diffusion-reaction problems [30].

The application of spatial stabilization techniques reduces dramatically the oscillations which spread in Galerkin methods over the whole computational domain. The remaining oscillations are much smaller and located only in the vicinity of interior and boundary layers. In order to overcome these oscillations, methods for shock or discontinuity capturing can be used. Numerical and analytical investigations can be found e.g. in [19–21].

In order to obtain numerical solutions which are highly accurate in space and time, the use of higher order time discretization schemes is essential. We will apply in this paper higher order time discretization schemes of variational type which will be combined with spatial stabilizations by SUPG and LPS. Continuous Galerkin–Petrov schemes (cGP) and discontinuous Galerkin schemes (dG) are considered. We denote by cGP(k) the method where the discrete solution space in time consists of continuous piece-wise polynomials of degree k and the discrete test space is built by discontinuous polynomials of degree $k-1$. Hence, it can be seen as a Petrov–Galerkin method. Note that this method has been named differently in the literature: discontinuous Galerkin–Petrov method in [34] or continuous Galerkin–Petrov (cGP) method in [17, 18, 28]. We call dG(k) those method where both solution space and test space are constructed by discontinuous polynomials of degree k. Since the time test functions for both considered temporal discretizations

are discontinuous in time, the solutions of these schemes can be calculated by a time marching process.

Spatial discretizations by discontinuous Galerkin methods were introduced by Reed and Hill for neutron transport problems, see [31]. The use of continuous and discontinuous finite element methods to discretize time-dependent problems has been analyzed for ordinary and partial differential equations by several authors. The time discretization by discontinuous Galerkin methods was introduced and analyzed in [11] for the numerical solution of ordinary differential equations and is combined with continuous finite element methods in space for parabolic problem in [12, 35, 36].

The combination of LPS methods in space and dG methods in time has been analyzed for transient convection-diffusion-reaction problems in [1]. The continuous Galerkin method in time for the heat equation has been studied by Aziz and Monk in [2]. They have proved optimal error estimates as well as super-convergence results at the endpoints of the discrete time intervals. Schieweck [34] has investigated the cGP(k)-method for linear ordinary differential equations in an abstract Hilbert space setting and for nonlinear system of ordinary differential equations in d space dimensions. He has proved A-stability and optimal error estimates of the associated cGP(k)-method. Moreover, it was shown that this discretization method has an energy decreasing property for the gradient flow equation of an energy functional. Numerical comparisons of dG and cGP methods as time discretization of heat equations and transient Stokes problems are presented in [17] and [18], respectively. Two families of variational time discretization methods for the numerical solution of nonlinear systems of ordinary differential equations were recently given in [28]. Furthermore, it was shown there that these methods can be interpreted as pure collocation and pure variational methods with special numerical integration. In [28], simple post-processing algorithms were presented which allow to increase the obtained accuracy in time by one order in time-integrated norms.

Multigrid methods are known as the efficient iterative methods for solving the large system of linear equations which arise from the discretization of the partial differential equations. We will apply them to the block systems which are obtained after the discretization in space and time. However, the multigrid methods are not used as solvers themselves but as preconditioners inside flexible GMRES iterations [33]. Due to the natural block structure related to the time discretization, block versions of SOR and SSOR are applied as smoothers inside the multigrid method. The grid transfer operations act separately on each block of unknowns.

The aim of our paper is to study numerically various combinations of spatial and temporal discretizations for solving (1). In particular, we will present numerical results for the combination of streamline-upwind Petrov–Galerkin methods and local projection stabilization methods in space with time discretizations by continuous Galerkin–Petrov methods and discontinuous Galerkin methods. The main focus lies on higher order methods in space and time. In total, four examples with different solution behavior will be considered. We investigate numerically a problem with dominating time error, a problems with exponential boundary layers, and two problems with rotating bodies. We will evaluate the applied schemes by their undershoots and overshoots. Furthermore, the long time behavior will

be examined. In addition, the performance of the flexible GMRES scheme with multigrid preconditioner will be investigated.

The remainder of the paper is organized as follows: Section 2 introduces the basic notation and presents some preliminaries that will be used later. In Section 3, we describe the semi-discretization of the SUPG method and the fully discrete formulations using cGP and dG in time, respectively. Section 4 presents the discretization in space by LPS methods and the corresponding fully discrete formulations obtained by time discretizations with cGP and dG, respectively. Section 5 contains the numerical studies and the evaluation of the methods. Numerical study of the multigrid methods are discussed in Section 6. The numerical studies confirm that the multigrid methods can be regarded as the most efficient iterative solver since the rate of convergence is almost independent of the mesh grid in space and the time step length.

2 Preliminaries

Throughout this paper, standard notation and conventions will be used. For a measurable set $G \subset \mathbb{R}^d$, the inner product in $L^2(G)$ and $L^2(G)^d$ will be denoted by $(\cdot,\cdot)_G$. The norm and semi-norm in $W^{m,p}(G)$ are given by $\|\cdot\|_{m,p,G}$ and $|\cdot|_{m,p,G}$, respectively. In the case $p = 2$, we write $H^m(G)$, $\|\cdot\|_{m,G}$, and $|\cdot|_{m,G}$ instead of $W^{m,2}(G)$, $\|\cdot\|_{m,2,G}$, and $|\cdot|_{m,2,G}$. If $G = \Omega$, the index G in inner products, norms, and semi-norms will be omitted. The subspace of functions from $H^1(\Omega)$ having zero boundary trace is denoted by $H_0^1(\Omega)$. We consider also some Bochner spaces. Let X be a Banach space with norm $\|\cdot\|_X$. We define

$$
\begin{aligned}
L^2(0,T;X) &:= \left\{ v : (0,T) \to X \ : \ \int_0^T \|v(t)\|_X^2 \, dt < \infty \right\}, \\
H^1(0,T;X) &:= \left\{ v \in L^2(0,T;X) \ : \ v' \in L^2(0,T;X) < \infty \right\}, \\
C(0,T;X) &:= \{ v : [0,T] \to X \ : \ v \text{ is continuous} \},
\end{aligned}
$$

where v' is the time derivative of v in the sense of distributions.

In order to write a weak form of (1), let us introduce the space $V := H_0^1(\Omega)$, its dual space $H^{-1}(\Omega)$, and $\langle\cdot,\cdot\rangle$ as the duality product between these spaces. A function u is a weak solution of problem (1), if

$$
u \in L^2\big(0,T;H_0^1(\Omega)\big), \qquad u' \in L^2\big(0,T;H^{-1}(\Omega)\big) \tag{3}
$$

with

$$
\langle u'(t), v\rangle + a(t;u(t),v) = \langle f(t), v\rangle \qquad \forall v \in V \tag{4}
$$

for almost all $t \in (0,T]$ and

$$
u(0) = u_0, \tag{5}
$$

where the form a is given by

$$
a(t;u,v) := \varepsilon(\nabla u, \nabla v) + (\boldsymbol{b}(t)\cdot\nabla u, v) + (\sigma(t)u, v).
$$

The first argument t of a is used to indicate that the coefficients inside a depend on time while a is bilinear with respect to the second and third arguments.

Note that (3) implies the continuity of u as a mapping of $[0,T] \to L^2(\Omega)$ such that the initial condition (5) is well-defined.

In the following, we shall denote by f', f'', and $f^{(k)}$ the first, second, and k-th order time derivative of f, respectively.

For finite element discretizations of (4), let $\{\mathcal{T}_h\}$ denote a family of shape regular triangulations of Ω into open d-simplices, quadrilaterals, or hexahedra such that

$$\overline{\Omega} = \bigcup_{K \in \mathcal{T}_h} \overline{K}.$$

The diameter of $K \in \mathcal{T}_h$ will be denoted by h_K and the mesh size h is defined by $h := \max_{K \in \mathcal{T}_h} h_K$. Let V_h be a finite element space defined on $\mathcal{T}_h$.

The standard Galerkin method applied to (4) consists in

Find $u_h \in H^1(0,T;V_h)$ such that $u_h(0) = u_{h,0}$ and for almost all $t \in (0,T]$

$$\big(u_h'(t), v_h\big) + a\big(t; u_h(t), v_h\big) = \big(f(t), v_h\big) \qquad \forall v_h \in V_h, \tag{6}$$

where $u_{h,0} \in V_h$ is a suitable approximation of u_0.

In the convection-dominant case ($\varepsilon \ll 1$), it is well-known that the standard Galerkin method (6) applied to (1) is unstable and leads to solutions which are globally polluted by spurious oscillations unless the discretization parameter h is sufficiently small.

3 Spatial stabilization by the SUPG method

3.1 The SUPG method

A popular remedy to enhance stability while keeping accuracy of underlying finite element methods is the streamline-upwind Petrov–Galerkin (SUPG) scheme. In the time-continuous case, the stabilized semi-discrete problem is defined as follows:

Find $u_h \in H^1(0,T;V_h)$ such that $u_h(0) = u_{h,0}$ and for almost every $t \in (0,T]$

$$\begin{aligned}\big(u_h'(t), v_h\big) + a_{\mathrm{SUPG}}\big(t; u_h(t), v_h\big) + \sum_{K \in \mathcal{T}_h} \delta_K \big(u_h'(t), \boldsymbol{b}(t) \cdot \nabla v_h\big)_K \\ = \big(f(t), v_h\big) + \sum_{K \in \mathcal{T}_h} \delta_K \big(f(t), \boldsymbol{b}(t) \cdot \nabla v_h\big)_K \qquad \forall v_h \in V_h. \end{aligned} \tag{7}$$

The bilinear form $a_{\mathrm{SUPG}}(t;\cdot,\cdot)$ is given by

$$a_{\mathrm{SUPG}}(t; u_h, v_h) := a(t; u_h, v_h) + \sum_{K \in \mathcal{T}_h} \delta_K \big(-\varepsilon \Delta u_h + \boldsymbol{b}(t) \cdot \nabla u_h + \sigma(t) u_h, \boldsymbol{b}(t) \cdot \nabla v_h\big)_K,$$

where δ_K, $K \in \mathcal{T}_h$, denote the local stabilization parameters which depend on the mesh cells $K \in \mathcal{T}_h$. Note again that the first argument t indicates that the coefficients inside the bilinear form are time-dependent. Let the inverse inequality

$$\|\Delta v_h\|_{0,K} \le c_{\mathrm{inv}} h_K^{-1} |v_h|_{1,K} \qquad \forall v_h \in V_h$$

hold for all $K \in \mathcal{T}_h$ where the constant $c_{\rm inv}$ is independent of K and h. We assume for the stabilization parameter δ_K that

$$0 < \delta_K \le \frac{1}{2} \min \left\{ \frac{\sigma_0}{\|\sigma\|_{0,\infty,K}}, \frac{h_K^2}{\varepsilon c_{\rm inv}} \right\} \qquad \forall K \in \mathcal{T}_h. \tag{8}$$

The mesh-dependent norm associated with $a_{\rm SUPG}$ is given by

$$|||v|||_{\rm SUPG} := \left\{ \varepsilon |v|_1^2 + \sigma_0 \|v\|_0^2 + \sum_{K \in \mathcal{T}_h} \delta_K \|\boldsymbol{b} \cdot \nabla v\|_{0,K}^2 \right\}^{1/2}.$$

If the norm $|||\cdot|||$ will be evaluated for a time-dependent function v at time t then the convection field $\boldsymbol{b}$ will be evaluated at the same time point t.

The steady-state problem associated with (1) is given by

$$\begin{aligned} -\varepsilon \Delta U + \boldsymbol{b} \cdot \nabla U + \sigma U &= f \qquad \text{in } \Omega, \\ U &= 0 \qquad \text{on } \partial\Omega. \end{aligned} \tag{9}$$

Its SUPG discretization reads

Find $U_h \in V_h$ such that

$$a_{\rm SUPG}(U_h, v_h) = (f, v_h) + \sum_{K \in \mathcal{T}_h} (f, \boldsymbol{b} \cdot v_h)_K \qquad \forall v_h \in V_h$$

where the bilinear form $a_{\rm SUPG}$ is independent of time.

With the help of the local mesh Péclet number

$$Pe_K := \frac{\|b\|_{0,\infty,K}\, h_K}{2\varepsilon}, \qquad K \in \mathcal{T}_h,$$

we set δ_K, $K \in \mathcal{T}_h$, according to

$$\delta_K := \begin{cases} \delta_0 h_K & \text{if } Pe_K > 1, \\ \delta_1 h_K^2/\varepsilon & \text{if } Pe_K \le 1, \end{cases} \tag{10}$$

where δ_0 and δ_1 are non-negative constants.

We have the following a priori estimate, see [32, Theorem 3.27].

Theorem 1 *Let the data of* (9) *be sufficiently smooth. The stabilization parameters* δ_K, $K \in \mathcal{T}_h$, *fulfill* (8) *and* (10). *Furthermore,* $\mathbb{P}_r$*-elements on simplices and* $\mathbb{Q}_r$*-elements on quadrilaterals and hexahedra are used. Then, the global error estimate*

$$\|U - U_h\|_{SUPG} \le C(\varepsilon^{1/2} + h^{1/2}) h^r \|U\|_{r+1}$$

holds true where the positive constant C *is independent of* ε *and* h.

