Comput. Methods Appl. Mech. Engrg. 199 (2010) 582–597

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

A hybridizable discontinuous Galerkin method for Stokes flow

N.C. Nguyen [a,*], J. Peraire [a], B. Cockburn [b]

[a] Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
[b] School of Mathematics, University of Minnesota, Minneapolis, MN 55455, USA

ARTICLE INFO

Article history:
Received 21 February 2009
Received in revised form 28 July 2009
Accepted 19 October 2009
Available online 1 November 2009

Keywords:
Finite element methods
Discontinuous Galerkin methods
Hybrid/mixed methods
Augmented Lagrangian
Stokes flow

ABSTRACT

In this paper, we introduce a hybridizable discontinuous Galerkin method for Stokes flow. The method is devised by using the discontinuous Galerkin methodology to discretize a velocity–pressure–gradient formulation of the Stokes system with appropriate choices of the numerical fluxes and by applying a hybridization technique to the resulting discretization. One of the main features of this approach is that it reduces the globally coupled unknowns to the numerical trace of the velocity and the mean of the pressure on the element boundaries, thereby leading to a significant reduction in the size of the resulting matrix. Moreover, by using an augmented lagrangian method, the globally coupled unknowns are further reduced to the numerical trace of the velocity only. Another important feature is that the approximations of the velocity, pressure, and gradient converge with the optimal order of $k+1$ in the L^2-norm, when polynomials of degree $k \geqslant 0$ are used to represent the approximate variables. Based on the optimal convergence of the HDG method, we apply an element-by-element postprocessing scheme to obtain a new approximate velocity, which converges with order $k+2$ in the L^2-norm for $k \geqslant 1$. The postprocessing performed at the element level is less expensive than the solution procedure. Numerical results are provided to assess the performance of the method.

1. Introduction

In this paper, we introduce a hybridizable discontinuous Galerkin (HDG) method for the Stokes system

$$\begin{aligned} -\nu\Delta\boldsymbol{u}+\nabla p &= \boldsymbol{f}, && \text{in } \Omega,\\ \nabla\cdot\boldsymbol{u} &= 0, && \text{in } \Omega,\\ \boldsymbol{u} &= \boldsymbol{g}, && \text{on } \partial\Omega, \end{aligned} \tag{1}$$

where Ω is a bounded domain in $\mathbb{R}^d$ with Lipschitz boundary $\partial\Omega$, ν is a viscosity, $\boldsymbol{f}\in[L^2(\Omega)]^d$ is a given source term, and $\boldsymbol{g}\in[H^{1/2}(\partial\Omega)]^d$ is the Dirichlet boundary data. We assume that ν is a constant function on Ω and that $\boldsymbol{g}$ satisfies the compatibility condition

$$\int_{\partial\Omega}\boldsymbol{g}\cdot\boldsymbol{n}=0,$$

where $\boldsymbol{n}$ denotes the outward unit normal vector on the boundary $\partial\Omega$. We thus continue the study of HDG methods for second-order, symmetric elliptic problems [5,9,11], convection–diffusion problems [15], and nonlinear convection–diffusion problems [16]. Our long-term goal is to devise HDG methods for the incompressible Navier–Stokes equations; the consideration of the Stokes system is thus a necessary intermediate step towards this goal.

Several hybridizable methods have been developed for the velocity–pressure–vorticity formulations of the Stokes flow. Hybridization, as a technique to avoid the construction of divergence-free velocities, was introduced in [4] for an LDG method. The technique was then extended to a classical mixed method for the two-dimensional and three-dimensional Stokes problems in [6,7]. This was the first hybridization of the Stokes problem giving rise to a global system involving the degrees of freedom of unknowns (the tangential velocity and the pressure) defined solely on the faces of the elements. Recently, motivated by the fact that the HDG methods for second-order symmetric elliptic equations [9] can be efficiently implemented and are more accurate than other DG methods [5,11], a new class of HDG methods was introduced for the Stokes problem [8]. It was shown to be hybridizable in four different ways including a tangential velocity/pressure hybridization and a velocity/average pressure hybridization.

In this paper, we apply the approach proposed in [8] to devise a new HDG method for the velocity–pressure–gradient formulation of the Stokes equations

$$\begin{aligned} \mathbf{L}-\nabla\boldsymbol{u} &= \mathbf{0}, && \text{in } \Omega,\\ \nabla\cdot(-\nu\mathbf{L}+p\mathbf{I}) &= \boldsymbol{f}, && \text{in } \Omega,\\ \nabla\cdot\boldsymbol{u} &= 0, && \text{in } \Omega,\\ \boldsymbol{u} &= \boldsymbol{g}, && \text{on } \partial\Omega. \end{aligned} \tag{2}$$

* Corresponding author.
E-mail address: cuongng@mit.edu (N.C. Nguyen).