In the following, we will discretize problem (7) in time by using continuous Galerkin–Petrov (cGP) and discontinuous Galerkin (dG) methods. To this end, we consider a partition $0 = t_0 < t_1 < \cdots < t_N = T$ of the time interval $I := [0, T]$ and set $I_n := (t_{n-1}, t_n]$, $\tau_n := t_n - t_{n-1}$, $n = 1, \ldots, N$, and $\tau := \max\limits_{1 \le n \le N} \tau_n$. For a given non-negative integer k, we define the fully discrete time-continuous spaces

$$X_k := \left\{ u \in C(I, V_h) : u|_{I_n} \in \mathbb{P}_k(I_n, V_h),\ n = 1, \ldots, N \right\}$$

and time-discontinuous spaces

$$Y_k := \Big\{ v \in L^2(I, V_h) : v|_{I_n} \in \mathbb{P}_k(I_n, V_h),\ n = 1, \ldots, N \Big\}$$

where

$$\mathbb{P}_k(I_n, V_h) := \Big\{ u : I_n \to V_h : u(t) = \sum_{j=0}^{k} U^j t^j,\ U^j \in V_h,\ j = 0, \ldots, k \Big\}$$

denotes the space of V_h-valued polynomials of order k in time on I_n. The functions in the space Y_k are allowed to be discontinuous at the nodes t_n, $n = 1, \ldots, N-1$. For such functions the left-sided value u_n^-, right-sided value u_n^+, and the jump $[u]_n$ are defined by

$$u_n^- := \lim_{t \to t_n - 0} u(t), \qquad u_n^+ := \lim_{t \to t_n + 0} u(t), \qquad [u]_n := u_n^+ - u_n^-.$$

In what follows the SUPG method combined with the cGP method will be called SUPG/cGP while the combination of the SUPG method with the dG method will be called SUPG/dG.

3.2 The method SUPG/cGP(k)

In this section, we describe the combination of the cGP time discretization scheme with the SUPG finite element method in space to get a fully discrete version of (7). It reads

Find $u_{h,\tau} \in X_k$ such that $u_{h,\tau}(0) = u_{h,0}$ and

$$\begin{aligned} &\int_0^T \Big\{ (u'_{h,\tau}(t), v_{h,\tau}(t)) + a_{\mathrm{SUPG}}\big(t; u_{h,\tau}(t), v_{h,\tau}(t)\big) \\ &\qquad\qquad + \sum_{K \in \mathcal{T}_h} \delta_K \big(u'_{h,\tau}(t), \boldsymbol{b}(t) \cdot \nabla v_{h,\tau}(t) \big)_K \Big\} dt \\ &= \int_0^T \Big\{ \big(f(t), v_{h,\tau}(t) \big) + \sum_{K \in \mathcal{T}_h} \delta_K \big(f(t), \boldsymbol{b}(t) \cdot \nabla v_{h,\tau}(t) \big)_K \Big\} dt \qquad \forall v_{h,\tau} \in Y_{k-1}. \end{aligned} \tag{11}$$

Since the test functions are allowed to be discontinuous at the discrete time points t_n, $n = 1, \ldots, N-1$, we can choose test functions $v_{h,\tau}(t) = v_h \psi(t)$ with a time independent $v_h \in V_h$ and a scalar function $\psi : I \to \mathbb{R}$ which is zero on $I \setminus I_n$ and a polynomial of degree less than or equal to $k-1$ on I_n. Then, the solution of the cGP(k)-method can be determined by successively solving a local problem on each time interval I_n.

The fully discrete I_n-problem associated with (11) reads as follows:

Find $u_{h,\tau}\big|_{I_n} \in \mathbb{P}_k(I_n, V_h)$ such that for all

$$\int_{I_n} \left\{ (u'_{h,\tau}(t), v_h) + a_{\mathrm{SUPG}}(t; u_{h,\tau}(t), v_h) + \sum_{K\in\mathcal{T}_h} \delta_K \big(u'_{h,\tau}(t), \boldsymbol{b}(t)\cdot\nabla v_h\big)_K \right\} \psi(t) dt$$
$$= \int_{I_n} \left\{ \big(f(t), v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K \big(f(t), \boldsymbol{b}(t)\cdot\nabla v_h\big)_K \right\} \psi(t) dt \qquad \forall v_h \in V_h, \forall \psi \in \mathbb{P}_{k-1}(I_n)$$

with $u_{h,\tau}\big|_{I_1}(t_0) = u_{h,0}$ and $u_{h,\tau}\big|_{I_n}(t_{n-1}) := u_{h,\tau}\big|_{I_{n-1}}(t_{n-1})$ for $n \geq 2$.

We apply the $(k+1)$-points Gauß-Lobatto quadrature rule for the numerical integration of the time integrals. This formula is exact for polynomials of degree less than or equal to $2k-1$. We denote by $\hat{t}_j$ and $\hat{\omega}_j$, $j = 0, \ldots, k$, the Gauß-Lobatto points and the corresponding quadrature weights on $[-1, 1]$, respectively. Let $\hat{\phi}_j \in \mathbb{P}_k$, $j = 0, \ldots, k$, and $\hat{\psi}_j \in \mathbb{P}_{k-1}$, $j = 1, \ldots, k$, denote the Lagrange basis functions with respect to $\hat{t}_j$, $j = 0, \ldots, k$, and $\hat{t}_j$, $j = 1, \ldots, k$, respectively. The time polynomials $\phi_{n,j} \in \mathbb{P}_k(I_n)$, $j = 0, \ldots, k$, and $\psi_{n,j} \in \mathbb{P}_{k-1}(I_n)$, $j = 1, \ldots, k$, are defined by

$$\phi_{n,j}(t) := \hat{\phi}_j\big(T_n^{-1}(t)\big) \qquad \text{and} \qquad \psi_{n,j}(t) := \hat{\psi}_j\big(T_n^{-1}(t)\big)$$

with the affine transformation

$$T_n : [-1, 1] \to \overline{I}n, \quad \hat{t} \mapsto t_{n-1} + \frac{\tau_n}{2}(\hat{t} + 1), \tag{12}$$

see [28]. In order to determine the local solution $u_{h,\tau}|_{I_n}$, we represent it by

$$u_{h,\tau}\big|_{I_n}(t) = \sum_{j=0}^{k} U_{n,h}^j \phi_{n,j}(t) \quad \forall t \in I_n,$$

with coefficients $U_{n,h}^j \in V_h$, $j = 0, \ldots, k$. The particular ansatz ensures

$$u_{h,\tau}(t_{n,j}) = U_{n,h}^j, \qquad j = 0, \ldots, k,$$

where

$$t_{n,j} := T_n(\hat{t}_j), \qquad j = 0, \ldots, k.$$

Since $t_{n,0} = t_{n-1}$ and $t_{n,k} = t_n$ hold, the initial condition is equivalent to the conditions

$$U_{1,h}^0 = u_{h,0} \qquad \text{and} \qquad U_{n,h}^0 = u_{h,\tau}\big|_{I_n}(t_{n-1}) = U_{n-1,h}^k \quad \text{if } n \geq 2.$$

Using the properties of the basis functions in time, we obtain the following coupled system of equations:

For $U_{1,h}^0 = u_{h,0}$ and $U_{n,h}^0 = U_{n-1,h}^k$ if $n \geq 2$, find the coefficients $U_{n,h}^j \in V_h$, $j = 1, \ldots k$, such that

$$\sum_{j=0}^{k} \alpha_{i,j}^c \left\{ \big(U_{n,h}^j, v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K \big(U_{n,h}^j, \boldsymbol{b}(t_{n,j})\cdot\nabla v_h\big)_K \right\} + \frac{\tau_n}{2} a_{\mathrm{SUPG}}(t_{n,i}; U_{n,h}^i, v_h)$$
$$= \frac{\tau_n}{2} \left\{ \big(f(t_{n,i}), v_h\big) + \beta_i^c \big(f(t_{n-1}), v_h\big) \right\}$$
$$+ \frac{\tau_n}{2} \sum_{K\in\mathcal{T}_h} \delta_K \left\{ \big(f(t_{n,i}), \boldsymbol{b}(t_{n,i})\cdot\nabla v_h\big)_K + \beta_i^c \big(f(t_{n-1}), \boldsymbol{b}(t_{n-1})\cdot\nabla v_h\big)_K \right\}$$

for $i = 1, \dots, k$ and for all $v_h \in V_h$, where $\alpha_{i,j}^c$ and β_i^c are defined by

$$\alpha_{i,j}^c := \hat{\phi}_j'(\hat{t}_i) + \beta_i^c \hat{\phi}_j'(\hat{t}_0), \quad \beta_i^c := \hat{\omega}_0 \hat{\psi}_i(\hat{t}_0), \qquad i = 1, \dots, k,\ j = 0, \dots, k, \tag{13}$$

see [28]. Let $\{b_1, \dots, b_{\mathcal{M}}\}$ be a finite element basis of the space V_h and $\boldsymbol{\xi}_n^j \in \mathbb{R}^{\mathcal{M}}$ denotes the nodal vector associated with the finite element function $U_{n,h}^j \in V_h$, i.e.,

$$U_{n,h}^j(x) = \sum_{\nu=1}^{\mathcal{M}} \big(\boldsymbol{\xi}_n^j\big)_\nu b_\nu(x), \qquad x \in \Omega.$$

Furthermore, we define by

$$\begin{aligned}
(M)_{s,\nu} &:= (b_\nu, b_s), \qquad (C_n^j)_{s,\nu} := \sum_{K \in \mathcal{T}_h} \delta_K \big(b_\nu, \boldsymbol{b}(t_{n,j}) \cdot \nabla b_s\big)_K, \\
(A_n^j)_{s,\nu} &= a_{\mathrm{SUPG}}\big(t_{n,j}, b_\nu, b_s\big), \\
(F_n^j)_\nu &:= \big(f(t_{n,j}), b_\nu\big) + \sum_{K \in \mathcal{T}_h} \delta_K \big(f(t_{n,j}), \boldsymbol{b}(t_{n,j}) \cdot \nabla b_\nu\big)_K
\end{aligned}$$

the mass matrix $M \in \mathbb{R}^{\mathcal{M} \times \mathcal{M}}$, the matrices $C_n^j \in \mathbb{R}^{\mathcal{M} \times \mathcal{M}}$ associated with the additional time derivative term, the stiffness matrices $A_n^j \in \mathbb{R}^{\mathcal{M} \times \mathcal{M}}$, and the discrete source term vectors $F_n^j \in \mathbb{R}^{\mathcal{M}}$. Note that the time-dependence of A_n^j and C_n^j is due to the time-dependent coefficients $\boldsymbol{b}$ and c.

Using the above notation, the numerically integrated fully discrete I_n-problem gives the following $(k \times k)$-block-system:

Find $\boldsymbol{\xi}_n^j \in \mathbb{R}^{\mathcal{M}}$, $j = 1, \dots, k$, such that

$$\sum_{j=0}^{k} \alpha_{i,j}^c \Big\{ M + C_n^j \Big\} \boldsymbol{\xi}_n^j + \frac{\tau_n}{2} A_n^i \boldsymbol{\xi}_n^i = \frac{\tau_n}{2} \Big\{ F_n^i + \beta_i^c \big(F_n^0 - A_n^0 \boldsymbol{\xi}_n^0 \big) \Big\}, \qquad i = 1, \dots, k. \tag{14}$$

Note that $\boldsymbol{\xi}_n^0$ is given either via the discrete initial condition $u_{h,0}$ for $n = 1$ or by $\boldsymbol{\xi}_n^0 = \boldsymbol{\xi}_{n-1}^k$ for $n \geq 2$.

In the following we present the methods SUPG/cGP(1) and SUPG/cGP(2) in more detail.