0045-7825/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2009.10.007

Here $\mathbf{L} = \nabla \boldsymbol{u}$ is the second-order velocity gradient tensor and $\mathbf{I}$ is the second-order identity tensor. The resulting method is closely related to the DG method proposed in [12] and to the mixed method proposed in [17]. Indeed, the only difference between the HDG method we propose here and the DG method considered in [12] is the definition of the numerical traces. However, this difference allows us to hybridize the method, to obtain optimally convergent approximations for the velocity gradient and the pressure, and to obtain superconvergence properties for the velocity. The method proposed in [17] can be hybridized to give a globally coupled system in terms of the velocity on the faces of the elements and the pressure in the whole domain, whereas our method can be hybridized to give a system involving the velocity on the faces and the average of the pressure on the elements. For $k = 0$, the method proposed in [17] is not defined. For $k \geqslant 1$, it uses smaller spaces and is able to provide approximations converging with the same order of accuracy as ours. However, it does not have the ability of handling nonmatching meshes and variable-degree approximations typical of DG methods.

The HDG method applied to the above system is devised as follows. First, we introduce the numerical trace of the velocity and the mean of the pressure on each element as new approximate variables. On each element of the triangulation, we can now express the approximate velocity, pressure and gradient in terms of the numerical trace of the velocity and the mean of pressure, which we shall refer as the local solver. Furthermore, new equations have to be added to the system to render it solvable. These equations enforce the continuity of the normal components of the total flux. In this way, the velocity, pressure, and gradient can be expressed in an element-by-element fashion in terms of the numerical trace of the velocity and the mean of the pressure. This allows us to obtain the final system involving only the numerical trace of the velocity and the mean of the pressure, thereby reducing the globally coupled unknowns significantly. In addition to the reduction in the number of unknowns, the HDG method possesses various advantages related to convergence properties and postprocessing of the numerical solution.

We show that the HDG method is well-defined, that is, that the numerical solution exists and is unique. We present numerical examples to examine the accuracy and convergence properties of the method. They indicate that when polynomials of degree $k \geqslant 0$ are used to represent the velocity, pressure, and velocity gradient, all the variables converge optimally with the order $k+1$ in the L^2-norm for Stokes problems with smooth solution. Moreover, we use an element-by-element postprocessing to obtain a new approximate velocity which converges with order $k+2$ in the L^2-norm for $k \geqslant 1$. Since the local postprocessing is performed at the element level, the postprocessed velocity is less expensive to compute than the original approximate solution. Therefore, the $k+1$ convergent pressure and $k+2$ convergent velocity can be computed at the cost of a DG approximation using polynomials of degree k.

Finally, we propose an efficient implementation of the HDG method via the augmented Lagrangian approach by introducing a time derivative of the pressure into the continuity equation. In this way, we can express the pressure in terms of the velocity, thereby eliminating the mean of the pressure from the local solver. As a result, we arrive at a system in terms of the approximate trace of the velocity only. The main advantage of this implementation strategy is that the new system has less degrees of freedom than the original system involving both the approximate trace of the velocity and the mean of the pressure.

The paper is organized as follows. In Section 2 we introduce the HDG method for solving the Stokes system and apply a local postprocessing to compute a new approximation of the velocity. In Section 3 we describe the detailed implementation of the HDG method. In Section 4 we provide numerical results to assess the convergence and accuracy of the method. Finally, in Section 5 we present some concluding remarks.

2. The hybridizable discontinuous Galerkin method

2.1. Notation

Before describing the HDG method, we need to introduce some notation. We denote by $\mathscr{T}_h$ a collection of disjoint regular elements K that partition Ω and set $\partial\mathscr{T}_h := \{\partial K : K \in \mathscr{T}_h\}$. For an element K of the collection $\mathscr{T}_h$, $F = \partial K \cap \partial\Omega$ is the boundary face if the $d-1$ Lebesgue measure of F is nonzero. For two elements K^+ and K^- of the collection $\mathscr{T}_h$, $F = \partial K^+ \cap \partial K^-$ is the interior face between K^+ and K^- if the $d-1$ Lebesgue measure of F is nonzero. We denote by $\mathscr{E}_h^o$ and $\mathscr{E}_h^\partial$ the set of interior and boundary faces, respectively. We set $\mathscr{E}_h = \mathscr{E}_h^o \cup \mathscr{E}_h^\partial$.

Let $\boldsymbol{n}^+$ and $\boldsymbol{n}^-$ be the outward unit normal vectors on two neighboring elements ∂K^+ and ∂K^-, respectively. We use $(\mathbf{G}^\pm, \boldsymbol{v}^\pm, q^\pm)$ to denote the traces of $(\mathbf{G}, \boldsymbol{v}, q)$ on F from the interior of $K^\pm$, where $\mathbf{G}$, $\boldsymbol{v}$, and q are second-order tensorial, vectorial, and scalar functions, respectively. Then, we define the jumps $[\![\cdot]\!]$ as follows. For $F \in \mathscr{E}_h^o$, we set

$$
\begin{aligned}
&[\![\mathbf{G}\boldsymbol{n}]\!] = \mathbf{G}^+\boldsymbol{n}^+ + \mathbf{G}^-\boldsymbol{n}^-,\\
&[\![\boldsymbol{v}\cdot\boldsymbol{n}]\!] = \boldsymbol{v}^+\cdot\boldsymbol{n}^+ + \boldsymbol{v}^-\cdot\boldsymbol{n}^-,\\
&[\![\boldsymbol{v}\otimes\boldsymbol{n}]\!] = \boldsymbol{v}^+\otimes\boldsymbol{n}^+ + \boldsymbol{v}^-\otimes\boldsymbol{n}^-,\\
&[\![q\boldsymbol{n}]\!] = q^+\boldsymbol{n}^+ + q^-\boldsymbol{n}^-.
\end{aligned}
$$

For $F \in \mathscr{E}_h^\partial$, the set of boundary edges on which $\mathbf{G}$, $\boldsymbol{v}$ and q are single-valued, we set

$$
\begin{aligned}
&[\![\mathbf{G}\boldsymbol{n}]\!] = \mathbf{G}\boldsymbol{n},\\
&[\![\boldsymbol{v}\cdot\boldsymbol{n}]\!] = \boldsymbol{v}\cdot\boldsymbol{n},\\
&[\![\boldsymbol{v}\otimes\boldsymbol{n}]\!] = \boldsymbol{v}\otimes\boldsymbol{n},\\
&[\![q\boldsymbol{n}]\!] = q\boldsymbol{n},
\end{aligned}
$$

where $\boldsymbol{n}$ is the unit outward normal to $\partial\Omega$. Here $\cdot$ denotes the usual dot product and $\otimes$ denotes the usual dyadic or tensor product.

Now, let $\mathscr{P}_k(D)$ denote the space of polynomials of degree at most k on a domain D and let $L^2(D)$ be the space of square integrable functions on D. We set $\boldsymbol{\mathscr{P}}_k(D) = [\mathscr{P}_k(D)]^d$, $\mathbf{P}_k(D) = [\mathscr{P}_k(D)]^{d\times d}$, $\boldsymbol{L}^2(D) = [L^2(D)]^d$, and $\mathbf{L}^2(D) = [L^2(D)]^{d\times d}$. We introduce discontinuous finite element approximation spaces for the gradient, velocity, and pressure as

$$
\begin{aligned}
&\mathbf{Y}_h = \{\mathbf{G} \in \mathbf{L}^2(\mathscr{T}_h) : \mathbf{G}|_K \in \mathbf{P}_k(K),\ \forall K \in \mathscr{T}_h\},\\
&\boldsymbol{V}_h = \{\boldsymbol{v} \in \boldsymbol{L}^2(\mathscr{T}_h) : \boldsymbol{v}|_K \in \boldsymbol{\mathscr{P}}_k(K),\ \forall K \in \mathscr{T}_h\},\\
&P_h = \{q \in L^2(\mathscr{T}_h) : q|_K \in \mathscr{P}_k(K),\ \forall K \in \mathscr{T}_h\}.
\end{aligned}
$$

In addition, we introduce a finite element approximation space for the approximate trace of the velocity

$$
\boldsymbol{M}_h = \{\boldsymbol{\mu} \in \boldsymbol{L}^2(\mathscr{E}_h) : \boldsymbol{\mu}|_F \in \boldsymbol{\mathscr{P}}_k(F),\ \forall F \in \mathscr{E}_h\}.
$$