SUPG/cGP(1) We use the 2-points Gauß-Lobatto formula with points $t_{n,0} = t_{n-1}$, $t_{n,1} = t_n$, and weights $\hat{\omega}_0 = \hat{\omega}_1 = 1$, which yields the well known trapezoidal rule. We get $\alpha_{1,0}^c = -1$, $\alpha_{1,1}^c = 1$, and $\beta_1^c = 1$, see [28], and problem (14) leads to the following equation for the only coefficient $\boldsymbol{\xi}_n^1 \in \mathbb{R}^{\mathcal{M}}$

$$\Big\{ M + C_n^1 + \frac{\tau_n}{2} A_n^1 \Big\} \boldsymbol{\xi}_n^1 = \Big\{ M + C_n^0 - \frac{\tau_n}{2} A_n^0 \Big\} \boldsymbol{\xi}_n^0 + \frac{\tau_n}{2} \Big\{ F_n^0 + F_n^1 \Big\}.$$

SUPG/cGP(2) Here, we apply the 3-points Gauß-Lobatto formula with points $t_{n,0} = t_{n-1}$, $t_{n,1} = (t_{n-1} + t_n)/2$, $t_{n,2} = t_n$, and reference weights $\hat{\omega}_0 = \hat{\omega}_2 = 1/3$, $\hat{\omega}_1 = 4/3$, which is known as Simpson's rule. Then, we obtain the coefficients

$$(\alpha_{i,j}^c) = \begin{pmatrix} -\frac{5}{4} & 1 & \frac{1}{4} \\ 2 & -4 & 2 \end{pmatrix}, \qquad (\beta_i^c) = \begin{pmatrix} \frac{1}{2} \\ -1 \end{pmatrix}. \tag{15}$$

On the time interval $I_n = (t_{n-1}, t_n]$ we have to solve for the vectors $\boldsymbol{\xi}_n^1$ and $\boldsymbol{\xi}_n^2$ which correspond to

$$U_n^1 = u_{h,\tau}(t_{n,1}) = u_{h,\tau}\left(\frac{t_{n-1}+t_n}{2}\right) \qquad \text{and} \qquad U_n^2 = u_{h,\tau}(t_{n,2}) = u_{h,\tau}(t_n).$$

The corresponding coupled (2×2)-block-system for $\boldsymbol{\xi}_n^1,\ \boldsymbol{\xi}_n^2 \in \mathbb{R}^{\mathcal{M}}$ reads:

$$\begin{aligned} &\begin{bmatrix} M + C_n^1 + \frac{\tau_n}{2} A_n^1 & \frac{1}{4}(M + C_n^2) \\ -4(M + C_n^1) & 2\left(M + C_n^2\right) + \frac{\tau_n}{2} A_n^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_n^1 \\ \boldsymbol{\xi}_n^2 \end{bmatrix} \\ &\qquad = \begin{bmatrix} \left(\frac{5}{4}(M + C_n^0) - \frac{\tau_n}{4} A_n^0\right)\boldsymbol{\xi}_n^0 + \frac{\tau_n}{2} F_n^1 + \frac{\tau_n}{4} F_n^0 \\ \left(-2(M + C_n^0) + \frac{\tau_n}{2} A_n^0\right)\boldsymbol{\xi}_n^0 + \frac{\tau_n}{2} F_n^2 - \frac{\tau_n}{2} F_n^0 \end{bmatrix}. \end{aligned} \tag{16}$$

3.3 The method SUPG/dG(k)

In this subsection we describe the details of SUPG/dG(k) which combines the discontinuous Galerkin method dG(k) in time with the SUPG method in space.

The fully discrete method SUPG/dG(k) reads:

Find $u_{h,\tau} \in Y_k$ such that

$$\begin{aligned} &\sum_{n=1}^{N} \int_{I_n} \Big\{ (u'_{h,\tau}(t), v_{h,\tau}(t)) + \sum_{K \in \mathcal{T}_h} \delta_K \big(u'_{h,\tau}(t), \boldsymbol{b}(t) \cdot \nabla v_{h,\tau}(t)\big)_K \\ &+ a_{\mathrm{SUPG}}\big(t; u_{h,\tau}(t), v_{h,\tau}(t)\big) \Big\} dt + \sum_{n=1}^{N-1} \big([u_{h,\tau}]_n, v_n^+\big) + (u_0^+, v_0^+) \\ &\qquad + \sum_{n=1}^{N-1} \sum_{K \in \mathcal{T}_h} \delta_K \big([u_{h,\tau}]_n, \boldsymbol{b}(t_n) \cdot \nabla v_n^+\big)_K + \sum_{K \in \mathcal{T}_h} \delta_K \big(u_0^+, \boldsymbol{b}(t_0) \cdot \nabla v_0^+\big)_K \\ &= (u_{h,0}, v_0^+) + \sum_{K \in \mathcal{T}_h} \delta_K \big(u_{h,0}, \boldsymbol{b}(t_0) \cdot \nabla v_0^+\big)_K \\ &\qquad + \int_0^T \Big\{ (f(t), v_{h,\tau}(t)) + \sum_{K \in \mathcal{T}_h} \delta_K \big(f(t), \boldsymbol{b}(t) \cdot \nabla v_{h,\tau}(t)\big)_K \Big\} dt \end{aligned} \tag{17}$$

for all $v_{h,\tau} \in Y_k$.

Since the test functions in time are discontinuous, we choose $v_{h,\tau}(t) = v_h \psi(t)$ with $v_h \in V_h$ and a scalar function $\psi : I \to \mathbb{R}$ which is zero on $I \setminus I_n$ and a polynomial of degree less than or equal to k on I_n. Hence, the problem (17) can be solved by a time-marching process.

The I_n-problem of the fully discrete method SUPG/dG(k) reads:

Given u_n^- with $u_0^- = u_{h,0}$, find $u_{h,\tau}\big|_{I_n} \in \mathbb{P}_k(I_n, V_h)$ such that for all $\psi \in \mathbb{P}_k(I_n)$

$$\begin{aligned}
&\int_{I_n} \Big\{ \big(u'_{h,\tau}(t), v_h\big) + a_{\mathrm{SUPG}}\big(t; u_{h,\tau}(t), v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K\big(u'_{h,\tau}(t), \boldsymbol{b}(t)\cdot\nabla v_h\big) \Big\} \psi(t) dt \\
&\quad + \Big\{ \big([u_{h,\tau}]_{n-1}, v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K\big([u_{h,\tau}]_{n-1}, \boldsymbol{b}(t_{n-1})\cdot\nabla v_h\big) \Big\} \psi(t_{n-1}) \\
&\qquad = \int_{I_n} \Big\{ \big(f(t), v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K\big(f(t), \boldsymbol{b}(t)\cdot\nabla v_h\big)_K \Big\} \psi(t) dt \qquad \forall v_h \in V_h. \qquad (18)
\end{aligned}$$

In order to evaluate the time integrals numerically, the $(k+1)$-points right-sided Gauß-Radau quadrature formula is applied. Using this quadrature rule, polynomials up to degree $2k$ are integrated exactly. Let $\hat{t}_j$ and $\hat{\omega}_j$, $j = 1, \ldots, k+1$, denote the points and the weights for the $(k+1)$-points right-sided Gauß-Radau quadrature formula on $[-1, 1]$, respectively. In particular, we have $\hat{t}_{k+1} = 1$. The Lagrange basis functions with respect to these points are denoted by $\hat{\phi}_j$, $j = 1, \ldots, k+1$. Following [28], the polynomial functions $\phi_{n,j}(t) \in \mathbb{P}_k(I_n)$ are defined by

$$\phi_{n,j}(t) := \hat{\phi}_j\big(T_n^{-1}(t)\big)$$

with the affine mapping T_n from (12). Since $u_{h,\tau}$ restricted to the interval I_n is a V_h-valued polynomial of degree less than or equal to k, it can be represented as

$$u_{h,\tau}\big|_{I_n}(t) := \sum_{j=1}^{k+1} U_{n,h}^j \phi_{n,j}(t)$$

where $U_{n,h}^j \in V_h$, $j = 1, \ldots, k+1$. Due to this choice of the ansatz basis, we have

$$u_{h,\tau}(t_{n,j}) = U_{n,h}^j, \qquad j = 1, \ldots, k+1,$$

where

$$t_{n,j} := T_n(\hat{t}_j), \qquad j = 1, \ldots, k+1.$$

As particular test functions in time, we choose

$$\psi_{n,j}(t) := \frac{1}{\hat{\omega}_j} \hat{\phi}_j\big(T_n^{-1}(t)\big), \qquad j = 1, \ldots, k+1.$$

Using the above setting in (18), we get the following system of equations:

Find the coefficients $U_{n,h}^j \in V_h$, $j = 1, \ldots k+1$, such that

$$\begin{aligned}
&\sum_{j=1}^{k+1} \alpha_{i,j}^d \Big\{ \big(U_{n,h}^j, v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K\big(U_{n,h}^j, \boldsymbol{b}(t_{n,j})\cdot\nabla v_h\big)_K \Big\} + \frac{\tau_n}{2} a_{\mathrm{SUPG}}\big(t_{n,i}; U_{n,h}^i, v_h\big) \\
&\qquad = \beta_i^d \Big\{ \big(U_{n,h}^0, v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K\big(U_{n,h}^0, \boldsymbol{b}(t_{n-1})\cdot\nabla v_h\big)_K \Big\} \\
&\qquad\quad + \frac{\tau_n}{2} \Big\{ \big(f(t_{n,i}), v_h\big) + \sum_{K\in\mathcal{T}_h} \delta_K\big(f(t_{n,i}), \boldsymbol{b}(t_{n,i})\cdot\nabla v_h\big)_K \Big\}
\end{aligned}$$

for $i = 1, \ldots, k+1$ and for all $v_h \in V_h$, where

$$\alpha_{i,j}^d := \hat{\phi}_j'(\hat{t}_i) + \beta_i^d \hat{\phi}_j(-1), \qquad \beta_i^d := \frac{1}{\hat{\omega}_i}\hat{\phi}_i(-1), \qquad U_{n,h}^0 = U_{n-1,h}^-. \tag{19}$$

Similarly as in the cGP-method, we use a basis $\{b_\mu\}_{1\le\mu\le\mathcal{M}}$ of V_h, the matrices M, C_n^j, A_n^j, and the source term vectors F_n^j. Then, the fully discrete $(k+1)\times(k+1)$-block-system of the I_n-problem reads as follows:

Find $\boldsymbol{\xi}_n^j \in \mathbb{R}^{\mathcal{M}}$, $j = 1, \ldots, k+1$, such that

$$\sum_{j=1}^{k+1} \alpha_{i,j}^d \big\{M + C_n^j\big\}\boldsymbol{\xi}_n^j + \frac{\tau_n}{2} A_n^i \boldsymbol{\xi}_n^i = \beta_i^d \big\{M + C_n^0\big\}\boldsymbol{\xi}_n^0 + \frac{\tau_n}{2} F_n^i, \qquad i = 1, \ldots, k+1, \tag{20}$$

where $\boldsymbol{\xi}_n^j$ denotes the nodal vector of $U_{n,h}^j \in V_h$. The vector $\boldsymbol{\xi}_n^0$ is obtained either from the discrete initial condition $u_{u,0}$ (for $n = 1$) or by $\boldsymbol{\xi}_n^0 := \boldsymbol{\xi}_{n-1}^{k+1}$ (for $n \ge 2$).

In the following, we presents the schemes for the cases $k = 0$ and $k = 1$.

SUPG/dG(0) The 1-point Gauß-Radau formula with point $t_{n,1} = t_n$ and weight $\hat{\omega}_1 = 2$ gives the well-known implicit Euler method, i.e., the I_n-problem is the following one-block equation for $\boldsymbol{\xi}_n^1 \in \mathbb{R}^{\mathcal{M}}$:

$$\big(M + C_n^1 + \tau_n A_n^1\big)\boldsymbol{\xi}_n^1 = (M + C_n^0)\boldsymbol{\xi}_n^0 + \tau_n F_n^1.$$

SUPG/dG(1) The 2-points Gauß-Radau formula with points $t_{n,1} = t_{n-1} + \tau_n/3$, $t_{n,2} = t_n$ and weights $\hat{\omega}_1 = 3/2$, $\hat{\omega}_2 = 1/2$ yields on the time interval I_n the following coupled (2×2)-block-system for $\boldsymbol{\xi}_n^1, \boldsymbol{\xi}_n^2 \in \mathbb{R}^{\mathcal{M}}$:

$$\begin{bmatrix} \frac{3}{4}(M + C_n^1) + \frac{\tau_n}{2}A_n^1 & \frac{1}{4}(M + C_n^2) \\ -\frac{9}{4}(M + C_n^1) & \frac{5}{4}(M + C_n^2) + \frac{\tau_n}{2}A_n^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_n^1 \\ \boldsymbol{\xi}_n^2 \end{bmatrix} = \begin{bmatrix} (M + C_n^0)\boldsymbol{\xi}_n^0 + \frac{\tau_n}{2}F_n^1 \\ -(M + C_n^0)\boldsymbol{\xi}_n^0 + \frac{\tau_n}{2}F_n^2 \end{bmatrix}.$$

4 Spatial stabilization by the LPS method

In this section we consider the local projection stabilization (LPS) method. Compared to the SUPG method the LPS method is weakly consistent only. However, its application does not required the computation of the second order derivatives and time derivatives. Hence, it can be easily applied to time-dependent problems. Furthermore, it is possible to relax the strong coupling between various components of the solution in the SUPG stabilization.

We concentrate on the one-level local projection stabilization method in which approximation and projection spaces are defined on the same mesh. Let $D(K)$, $K \in \mathcal{T}_h$, be finite dimensional spaces and $\pi_K : L^2(K) \to D(K)$ the local L^2 projection into $D(K)$. The local fluctuation operator $\kappa_K : L^2(K) \to L^2(K)$ is given by $\kappa_K v := v - \pi_K v$. The stabilization term S_h is defined by

$$S_h(u_h, v_h) := \sum_{K\in\mathcal{T}_h} \mu_K \big(\kappa_K \nabla u_h, \kappa_K \nabla v_h\big)_K \tag{21}$$

where μ_K, $K \in \mathcal{T}_h$, are user chosen non-negative constants. The local projection stabilization gives additional control on the fluctuation of the gradient.

The stabilized semi-discrete problem for u_h reads:

Find $u_h \in H^1(0,T;V_h)$ such that $u_h(0) = u_{h,0}$ and for almost all $t \in (0,T]$

$$\big(u_h'(t), v_h\big) + a_{\mathrm{LPS}}\big(t; u_h(t), v_h\big) = \big(f, v_h\big) \qquad \forall v_h \in V_h \tag{22}$$

where the stabilized bilinear form $a_{LPS}(t;\cdot,\cdot)$ is given by

$$a_{\mathrm{LPS}}(t; u_h, v_h) := a(t; u_h, v_h) + S_h(u_h, v_h).$$

The mesh-dependent norm associated with a_{LPS} is given by

$$|||v|||_{\mathrm{LPS}} := \left\{ \varepsilon |v|_1^2 + \sigma_0 \|v\|_0^2 + \sum_{K \in \mathcal{T}_h} \mu_K \|\kappa_h \nabla v\|_{0,K}^2 \right\}^{1/2}.$$

The LPS discretization of the steady-state convection-diffusion problem (9) reads

Find $U_h \in V_h$ such that

$$a_{\mathrm{LPS}}(U_h, v_h) = (f, v_h) \qquad \forall v_h \in V_h$$

where a_{LPS} is independent of time.

In order to get suitable results, the approximation space V_h and and local projection spaces $D(K)$, $K \in \mathcal{T}_h$, have to fulfill certain compatibility conditions, see [29]. Compared to standard finite element spaces, the approximation spaces are locally enriched by bubble functions.

The following a priori error estimate for the steady-state convection-diffusion problem (9) is well known, see [32, Theorem 3.74].

Theorem 2 *Let the data of* (9) *be sufficiently smooth. We choose the stabilization parameters $\mu_K = \mu_0 h_K$, $K \in \mathcal{T}_h$, with a positive constant μ_0. Furthermore, we use enriched elements of order r and projection spaces $D(K) = \mathbb{P}_{r-1}(K)$. Then, there exists a positive constant C independent of h and ε such that the error estimate*

$$\|U - U_h\|_{LPS} \leq C(\varepsilon^{1/2} + h^{1/2}) h^r \|U\|_{r+1}$$

holds true.

The application of cGP and dG time discretizations to spatial discretizations stabilized by the local projection method follows the same lines as for SUPG/cGP and SUPG/dG methods. Hence, we will concentrate on the main differences.

The matrices A_n^j are now defined using the local projection bilinear form a_{LPS}, i.e.,

$$(A_n^j)_{s,\nu} := a_{\mathrm{LPS}}(t_{n,j}; b_\nu, b_s),$$

where the time-dependence of A_n^j comes from the time-dependent coefficients $\boldsymbol{b}$ and c.

Since there is no coupling term between the time derivative and the derivative in streamline direction, the matrices C_n^j are no longer present. This simplifies the resulting discrete systems and has a positive influence on the assembling time. Let again $\boldsymbol{\xi}_n^j$ denote the nodal vector of $U_{n,h}^j \in V_h$.

Finally we present the fully discrete systems for LPS/cGP and LPS/dG.

The LPS/cGP(k) methods reads:

Find $\boldsymbol{\xi}_n^j \in \mathbb{R}^{\mathcal{M}}$, $j = 1, \ldots, k$, such that

$$\sum_{j=0}^{k} \alpha_{i,j}^c M \boldsymbol{\xi}_n^j + \frac{\tau_n}{2} A_n^i \boldsymbol{\xi}_n^i = \frac{\tau_n}{2} \Big\{ F_n^i + \beta_i^c \big(F_n^0 - A_n^0 \boldsymbol{\xi}_n^0 \big) \Big\}, \qquad i = 1, \ldots, k.$$

Note that $\boldsymbol{\xi}_n^0$ is given either via the initial condition $u_{h,0}$ for $n = 0$ or by $\boldsymbol{\xi}_n^0 = \boldsymbol{\xi}_{n-1}^k$ for $n \geq 2$.

The LPS/dG(k) methods is given by

Find $\boldsymbol{\xi}_n^j \in \mathbb{R}^{\mathcal{M}}$, $j = 1, \ldots, k+1$, such that

$$\sum_{j=1}^{k+1} \alpha_{i,j}^d M \boldsymbol{\xi}_n^j + \frac{\tau_n}{2} A_n^i \boldsymbol{\xi}_n^i = \beta_i^d M \boldsymbol{\xi}_n^0 + \frac{\tau_n}{2} F_n^i, \qquad i = 1, \ldots, k+1. \tag{23}$$

The nodal vector $\boldsymbol{\xi}_n^0$ is obtained either from the initial condition $u_{h,0}$ for $n = 0$ or by $\boldsymbol{\xi}_n^0 = \boldsymbol{\xi}_{n-1}^{k+1}$ for $n \geq 2$.

5 Numerical tests

In this section we present some numerical experiments to assess accuracy and performance of combinations of spatial stabilization techniques with higher order variational time discretization schemes. All computations were performed with the finite element code MooNMD [22].

In our numerical calculations for the SUPG stabilization, we applied $\mathbb{P}_r$-elements on triangles and mapped $\mathbb{Q}_r$-elements on quadrilaterals, see [9]. For the LPS stabilization, the local projection space $D(K) = \mathbb{P}_{r-1}(K)$ together with locally enriched approximation spaces were used on both triangles and quadrilaterals. On triangular cells, the local approximation space is given by

$$\mathbb{P}_r^{\text{bubble}}(K) := \mathbb{P}_r(K) + b_{\triangle,K} \cdot \mathbb{P}_{r-1}(K)$$

where $b_{\triangle,K}$ denotes a cubic bubble function on K which vanishes on ∂K. On quadrilaterals, mapped finite element spaces [9] were used where the enriched spaces on the reference cell $\widehat{K} = (-1, 1)^2$ are defined by

$$\mathbb{Q}_r^{\text{bubble}}(\widehat{K}) := \mathbb{Q}_r(\widehat{K}) + \text{span}\big\{ \hat{b}_{\square} \hat{x}_i^{r-1},\ i = 1, 2 \big\}$$

with the biquadratic bubble function $\hat{b}_{\square} = (1 - \hat{x}_1^2)(1 - \hat{x}_2^2)$ on the reference square $\widehat{K}$.

We will use

$$\|v\|_{\text{cGP}} := \left\{ \int_0^T \Big(\|v'(t)\|_0^2 + \|v(t)\|_{\text{S}}^2 \Big) dt \right\}^{1/2},$$

$$\|v\|_{\text{dG}} := \left\{ \int_0^T \|v(t)\|_{\text{S}}^2 \, dt + \frac{1}{2} \sum_{n=1}^{N-1} \|[v]_n\|_0^2 + \frac{1}{2} \|v_0^+\|_0^2 + \frac{1}{2} \|v_N^-\|_0^2 \right\}^{1/2}$$

as the norms associated with the cGP and the dG methods, respectively, where $\|\cdot\|_{\mathrm{S}}$ corresponds to the spatial stabilization, i.e., $\|\cdot\|_{\mathrm{S}} = |||\cdot|||_{\mathrm{SUPG}}$ or $\|\cdot\|_{\mathrm{S}} = |||\cdot|||_{\mathrm{LPS}}$. Furthermore, errors in the norm

$$\|v\|_{L^2(L^2)} := \left\{ \int_0^T \|v(t)\|_0^2 \, dt \right\}^{1/2}$$

will be considered.

Combining the known error estimates for temporal discretizations of ordinary differential equations [2, 28, 37] with the established error estimates for stabilized spatial discretizations on sequences of successively refined meshes [32], we expect the following error estimates.

Theorem 3 *Let u be the solution of problem (1). Denote by $u_{h,\tau}$ either the solution of the fully discrete cGP(k) or the solution of the fully discrete dG(k) method. Let the data of problem (1) be sufficiently smooth. For the SUPG stabilization in space, we use element of order r and parameters δ_K according to (8) and (10). If LPS is used, we choose enriched elements of order r and $D(K) = \mathbb{P}_{r-1}(K)$ as local projection space. The LPS parameters μ_K are chosen as $\mu_K = \mu_0 h_K$, $K \in \mathcal{T}_h$, with a positive constant μ_0. Then, there exists a positive constant C independent of ε, h, and τ such that we have*

$$\begin{aligned} \|u - u_{h,\tau}\|_{cGP} &\leq C\Big[(\varepsilon^{1/2} + h^{1/2})h^r + \tau^k\Big], \\ \|u - u_{h,\tau}\|_{L^2(L^2)} &\leq C\big(h^{r+1} + \tau^{k+1}\big) \end{aligned}$$

for the cGP(k) method and

$$\begin{aligned} \|u - u_{h,\tau}\|_{dG} &\leq C\Big[(\varepsilon^{1/2} + h^{1/2})h^r + \tau^{k+1/2}\Big], \\ \|u - u_{h,\tau}\|_{L^2(L^2)} &\leq C\big(h^{r+1} + \tau^{k+1}\big) \end{aligned}$$

for the dG(k) method. The constant C depends in both cases on u.

The discrete ℓ^∞-norm is defined by

$$\|v\|_\infty := \max_{1 \leq n \leq N} \|v(t_n)\|_0.$$

It is known from [2] and [37] that the methods cGP(k) and dG(k) are superconvergent of order $2k$ and $2k+1$ at the discrete time points t_n, $n = 1, \ldots, N$, respectively. Combining this with estimates for the spatial error, we expect the following estimate for the error in the ℓ^∞-norm.

Theorem 4 *Let u be the solution of problem (1). Denote by $u_{h,\tau}$ either the solution of the fully discrete cGP(k) or the solution of the fully discrete dG(k) method. If SUPG is applied, elements of order r and stabilization parameters δ_K according to (8) and (10) were used. In case of LPS, enriched element of order r, $D(K) = \mathbb{P}_{r-1}(K)$, $\mu_K = \mu_0 \, h_K$ with $\mu_0 > 0$ were used. Then, the error estimate*

$$\|u - u_{h,\tau}\|_\infty \leq C\big(\tau^{2k} + h^{r+1}\big)$$

holds true for the cGP(k) method while we have the error estimate

$$\|u - u_{h,\tau}\|_\infty \le C\big(\tau^{2k+1} + h^{r+1}\big)$$

for the dG(k) method where the constant C in both cases depends on u, but is independent of ε, h, and τ.

Following [28], a simple post-processing of the time-discrete solutions $u_{h,\tau}$ allows to obtain numerical approximations which are in the integral-based norms $\|\cdot\|_{\mathrm{dG}}$, $\|\cdot\|_{\mathrm{cGP}}$, and $\|\cdot\|_{L^2(L^2)}$ one order better.

First we describe the post-processing for time discretizations by cGP(k). Let $u_{h,\tau}$ denote the solution of SUPG/cGP(k) or LPS/cGP(k), respectively. The post-processed solution $\Pi u_{h,\tau}$ on the time interval I_n is given by

$$(\Pi u_{h,\tau})(t) = u_{h,\tau}(t) + a_n \zeta_n(t), \qquad t \in I_n,$$

where

$$\zeta_n(t) = \frac{\tau_n}{2}\hat{\zeta}(\hat{t}), \qquad \hat{t} := T_n^{-1}(t),$$

with T_n from (12). The polynomial $\hat{\zeta} \in \mathbb{P}_{k+1}$ vanishes in all Gauß-Lobatto points $\hat{t}_j$, $j = 0, \dots, k$, and is scaled such that $\hat{\zeta}'(1) = 1$. The nodal vector $\boldsymbol{\gamma}_n$ of the finite element function $a_n \in V_h$ is the solution of

$$M\boldsymbol{\gamma}_n = F_n^k - A_n^k \boldsymbol{\xi}_n^k - M\boldsymbol{\eta}_n^k$$

where $\boldsymbol{\eta}_n^k$ denotes the nodal representation of $u'_{h,\tau}(t_n) \in V_h$. In the case that SUPG is used as spatial discretization, the matrix M has to be replaced at both occurrences by $M + C_n^k$.

The post-processing of time discretizations by dG(k) is even simpler. The post-processed solution $\Pi u_{h,\tau}$ of the solution $u_{h,\tau}$ of SUPG/dG(k) or LPS/dG(k) on the interval I_n can be represented as

$$(\Pi u_{h,\tau})(t) = u_{h,\tau}(t) + b_n \vartheta_n(t), \qquad t \in I_n,$$

where

$$\vartheta_n(t) = \frac{\tau_n}{2}\hat{\vartheta}(\hat{t}), \qquad \hat{t} := T_n^{-1}(t),$$

with T_n from (12). The polynomial $\hat{\vartheta} \in \mathbb{P}_{k+1}$ is uniquely defined by $\hat{\vartheta}(\hat{t}_j) = 0$ for all Gauß-Radau points $\hat{t}_j$, $j = 1, \dots, k+1$, and $\hat{\vartheta}'(1) = 1$. The finite element function $b_n \in V_h$ is obtained by

$$b_n := \frac{1}{\vartheta_n(t_{n-1})}\big(u_{n-1}^- - u_{n-1}^+\big),$$

i.e., it is just the scaled difference between the initial condition u_{n-1}^- at $t = t_{n-1}$ and the calculated solution u_{n-1}^+ at the same place $t = t_{n-1}$.