We also set

$$
\boldsymbol{M}_h(\boldsymbol{g}) = \{\boldsymbol{\mu} \in \boldsymbol{M}_h : \boldsymbol{\mu} = \mathsf{P}\boldsymbol{g} \text{ on } \partial\Omega\},
$$

where P denotes the L^2-projection into the space $\{\boldsymbol{\mu}|_{\partial\Omega}\ \forall \boldsymbol{\mu} \in \boldsymbol{M}_h\}$. Note that $\boldsymbol{M}_h$ consists of functions which are continuous inside the faces (or edges) $F \in \mathscr{E}_h$ and discontinuous at their borders. We further denote by $\overline{\Psi}_h$ the set of functions in $L^2(\partial\mathscr{T}_h)$ that are constant on each ∂K for all elements K

$$
\overline{\Psi}_h = \{r \in L^2(\partial\mathscr{T}_h) : r \in \mathscr{P}_0(\partial K),\ \forall K \in \mathscr{T}_h\}.
$$

The mean of our approximate pressure will belong to this space. Here the mean is defined as follows. For a given function q in

$L^2(\partial\mathscr{T}_h)$, we use $\bar{q}$ to define the mean of q on the element boundaries ∂K of an element K as follows:

$$\bar{q}|_{\partial K} = \frac{1}{|\partial K|}\int_{\partial K} q.$$

Obviously, we have $\bar{q} = q$ for any q in $\overline{\Psi}_h$.

Finally, we define various inner products for our finite element spaces. For functions r, q in $L^2(D)$, we denote $(r,q)_D = \int_D rq$ if D is a domain in $\mathbb{R}^d$ and $\langle r,q\rangle_D = \int_D rq$ if D is a domain in $\mathbb{R}^{d-1}$. Likewise, for functions $\boldsymbol{w}, \boldsymbol{v}$ in $\boldsymbol{L}^2(D)$, we denote $(\boldsymbol{w},\boldsymbol{v})_D = \int_D \boldsymbol{w}\cdot\boldsymbol{v}$ if D is a domain in $\mathbb{R}^d$ and $\langle \boldsymbol{w},\boldsymbol{v}\rangle_D = \int_D \boldsymbol{w}\cdot\boldsymbol{v}$ if D is a domain in $\mathbb{R}^{d-1}$. For functions $\mathbf{H}, \mathbf{G}$ in $\mathbf{L}^2(D)$, we denote $(\mathbf{H},\mathbf{G})_D = \int_D \mathbf{H}:\mathbf{G}$ if D is a domain in $\mathbb{R}^d$ and $\langle \mathbf{H},\mathbf{G}\rangle_D = \int_D \mathbf{H}:\mathbf{G}$ if D is a domain in $\mathbb{R}^{d-1}$; recall the standard notation $\mathbf{H}:\mathbf{G} = \mathrm{tr}(\mathbf{H}^T\mathbf{G})$, where tr is the trace operator. We define the volume inner products as

$$(r,q)_{\mathscr{T}_h} = \sum_{K\in\mathscr{T}_h}(r,q)_K,$$
$$(\boldsymbol{w},\boldsymbol{v})_{\mathscr{T}_h} = \sum_{K\in\mathscr{T}_h}(\mathrm{w},\boldsymbol{v})_K,$$
$$(\mathbf{H},\mathbf{G})_{\mathscr{T}_h} = \sum_{K\in\mathscr{T}_h}(\mathbf{H},\mathbf{G})_K,$$

for $r, q \in L^2(\mathscr{T}_h)$, $\boldsymbol{w}, \boldsymbol{v} \in \boldsymbol{L}^2(\mathscr{T}_h)$, and $\mathbf{H}, \mathbf{G} \in \mathbf{L}^2(\mathscr{T}_h)$. We also define the boundary inner products as

$$\langle r,q\rangle_{\partial\mathscr{T}_h} = \sum_{K\in\mathscr{T}_h}\langle r,q\rangle_{\partial K},$$
$$\langle \boldsymbol{w},\boldsymbol{v}\rangle_{\partial\mathscr{T}_h} = \sum_{K\in\mathscr{T}_h}\langle \boldsymbol{w},\boldsymbol{v}\rangle_{\partial K},$$
$$\langle \mathbf{H},\mathbf{G}\rangle_{\partial\mathscr{T}_h} = \sum_{K\in\mathscr{T}_h}\langle \mathbf{H},\mathbf{G}\rangle_{\partial K},$$

for $r, q \in L^2(\mathscr{E}_h)$, $\boldsymbol{w}, \boldsymbol{v} \in \boldsymbol{L}^2(\mathscr{E}_h)$, and $\mathbf{H}, \mathbf{G} \in \mathbf{L}^2(\mathscr{E}_h)$.