The expected error estimates for the post-processed solution $\Pi u_{h,\tau}$ are collected in the following theorem.

Theorem 5 *Let u be the solution of the problem (1) and $\Pi u_{h,\tau}$ the post-process solution of a time discretization by cGP(k) or dG(k). For the SUPG stabilization in space, we use element of order r and parameters δ_K according to (8) and (10). If LPS is used, we choose enriched elements of order r and $D(K) = \mathbb{P}_{r-1}(K)$ as local projection space. The LPS parameters μ_K are chosen as $\mu_K = \mu_0 h_K$ with a positive constant μ_0. Then, there exists a positive constant C depending on u, but independent of ε, h, and τ, such that the error estimates*

$$\begin{aligned}\|u - \Pi u_{h,\tau}\|_{cGP} &\leq C\Big[(\varepsilon^{1/2} + h^{1/2})h^r + \tau^{k+1}\Big],\\ \|u - \Pi u_{h,\tau}\|_{L^2(L^2)} &\leq C\big(h^{r+1} + \tau^{k+2}\big)\end{aligned}$$

for the post-processed cGP(k) method and

$$\begin{aligned}\|u - \Pi u_{h,\tau}\|_{dG} &\leq C\Big[(\varepsilon^{1/2} + h^{1/2})h^r + \tau^{k+2}\Big],\\ \|u - \Pi u_{h,\tau}\|_{L^2(L^2)} &\leq C\big(h^{r+1} + \tau^{k+2}\big)\end{aligned}$$

for the post-processed dG(k) method hold true.

5.1 An example with dominating time error

To assess the effect of the time discretization, we exclude the spatial error in the this example. We consider problem (1) on $\Omega = (0,1)^2$ with $\varepsilon = 10^{-8}$, $\boldsymbol{b} = (1,2)$, $\sigma = 1$, and $T = 1$. The right-hand side f and the initial condition u_0 are chosen such that

$$u(t,x,y) = x(1-x)y(1-y)\sin(50t)$$

is the solution of (1) equipped with homogeneous Dirichlet boundary conditions.

We consider third order elements on a mesh consisting of 16×16 squares. This means that $\mathbb{Q}_3$ elements are used for SUPG while $\mathbb{Q}_3^{\text{bubble}}$ elements with projection onto $\mathbb{P}_2(K)$ are taken for LPS. Note that for any time t the solution u can be represented exactly by a function from the finite element space V_h. Hence, all occurring errors will result from the temporal discretization. The higher order time discretization methods cGP(k) and dG(k) with $k = 2$ and $k = 3$ are applied.

We report in Tables 1–8 the errors and convergence orders for the methods cGP(k) and dG(k), $k = 2, 3$, in combination with spatial stabilizations by SUPG and LPS, respectively. We see that both cGP(k) and dG(k) are accurate of order $k+1$ in the $L^2(L^2)$-norm while the orders k and $k+1/2$ are obtained in cGP-norm and the dG-norm, respectively. These results are in agreement with Theorem 3.

It can be seen from the Tables 3, 4 and 7, 8 that, as predicted by Theorem 5, the post-processing allows to get solutions which provide the convergence order $k+2$ in the $L^2(L^2)$-norm while the convergence orders $k+1$ and $k+2$ in the cGP-norm and dG-norm are obtained, respectively.

Furthermore, comparing the numerical errors in Tables 1, 2, and 5, 6, we note that both spatial stabilization techniques perform quite similar. No essential difference can be seen. Comparing the values in the discrete ℓ^∞-norms, it becomes

Table 1 Example 5.1: errors and convergence orders for SUPG/cGP(2) and LPS/cGP(2)-method.

	$\|u-u_{h,\tau}\|_{L^2(L^2)}$		$\|u-u_{h,\tau}\|_{\infty}$		$\|u-u_{h,\tau}\|_{\mathrm{cGP}}$	
	SUPG	LPS	SUPG	LPS	SUPG	LPS
τ	error	error	error	error	error	error
1/80	1.028-3	1.029-3	3.832-5	4.346-5	5.193-1	5.193-1
1/160	1.281-4	1.281-4	2.182-6	2.584-6	1.322-1	1.322-1
1/320	1.603-5	1.605-5	1.340-7	1.608-7	3.320-2	3.320-2
1/640	2.004-6	2.004-6	8.344-9	1.005-8	8.309-3	8.309-3
1/1280	2.505-7	2.505-7	5.210-10	6.280-10	2.078-3	2.078-3
1/2560	3.131-8	3.131-8	3.256-11	3.925-11	5.195-4	5.195-4
1/5120	3.914-9	3.914-9	2.035-12	2.453-12	1.299-4	1.299-4
order	3	3	4	4	2	2
theory	3	3	4	4	2	2

Table 2 Example 5.1: errors and convergence orders for SUPG/cGP(3) and LPS/cGP(3).

	$\|u-u_{h,\tau}\|_{L^2(L^2)}$		$\|u-u_{h,\tau}\|_{\infty}$		$\|u-u_{h,\tau}\|_{\mathrm{cGP}}$	
	SUPG	LPS	SUPG	LPS	SUPG	LPS
τ	error	error	error	error	error	error
1/40	1.742-3	1.743-3	9.195-5	9.904-5	6.190-1	6.190-1
1/80	1.138-4	1.138-4	5.873-7	7.528-7	8.547-2	8.547-2
1/160	7.229-6	7.229-6	9.002-9	1.217-8	1.095-2	1.095-2
1/320	4.537-7	4.537-7	1.427-10	1.931-10	1.377-3	1.377-3
1/640	2.839-8	2.839-8	2.248-12	3.030-12	1.723-4	1.723-4
1/1280	1.775-9	1.775-9	3.522-14	4.736-14	2.155-5	2.155-5
order	4	4	6	6	3	3
theory	4	4	6	6	3	3

Table 3 Example 5.1: post-processed errors and convergence orders for SUPG/cGP(2) and LPS/cGP(2)-method.

	$\|u-\Pi u_{h,\tau}\|_{L^2(L^2)}$		$\|u-\Pi u_{h,\tau}\|_{\mathrm{cGP}}$	
	SUPG	LPS	SUPG	LPS
τ	error	error	error	error
1/80	4.866-4	4.869-4	1.528-1	1.528-1
1/160	3.036-5	3.038-5	1.988-2	1.988-2
1/320	1.898-6	1.900-6	2.510-3	2.510-3
1/640	1.186-7	1.187-7	3.145-4	3.145-4
1/1280	7.415-9	7.422-9	3.934-5	3.934-5
1/2560	4.635-10	4.639-10	4.918-6	4.918-6
1/5120	2.897-11	2.899-11	6.148-7	6.148-7
order	4	4	3	3
theory	4	4	3	3

obvious that the cGP(k)-methods, $k = 2, 3$, are super-convergent of order 4 and 6 while the dG(k)-methods, $k = 2, 3$, are super-convergent of order 5 and 7. This confirms the theory given in Theorem 4.

Table 4 Example 5.1: post-processed errors and convergence orders for SUPG/cGP(3) and LPS/cGP(3)-method.

	$\|u - \Pi u_{h,\tau}\|_{L^2(L^2)}$		$\|u - \Pi u_{h,\tau}\|_{\mathrm{cGP}}$	
	SUPG	LPS	SUPG	LPS
τ	error	error	error	error
1/40	1.020-3	1.020-3	2.471-1	2.471-1
1/80	3.357-5	3.357-5	1.752-2	1.752-2
1/160	1.069-6	1.069-6	1.130-3	1.130-3
1/320	3.357-8	3.357-8	7.118-5	7.118-5
1/640	1.050-9	1.050-9	4.458-6	4.458-6
1/1280	3.283-11	3.283-11	2.787-7	2.787-7
1/2560	1.026-12	1.026-12	1.742-8	1.742-8
1/5120	3.230-14	3.231-14	1.089-9	1.089-9
order	5	5	4	4
theory	5	5	4	4

Table 5 Example 5.1: errors and convergence orders for SUPG/dG(2) and LPS/dG(2).

	$\|u - u_{h,\tau}\|_{L^2(L^2)}$		$\|u - u_{h,\tau}\|_{\infty}$		$\|u - u_{h,\tau}\|_{\mathrm{dG}}$	
	SUPG	LPS	SUPG	LPS	SUPG	LPS
τ	error	error	error	error	error	error
1/40	6.412-3	6.414-3	8.729-4	8.784-4	8.543-2	8.526-2
1/80	8.456-4	8.456-4	1.599-5	1.627-5	1.781-2	1.780-2
1/160	1.080-4	1.080-4	4.552-7	4.710-7	3.294-3	3.293-3
1/320	1.358-5	1.358-5	1.392-8	1.446-8	5.875-4	5.875-4
1/640	1.700-6	1.700-6	4.327-10	4.501-10	1.040-4	1.040-4
1/1280	2.125-7	2.125-7	1.350-11	1.405-11	1.838-5	1.838-5
1/2560	2.657-8	2.657-8	4.226-13	4.397-13	3.248-6	3.248-6
1/5120	3.321-9	3.321-9	1.335-14	1.401-14	5.741-7	5.741-7
order	3	3	5	5	2.5	2.5
theory	3	3	5	5	2.5	2.5

Table 6 Example 5.1: errors and convergence orders for SUPG/dG(3) and LPS/dG(3).

	$\|u - u_{h,\tau}\|_{L^2(L^2)}$		$\|u - u_{h,\tau}\|_{\infty}$		$\|u - u_{h,\tau}\|_{\mathrm{dG}}$	
	SUPG	LPS	SUPG	LPS	SUPG	LPS
τ	error	error	error	error	error	error
1/40	1.479-3	1.479-3	2.870-5	3.015-5	2.507-2	2.505-2
1/80	1.017-4	1.017-4	1.264-7	1.413-7	2.544-3	2.543-3
1/160	6.514-6	6.514-6	1.140-9	1.042-9	2.310-4	2.309-4
1/320	4.097-7	4.097-7	1.187-11	8.350-12	2.056-5	2.056-5
1/640	2.564-8	2.564-8	1.151-13	6.621-14	1.821-6	1.820-6
1/1280	1.603-9	1.603-9	1.087-15	7.630-16	1.610-7	1.610-7
order	4	4	7	7	3.5	3.5
theory	4	4	7	7	3.5	3.5

5.2 A boundary layer problem

We will study the influence of the stabilization parameters on the behavior of the solution of a problem where the solution exhibits exponential boundary layers. To this end, we consider problem (1) on $\Omega = (0,1)^2$ with $\varepsilon = 10^{-8}$, $\boldsymbol{b} = (3 - tx, 4 - t^2 y)^T$, $\sigma = 1$, and $T = 1$. The right-hand side f, the initial data u_0, and the

Table 7 Example 5.1: post-processed errors and convergence orders for SUPG/dG(2) and LPS/dG(2)-method.

	$\|u - \Pi u_{h,\tau}\|_{L^2(L^2)}$		$\|u - \Pi u_{h,\tau}\|_{\mathrm{dG}}$	
	SUPG	LPS	SUPG	LPS
τ	error	error	error	error
1/20	3.895-2	3.926-2	5.676-2	4.664-2
1/40	3.605-3	3.609-3	5.165-3	3.668-3
1/80	2.283-4	2.284-4	2.892-4	2.287-4
1/160	1.446-5	1.447-5	1.810-5	1.447-5
1/320	9.074-7	9.074-7	1.132-6	9.075-7
1/640	5.677-8	5.677-8	7.076-8	5.677-8
1/1280	3.549-9	3.549-9	4.423-9	3.549-9
1/2560	2.218-10	2.218-10	2.764-10	2.218-10
1/5120	1.387-11	1.387-11	1.728-11	1.387-11
order	4	4	4	4
theory	4	4	4	4

Table 8 Example 5.1: post-processed errors and convergence orders for SUPG/dG(3) and LPS/dG(3)-method.

	$\|u - \Pi u_{h,\tau}\|_{L^2(L^2)}$		$\|u - \Pi u_{h,\tau}\|_{\mathrm{cGP}}$	
	SUPG	LPS	SUPG	LPS
τ	error	error	error	error
1/20	1.272-2	1.274-2	1.806-2	1.310-2
1/40	5.604-4	5.605-4	7.284-4	5.617-4
1/80	1.909-5	1.909-5	2.390-5	1.909-5
1/160	6.101-7	6.101-7	7.609-7	6.101-7
1/320	1.918-8	1.918-8	2.390-8	1.918-8
1/640	6.001-10	6.001-10	7.477-10	6.001-10
1/1280	1.876-11	1.876-11	2.338-11	1.876-11
1/2560	5.863-13	5.863-13	7.306-13	5.864-13
1/5120	1.871-14	1.872-14	2.422-14	1.886-14
order	5	5	5	5
theory	5	5	5	5

Dirichlet boundary conditions are chosen such that

$$u(t,x,y) = \sin(x)\Big(1 - e^{-(2+\cos 2t)(1-x)/\varepsilon}\Big)\, \sin(2y)\Big(1 - e^{-(3+\sin t)(1-y)/\varepsilon}\Big)$$

is the solution of (1). Note that u possesses exponential boundary layers at $x = 1$ and $y = 1$.

We restrict our study to the methods cGP(2) and dG(1) in time, regular meshes of squares, and $\mathbb{Q}_2$-elements for SUPG and $\mathbb{Q}_2^{\mathrm{bubble}}$-elements with projection onto $\mathbb{P}_1(K)$ for LPS.

To see the influence of the stabilization parameters, we set $\delta_K = \delta_0 h_K$ for SUPG and $\mu_k = \mu_0 h_K$ for LPS. The positive constants δ_0 and μ_0 are varied across the wide range from 10^{-6} to 10^6.

The computations were carried out on the refinement levels 4–7 (corresponding to 16×16 up to 128×128 squares) and with a time step length $\tau = 1/160$. All errors were computed on the subdomain $\widetilde{\Omega} := (0, 3/4) \times (0, 3/4)$ which excludes all

layers. Let

$$\|v\|_1 := \left\{ \int_0^T \|v(t)\|_{0,\widetilde{\Omega}}^2 \, dt \right\}^{1/2} \quad \text{and} \quad \|v\|_2 := \left\{ \int_0^T \|v(t)\|_{S,\widetilde{\Omega}}^2 \, dt \right\}^{1/2}$$

denote two norms where all included spatial integrals are restricted to $\widetilde{\Omega}$.

First we consider the cGP(2) method with both spatial stabilization techniques. Figure 1 plots the errors for SUPG/cGP(2) in the norms $\|\cdot\|_1$ and $\|\cdot\|_2$ versus the constant δ_0 inside the definition of the stabilization parameters. The corresponding results for LPS/cGP(2) with respect to the constant μ_0 are shown in Figure 2. Too small constants δ_0 and μ_0 result in large errors since the spatial discretization is under-stabilized. But also too large constants produce large errors due to over-stabilization. Comparing the results of cGP(2) combined with LPS and SUPG, we see that the optimal constant μ_0 is independent of the refinement level while the optimal constant δ_0 decreases from one mesh to the next finer one.

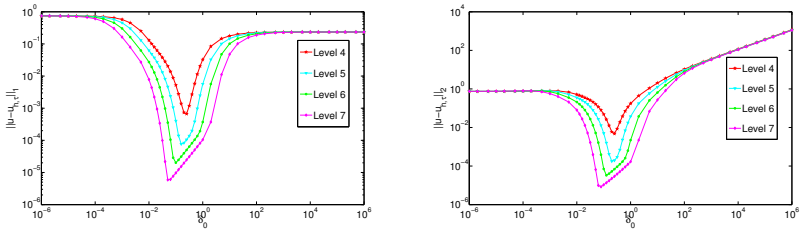

Fig. 1 Example 5.2: errors in the norms $\|\cdot\|_1$ (left) and $\|\cdot\|_2$ (right) versus the constant δ_0 for SUPG/cGP(2) on refinement levels 4–7.

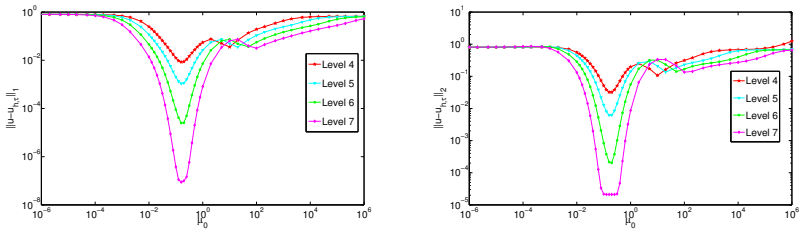

Fig. 2 Example 5.2: errors in the norms $\|\cdot\|_1$ (left) and $\|\cdot\|_2$ (right) versus the constant μ_0 for LPS/cGP(2) on refinement levels 4–7.

We consider now the influence of the constants δ_0 and μ_0 on the errors of the methods SUPG/dG(1) and LPS/dG(1). The corresponding errors in the norms $\|\cdot\|_1$ and $\|\cdot\|_2$ versus δ_0 and μ_0 are shown in Figures 3 and 4. The principal behavior of the dG(1) discretization in time coincides with the results of the cGP(2) method. Too small and too large constants provide large errors due to under- and over-stabilization. Also for the dG(1) method in time, the optimal constant δ_0 for SUPG depends on the mesh while the optimal value for μ_0 is mesh-independent.

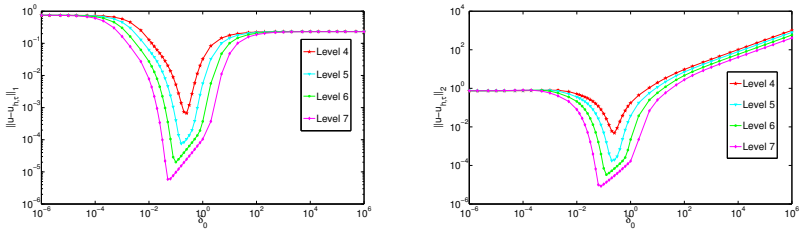

Fig. 3 Example 5.2: errors in the norms $\|\cdot\|_1$ (left) and $\|\cdot\|_2$ (right) versus the constant δ_0 for SUPG/dG(1) on refinement levels 4–7.

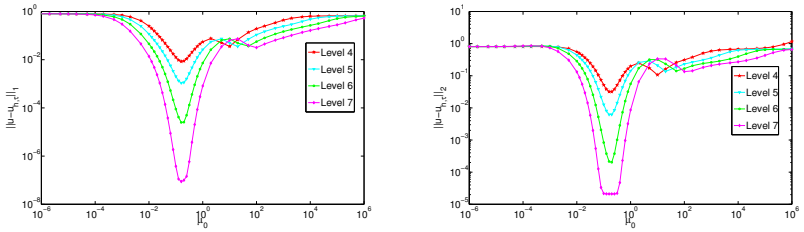

Fig. 4 Example 5.2: errors in the norms $\|\cdot\|_1$ (left) and $\|\cdot\|_2$ (right) versus the constant μ_0 for LPS/cGP(2) on refinement levels 4–7.

Comparing in Figures 1 and 3 the results for SUPG and both considered time discretizations in the norms $\|\cdot\|_1$ and $\|\cdot\|_2$, we observe that the $\|\cdot\|_1$-norm remains bounded even for very large values for the constant δ_0 while the $\|\cdot\|_2$ has no upper bound. The reason is that $\|\cdot\|_2$ includes a term which increases with increasing constant δ_0 inside the stabilization term. For the LPS stabilization, the behavior is different, see Figures 2 and 4. It seems that the error in both norms is bounded even if the constant μ_0 in the stabilization parameter $\mu_K = \mu_0 h_K$ is very large. This indicates that the over-stabilization by large μ_0 leads to solutions with almost vanishing fluctuations. Hence, the constant μ_0 has less influence on the norm.

The pictures in Figure 5 show the computed solution at the final time $T = 1$ for different time discretization methods and different spatial discretizations with local projection stabilization. We clearly observe that the numerical solution obtained with a suitably chosen stabilization (middle column) shows neither oscillations nor smearing. For under-stabilization (left column), the numerical solutions contain large oscillations all over the spatial domain. In the case of over-stabilization (right column), the behavior of LPS and SUPG are different. The LPS solution has oscillations which are concentrated near the exponential boundary layer. In contrast, the SUPG solution is smeared but shows much less oscillations.

5.3 Body rotating problem

In this section two problems with rotating bodies will be considered. The first one is given in $\Omega = (0,1)^2$ with $\varepsilon = 10^{-20}$, $\boldsymbol{b} = (0.5 - y, x - 0.5)^T$, $\sigma = 0$, $f = 0$. The initial condition consists of three disjoint bodies: a slit cylinder, a cone, and a

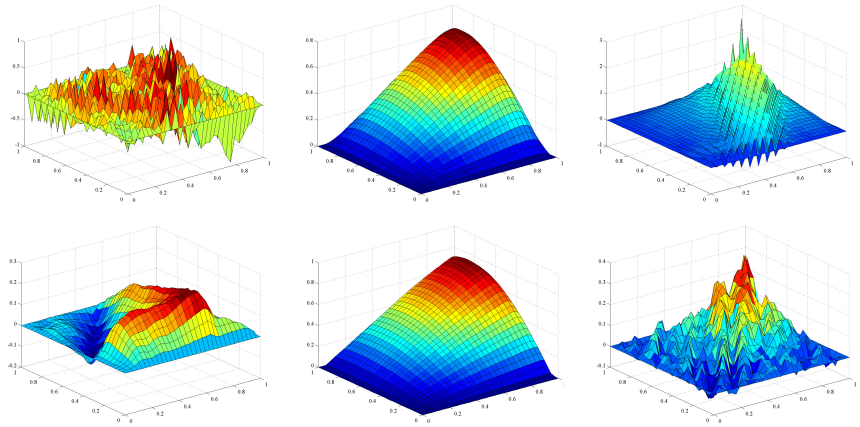

Fig. 5 Example 5.2, Computed solution at final time $T = 1$, time step length $\tau = 1/160$, LPS/cGP(2) with $\mathbb{Q}_2^{\text{bubble}}$ and projection onto $\mathbb{P}_1(K)$ (upper row), SUPG/cGP(2) with $\mathbb{Q}_2$ (bottom row); $\mu_0(\delta_0) = 10^{-4}$ (left), $\mu_0(\delta_0) = 0.1$ (middle), $\mu_0(\delta_0) = 100$ (right).

smooth hump, see left picture of Figure 6. This problem has already been studied numerically for finite element discretizations, see e.g. [1,23,25]. The second problem

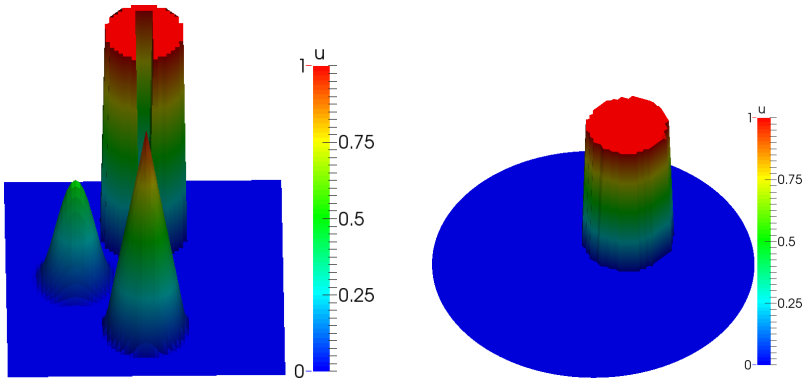

Fig. 6 Example 5.3: initial condition; left: problem in unit square with three bodies, right: problem in unit circle with one body.

is given in the unit circle $\Omega = \{(x_1, x_2) \in \mathbb{R}^2 \ : \ x_1^2 + x_2^2 < 1\}$ with $\varepsilon = 10^{-20}$, $\boldsymbol{b} = (-y, x)^T$, $\sigma = 0$, $f = 0$, and the initial condition

$$u_0(x, y) = \frac{1}{2}\left\{\tanh\left(1000e^{-10[(x-0.3)^2+(y-0.3)^2]}\right) + 1\right\}$$

which is shown in the right picture of Figure 6. A transient transport with this data was considered in [7].

In both examples, the rotation is counter-clockwise and the first revolution completes at $t = 2\pi$. Due to the very small diffusion coefficient $\varepsilon = 10^{-20}$, the solution after complete revolutions is essentially the same as the initial condition.

The aim of these examples is to study the effect of different time discretization schemes in combination with stabilized finite element methods in space. In particular, we use the methods cGP(2) and dG(1) in time. Both SUPG and LPS are used as spatial discretization.

We choose for the problem in the unit square a uniform grid consisting of 64×64 squares. This leads, included all Dirichlet nodes, to 98,817 degrees of freedoms for the second order LPS discretization with $\mathbb{Q}_2^{\text{bubble}}$-elements and projection onto $\mathbb{P}_1(K)$ and 66,049 degrees of freedom for the second order SUPG discretization with $\mathbb{Q}_2$-elements.

For the problem in the unit circle, triangular meshes were used. The coarsest mesh contains 8 congruent triangles which are obtained by connecting the origin with 8 equidistant points on the circle line. The calculations were made on refinement level 5 where the curved boundary was taken into account during refinement. Including the Dirichlet nodes on the boundary, there were 16,641 degrees of freedom for the second order SUPG discretization with $\mathbb{P}_2$-elements and 41,217 degrees of freedom the second order LPS discretization with $\mathbb{P}_2^{\text{bubble}}$-elements and projection onto $\mathbb{P}_1(K)$.

For both problems, we used $\delta_K = 0.25 h_K$ inside the SUPG stabilization while we set $\mu_K = 0.1 h_K$ for LPS. The time step length was fixed to $\tau = 10^{-3}$.

As in [23], we use

$$\text{var}(t) = \max_{(x,y)\in\Omega_h} u_h(t;x,y) - \min_{(x,y)\in\Omega_h} u_h(t;x,y)$$

as measure for under- and overshoots. For calculating the minimal and maximal values of u_h we used only the values at the vertices of the underlying mesh. Note that the optimal value of $\text{var}(t)$ equals to 1 for all t.

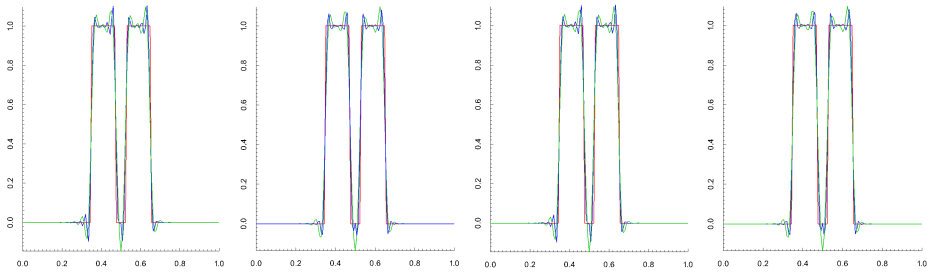

Fig. 7 Example 5.3: cut along the line $y = 0.75$ through the midpoint of the slit cylinder; red: initial condition, blue: solution after one complete revolution ($t = 2\pi$), green: solution after ten complete revolutions ($t = 20\pi$); from left: SUPG/dG(1), LPS/dG(1), SUPG/cGP(2), LPS/cGP(2).

In order to illustrate the solution behavior, we present in Figure 7 for the problem in the unit square with the three rotating bodies and the combinations SUPG/dG(1), LPS/dG(1), SUPG/cGP(2), and LPS/cGP(2) cuts along the line $y = 0.75$ through the midpoint of the slit cylinder after one (in blue) and ten (in green) complete revolutions together with the initial condition (in red). It is clearly to see that over- and undershoots occurs near the edges of the cylinder.

Both the differences between the two considered spatial discretizations and the differences between the two applied temporal discretizations are quite small. The over- and undershoots are of comparable size. The smooth hump was reproduced by all combinations very well. No undershoots are present near the cone, its height is slightly decreasing with time, and the initially sharp tip becomes rounded.

Figure 8 shows for the problem in the circular domain the solution after ten complete revolutions. It can be seen that all combinations of spatial stabilizations and temporal discretizations lead to smeared numerical solutions. Using the same stabilization method in space, the differences between the time discretizations are quite small. However, the choice of the spatial stabilization method has a much larger influence on the solution properties. In this example, the local projection stabilization produces a larger smearing compared to SUPG.

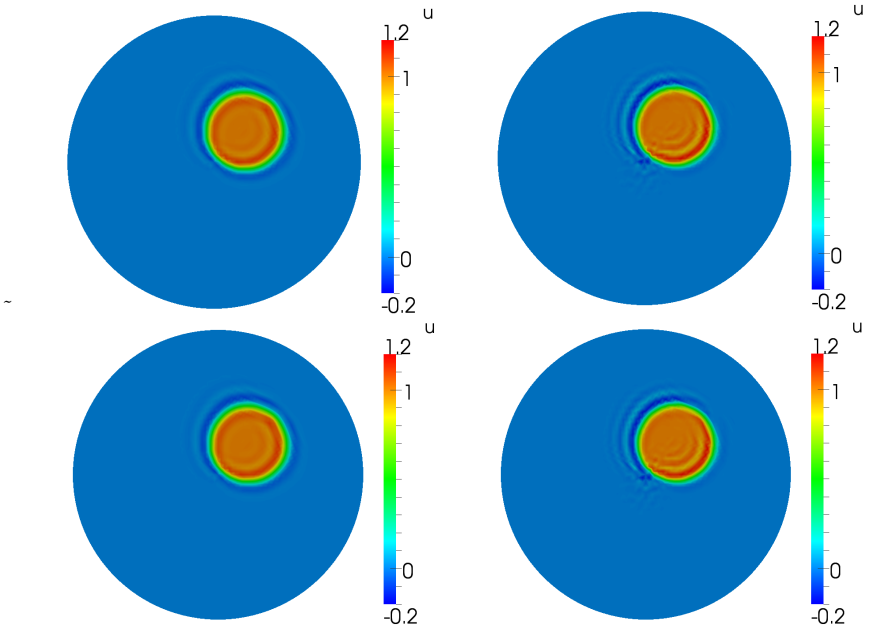

Fig. 8 Example 5.3: computed solution with second order elements at $t = 20\pi$, upper left: SUPG/dG(1), bottom left: SUPG/cGP(2), upper right: LPS/dG(1), bottom right: LPS/cGP(2).

To check the long time behavior of the spurious oscillations measured by $\mathrm{var}(t)$, we made computations for both problem till $T = 20\pi$ which corresponds to ten complete revolutions. The results are only slightly influenced by the applied temporal discretization. We present in Figure 9 the results for SUPG/dG(1) and LPS/dG(1) only. After an initial phase, the quantity $\mathrm{var}(t)$ shows a periodic behavior. It is interesting to observe that LPS is superior to SUPG for the problems in the unit square with three rotating bodies while the problem in the unit circle with one rotating body shows the opposite behavior. We observe for both problems that $\mathrm{var}(t)$ shows for LPS much larger oscillations than for SUPG.

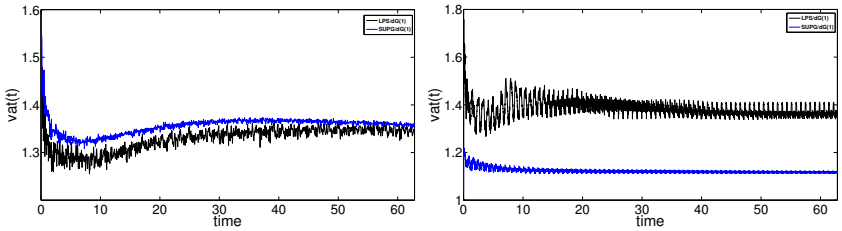

Fig. 9 Example 5.3: long time behavior of the measure var(t) for under- and overshoots, computed with second order elements till $T = 20\pi$, left: problem in unit square with three rotating bodies, right: problem in unit circle with one rotating body.

Compared to the unstabilized case, the remaining oscillations are much smaller and located near sharp layers only. To remove the remaining oscillations, shock or discontinuity capturing methods could be applied. For an overview, we refer to [19–21] and the references therein.

5.4 Transport of a species through a three-dimensional domain

This three-dimensional example models a typical situation which is encountered in applications, see [26]. For a given domain $\Omega = (0,1)^3$, a species enters the domain at the inlet and is transported through the domain to an outlet. In addition, the species is diffused somewhat and in the subregion where the species is transported, also a reaction occurs. The convection field points from the center of the inlet to the center of the outlet and it is not parallel to the coordinate axis.

The inlet is located at $\{0\} \times (5/8, 6/8) \times (5/8, 6/8)$ while the position of the outlet is $\{1\} \times (3/8, 4/8) \times (4/8, 5/8)$. The convection field is defined by $\mathbf{b} = (1, -1/4, -1/8)^T$, the diffusion parameter by $\varepsilon = 10^{-6}$, and the reaction by

$$\sigma(\mathbf{x}) = \begin{cases} 1 & \text{if } \operatorname{dist}(\mathbf{x}, g) \leq 0.1, \\ 0 & \text{else,} \end{cases}$$

where g is the line through the center of the inlet and the center of the outlet and $\operatorname{dist}(\mathbf{x}, g)$ denotes the shortest Euclidean distance of the point $\mathbf{x}$ to the line g. The given ratio of diffusion and convection is typical for many application. The boundary conditions at the inlet is prescribed by

$$u_{\text{in}} = \begin{cases} \sin(\pi t/2) & \text{if } t \in [0,1], \\ 1 & \text{if } t \in (1,2], \\ \sin(\pi(t-1)/2) & \text{if } t \in (2,3]. \end{cases}$$

Homogeneous Neumann boundary conditions are set at the outlet and homogeneous Dirichlet conditions at the rest of the boundary. There are no sources, i.e., $f = 0$. The initial condition is set to be $u_0(x) = 0$. In the time interval $(0,1)$, the inflow is increasing and the injected species is transported towards the outlet. Then, in the time interval $(1,2)$, there is a constant inflow and the species reaches the outlet. At the end of this time interval, there is a almost steady-state solution. Finally, the inflow decreases in the time interval $(2,3)$.

The simulations were performed on an equidistant hexahedral grid with $32 \times 32 \times 32$ cells, leading to 274,625 degrees of freedom (including Dirichlet nodes) and $\tau = 10^{-3}$. Cut planes of the solution at time $t = 2$ computed with dG(1), dG(2), cGP(2) and cGP(3) are given in Figure 10. These cut planes contain the line g between the center of the inlet and the center of the outlet. The numerical results show the large amount of spurious oscillations in the solutions computed with the SUPG method. Figure 10 demonstrates that the solutions are globally polluted with spurious oscillations. However, accurate results can be obtained by using different space discretization schemes, e.g., FEM-FCT, ENO, and WENO schemes, see [24]. Furthermore, there is almost no difference in the solutions computed with different time discretization schemes. As a measure of accuracy, the value of the solution (amount of species) at the center of the outlet was proposed in [26]. It can be observed in Figure 11 that all simulations gave very similar results.

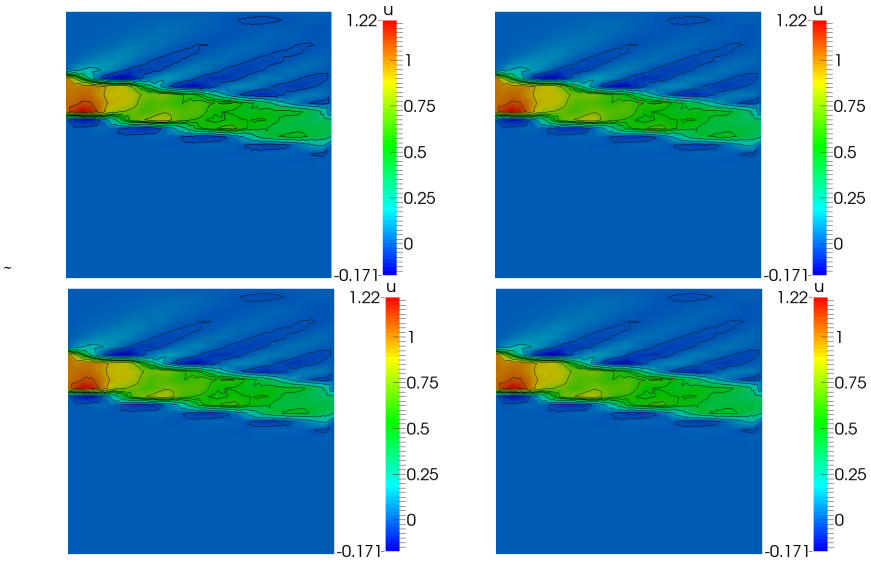

Fig. 10 Example 5.4: Transport of species through a three-dimensional domain, cGP(2), cGP(3) dG(1) and dG(2); from left to right, top to bottom.

6 Multigrid comparison

Section 3 showed that the temporal discretization with cGP($k+1$) and dG(k) leads in each time step to a $(k+1) \times (k+1)$ block system of linear equations. In order to solve it, a flexible GMRES method [33] with a multigrid method as preconditioner is applied. The grid transfer operations are applied component wise to $\boldsymbol{\xi}_n^j$ and the corresponding residual vectors. Block version of SOR and SSOR are used as smoothers within the multigrid. The blocks are built by all components $(\boldsymbol{\xi}_n^j)_i$ which belong to the same spatial degree of freedom i.

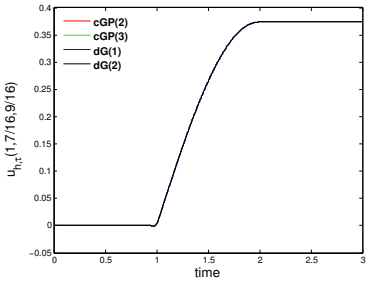

Fig. 11 Example 5.4: Transport of species through a three-dimensional domain; temporal development of the numerical solutions at the center of the outlet.

We present in this section some performance results of the multigrid methods applied to the different time discretization schemes. If not stated otherwise the $W(2,2)$ multigrid cycles with SOR smoother were applied. The parameter ω inside SOR and SSOR was set to 1 in all calculations. Since the performance of the multigrid method applied to the different stabilization techniques was quite similar, we will present results for the SUPG method only. In general, all given results are obtained by averaging over all time steps. Exception are given explicitly.

Table 9 present for Example 5.3 and for the different time discretization schemes the averaged number of multigrid cycle to achieve a Euclidean norm of the residual less than 10^{-10}. We clearly see that the number of needed multigrid cycles is independent of the number of used level, the time discretization scheme, and the time steps length. Hence, the multigrid method provides an optimal solver for the obtained block system.

Table 9 Example 5.3: Number of iteration per multigrid cycle on different numbers of multigrid levels for different time stepping schemes.

	dG(1), dG(2), dG(3)			
# Levels	$\tau = 2\pi/160$	$\tau = 2\pi/320$	$\tau = 2\pi/640$	$\tau = 2\pi/1280$
4	5,5,5	5,5,6	5,5,5	5,5,5
5	4,5,5	4,5,5	5,5,5	5,5,5
6	5,5,5	4,4,5	4,5,5	4,5,5
7	5,4,5	4,4,5	4,4,5	4,4,5
	cGP(2), cGP(3), cGP(4)			
4	5,5,6	5,6,6	5,6,6	5,5,5
5	4,5,5	5,5,5	5,5,5	5,5,5
6	4,4,5	4,5,5	4,5,5	5,5,5
7	5,5,5	4,4,5	4,4,5	4,5,5

For Example 5.3, the averaged computing times for solving the block systems per time step are shown in Table 10 (6 levels inside the multigrid) and Table 11 (7 levels inside the multigrid). The computing times for block systems of the same size, for instance dG(k) and cGP($k-1$), are very similar. Furthermore, the time per time step increases by a factor of 4 from 6 to 7 levels inside the multigrid. Hence, the computing time per step is proportional to the number of unknowns. This shows again the optimal performance of the multigrid.

Table 10 Example 5.3: Computing time per time step using 6 multigrid levels.

# time steps	cGP(2)	cGP(3)	cGP(4)	dG(1)	dG(2)	dG(3)
320	0.39	0.82	1.23	0.38	0.62	1.26
640	0.37	0.81	1.29	0.39	0.78	1.25
1280	0.47	0.79	1.28	0.39	0.81	1.27
2560	0.47	0.82	1.26	0.39	0.78	1.28
5120	0.47	0.81	1.27	0.39	0.64	1.05

Table 11 Example 5.3: Computing time per time step using 7 multigrid levels.

# time steps	cGP(2)	cGP(3)	cGP(4)	dG(1)	dG(2)	dG(3)
320	1.63	2.57	5.04	1.93	2.53	4.93
640	1.54	2.53	5.08	1.54	2.57	5.01
1280	1.56	3.17	4.95	1.53	2.60	4.96
2560	1.57	3.25	4.92	1.53	3.16	4.89
5120	1.54	3.17	4.92	1.54	2.53	3.99

The convergence rate per multigrid cycle in the course of time for Example 5.3 and Example 5.4 are presented in Figure 12. For Example 5.3, we clearly see that the convergence rate stays in a small range. The behavior for Example 5.4 is different. In an initial phase, the rates are changing only slightly. However, the convergence rate drops down around $t \approx 2$ since the solution becomes almost constant. Afterwards, the convergence rate increases since the solution starts to change again due to the changing boundary data. For Example 5.3, the differences between the different time discretization schemes are quite small. The deviation for Example 5.4 are a little bit large. Currently, we don't have an explanation of the different behavior of dG(1) and cGP(2) on the one hand and dG(2) and cGP(3) on the other hand.

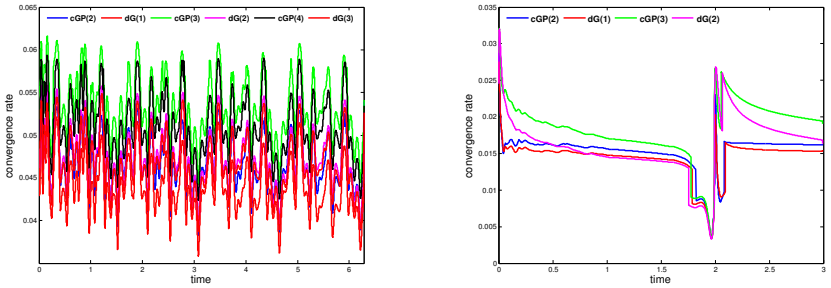

Fig. 12 Convergence rates for multigrid solver on refinement level 6 for Example 5.3 (left) and Example 5.4 (right).

Figure 13 shows the dependence of the averaged convergence rate on the number of pre-smoothing and post-smoothing steps for different time discretization schemes applied to Example 5.3. We observe that the behavior is very similar for all considered schemes. Furthermore, the convergence decreases dramatically with the number of used smoothing step. This is due to the fact that the block system

is dominated by the mass matrix which is much better conditioned as the stiffness matrix which is scaled by the time step length.

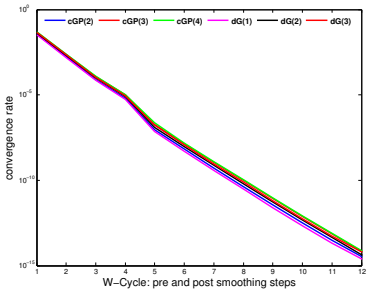

Fig. 13 Example 5.3: Mean convergence rate for changing numbers of pre-smoothing and post-smoothing steps on 5 multigrid levels for different time stepping schemes.

Table 12 presents the averaged number of multigrid cycles per time step, the mean computing time per multigrid cycle, the averaged time per time step, and the mean convergence rates for the smoothers SOR and SSOR applied to Example 5.1 on 6 multigrid levels and a time step length $\tau = 10^{-3}$. One clearly sees that the additional effort of SSOR results in larger times per multigrid cycle but less numbers of cycles and better convergence rate. In total, the SSOR smoother allows to obtain the required accuracy within a smaller computing time.

Table 12 Example 5.1: Number of iteration per multigrid cycle, corresponding time per cycle, time per time step, and convergence rates for smoothers SOR and SSOR on 6 multigrid levels for different time stepping schemes.

	iter		time/cycle		time/time step		convergence rate	
	SOR	SSOR	SOR	SSOR	SOR	SSOR	SOR	SSOR
cGP(2)	5	3	0.260	0.414	1.299	1.243	5.177e-02	2.780e-03
cGP(3)	5	3	0.547	0.809	2.733	2.426	5.551e-02	3.549e-03
cGP(4)	4	3	0.869	1.352	3.479	4.057	4.032e-02	3.424e-03
dG(1)	4	2	0.265	0.421	1.062	0.843	4.596e-02	2.074e-03
dG(2)	4	2	0.607	0.802	2.027	1.605	4.648e-02	2.338e-03
dG(3)	4	2	0.900	1.321	3.602	2.642	4.600e-02	2.269e-03

Acknowledgements The authors like to thank the German research foundation (DFG) for supporting the research under the grant MA 4713/2-1.

References

1. N. Ahmed, G. Matthies, L. Tobiska, and H. Xie. Discontinuous galerkin time stepping with local projection stabilization for transient convection-diffusion-reaction problems. *Comput. Methods Appl. Mech. Engrg.*, 200:1747–1756, 2011.
2. A. K. Aziz and P. Monk. Continuous finite elements in space and time for the heat equation. *Math. Comp.*, 52(186):255–274, 1989.

3. R. Becker and M. Braack. A finite element pressure gradient stabilization for the Stokes equations based on local projections. *Calcolo*, 38(4):173–199, 2001.
4. R. Becker and M. Braack. A two-level stabilization scheme for the Navier–Stokes equations. In M. Feistauer, V. Dolejší, P. Knobloch, and K. Najzar, editors, *Numerical mathematics and advanced applications*, pages 123–130, Berlin, 2004. Springer-Verlag.
5. P. B. Bochev, M. D. Gunzburger, and J. N. Shadid. Stability of the SUPG finite element method for transient advection-diffusion problems. *Comput. Methods Appl. Mech. Engrg.*, 193(23-26):2301–2323, 2004.
6. M. Braack and E. Burman. Local projection stabilization for the Oseen problem and its interpretation as a variational multiscale method. *SIAM J. Numer. Anal.*, 43(6):2544–2566, 2006.
7. E. Burman. Consistent SUPG-method for transient transport problems: stability and convergence. *Comput. Methods Appl. Mech. Engrg.*, 199(17-20):1114–1123, 2010.
8. E. Burman and G. Smith. Analysis of the space semi-discretized SUPG method for transient convection-diffusion equations. *Math. Models Methods Appl. Sci.*, 21(10):2049–2068, 2011.
9. P. G. Ciarlet. *The finite element method for elliptic problems*. North-Holland Publishing Co., Amsterdam, 1978. Studies in Mathematics and its Applications, Vol. 4.
10. R. Codina. Comparison of some finite element methods for solving the diffusion-convection-reaction equation. *Comput. Methods Appl. Mech. Engrg.*, 156(1-4):185–210, 1998.
11. K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational differential equations*. Cambridge University Press, Cambridge, 1996.
12. K. Eriksson, C. Johnson, and V. Thomée. Time discretization of parabolic problems by the discontinuous Galerkin method. *RAIRO Modél. Math. Anal. Numér.*, 19(4):611–643, 1985.
13. I. Harari and G. Hauke. Semidiscrete formulations for transient transport at small time steps. *Internat. J. Numer. Methods Fluids*, 54(6-8):731–743, 2007.
14. M.-C. Hsu, Y. Bazilevs, V. M. Calo, T. E. Tezduyar, and T. J. R. Hughes. Improving stability of stabilized and multiscale formulations in flow simulations at small time steps. *Comput. Methods Appl. Mech. Engrg.*, 199(13-16):828–840, 2010.
15. T. J. R. Hughes and A. N. 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)*, volume 34 of *AMD*, pages 19–35. Amer. Soc. Mech. Engrs. (ASME), New York, 1979.
16. T. J. R. Hughes, L. P. Franca, and M. Mallet. A new finite element formulation for computational fluid dynamics. VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multidimensional advective-diffusive systems. *Comput. Methods Appl. Mech. Engrg.*, 63(1):97–112, 1987.
17. S. Hussain, F. Schieweck, and S. Turek. Higher order Galerkin time discretizations and fast multigrid solvers for the heat equation. *J. Numer. Math.*, 19(1):41–61, 2011.
18. S. Hussain, F. Schieweck, and S. Turek. A note on accurate and efficient higher order Galerkin time stepping schemes for the nonstationary Stokes equations. *Open Numer. Methods J.*, 4:35–45, 2012.
19. V. John and P. Knobloch. On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations. I. A review. *Comput. Methods Appl. Mech. Engrg.*, 196(17-20):2197–2215, 2007.
20. V. John and P. Knobloch. On the performance of SOLD methods for convection-diffusion problems with interior layers. *Int. J. Comput. Sci. Math.*, 1(2-4):245–258, 2007.
21. V. John and P. Knobloch. On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations. II. Analysis for P_1 and Q_1 finite elements. *Comput. Methods Appl. Mech. Engrg.*, 197(21-24):1997–2014, 2008.
22. V. John and G. Matthies. MooNMD—a program package based on mapped finite element methods. *Comput. Vis. Sci.*, 6(2-3):163–169, 2004.
23. V. John and J. Novo. Error analysis of the SUPG finite element discretization of evolutionary convection-diffusion-reaction equations. *SIAM J. Numer. Anal.*, 49(3):1149–1176, 2011.
24. V. John and J. Novo. On (essentially) non-oscillatory discretizations of evolutionary convection-diffusion equations. *J. Comput. Phys.*, 231(4):1570–1586, 2012.
25. V. John and E. Schmeyer. Finite element methods for time-dependent convection-diffusion-reaction equations with small diffusion. *Comput. Methods Appl. Mech. Engrg.*, 198(3-4):475–494, 2008.

26. V. John and E. Schmeyer. On finite element methods for 3D time-dependent convection-diffusion-reaction equations with small diffusion. In A. Hegarty, N. Kopteva, E. O' Riordan, and M. Stynes, editors, *BAIL 2008 - Boundary and Interior Layers*, volume 69 of *Lecture Notes in Computational Science and Engineering*, pages 173–181. Springer, 2009.
27. G. Lube and D. Weiss. Stabilized finite element methods for singularly perturbed parabolic problems. *Appl. Numer. Math.*, 17(4):431–459, 1995.
28. G. Matthies and F. Schieweck. Higher order variational time discretizations for nonlinear systems of ordinary differential equations. Preprint 23/2011, Fakultät für Mathematik, Otto-von-Guericke-Universität Magdeburg, 2011.
29. G. Matthies, P. Skrzypacz, and L. Tobiska. A unified convergence analysis for local projection stabilisations applied to the Oseen problem. *M2AN Math. Model. Numer. Anal.*, 41(4):713–742, 2007.
30. G. Matthies, P. Skrzypacz, and L. Tobiska. Stabilization of local projection type applied to convection-diffusion problems with mixed boundary conditions. *Electron. Trans. Numer. Anal.*, 32:90–105, 2008.
31. W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
32. H.-G. Roos, M. Stynes, and L. Tobiska. *Robust numerical methods for singularly perturbed differential equations*, volume 24 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2008.
33. Y. Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, 1993.
34. F. Schieweck. *A*-stable discontinuous Galerkin-Petrov time discretization of higher order. *J. Numer. Math.*, 18(1):25–57, 2010.
35. D. Schötzau and C. Schwab. An *hp* a priori error analysis of the DG time-stepping method for initial value problems. *Calcolo*, 37(4):207–232, 2000.
36. D. Schötzau and C. Schwab. *hp*-discontinuous Galerkin time-stepping for parabolic problems. *C. R. Acad. Sci. Paris Sér. I Math.*, 333(12):1121–1126, 2001.
37. V. Thomée. *Galerkin finite element methods for parabolic problems*, volume 25 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006.