2.2. Formulation

The point of departure for devising an HDG method is to define a local solver which computes approximate solutions within each element once a discrete approximation to $\boldsymbol{u}$, say $\hat{\boldsymbol{u}}_h$, is obtained on the boundary of every mesh element. However, we note that the local Stokes problem on each element with a Dirichlet boundary data is not solvable. Hence, to render our local solver well-defined we introduce a new variable $\bar{\rho}_h$ which approximates the mean of pressure p on the element boundaries. Specifically, our local Stokes problem on each element K is the following:

$$\begin{aligned}
&\mathbf{L} - \nabla\boldsymbol{u} = \mathbf{0}, && \text{in } K,\\
&\nabla\cdot(-\nu\mathbf{L} + p\mathbf{I}) = \boldsymbol{f}, && \text{in } K,\\
&\nabla\cdot\boldsymbol{u} = 0, && \text{in } K,\\
&\boldsymbol{u} = \hat{\boldsymbol{u}}_h, && \text{on } \partial K,\\
&\bar{p} = \bar{\rho}_h,
\end{aligned} \tag{3}$$

for the given pair $(\hat{\boldsymbol{u}}_h, \bar{\rho}_h)$. It is obvious that this local Stokes problem is well-posed.

We first seek an approximation $(\mathbf{L}_h, \boldsymbol{u}_h, p_h) \in \mathbf{Y}_h \times \boldsymbol{V}_h \times P_h$ such that for all $K \in \mathscr{T}_h$,

$$\begin{aligned}
&(\mathbf{L}_h,\mathbf{G})_K + (\boldsymbol{u}_h, \nabla\cdot\mathbf{G})_K = \langle\hat{\boldsymbol{u}}_h, \mathbf{G}\boldsymbol{n}\rangle_{\partial K},\\
&(\nu\mathbf{L}_h - p_h\mathbf{I}, \nabla\boldsymbol{v})_K + \langle(-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I})\boldsymbol{n}, \boldsymbol{v}\rangle_{\partial K} = (\boldsymbol{f},\boldsymbol{v})_K,\\
&-(\boldsymbol{u}_h, \nabla q)_K = -\langle\hat{\boldsymbol{u}}_h\cdot\boldsymbol{n}, q - \bar{q}\rangle_{\partial K},\\
&\bar{p}_h = \bar{\rho}_h,
\end{aligned} \tag{4}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K)\times\boldsymbol{P}_k(K)\times P_k(K)$. Here the numerical traces $\hat{\boldsymbol{u}}_h$ and $-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I}$ are approximations to u and $-\nu\mathbf{L} + p\mathbf{I}$ over ∂K, respectively. Furthermore, $\bar{\rho}_h$ is an approximation of the mean of the pressure on the element boundary ∂K. Note that (4) is nothing but the DG discretization of the local Stokes problem (3). Note also that the presence of $\bar{q}$ in the third equation of (4) is necessary to enforce the identity $\langle\hat{\boldsymbol{u}}_h\cdot\boldsymbol{n}, q - \bar{q}\rangle_{\partial K} = 0$ for $q \in \overline{\Psi}_h$, since $(\boldsymbol{u}_h, \nabla q)_K = 0$ for $q \in \overline{\Psi}_h$.

Next, we specify the numerical trace $-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I}$ of the form

$$-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I} = -\nu\mathbf{L}_h + p_h\mathbf{I} + \mathbf{S}(\boldsymbol{u}_h - \hat{\boldsymbol{u}}_h)\otimes\boldsymbol{n}. \tag{5}$$

Here $\mathbf{S}$ is the second-order tensor consisting of stabilization parameters. It has an important effect on both the stability and accuracy of the resulting scheme. The selection of the stabilization tensor $\mathbf{S}$ will be discussed in our numerical experiments given in Section 4. Let us briefly motivate the choice of the above numerical trace. We want it to depend only on $(\mathbf{L}_h, \boldsymbol{u}_h, p_h, \hat{\boldsymbol{u}}_h)$, so that we can express $(\mathbf{L}_h, \boldsymbol{u}_h, p_h)$ in terms of $(\hat{\boldsymbol{u}}_h, \bar{\rho}_h)$. As we shall discuss in Section 2.3, this allows us to eliminate all other variables to obtain a weak formulation in terms of $(\hat{\boldsymbol{u}}_h, \bar{\rho}_h)$ only.

By adding the contribution of (4) over all the elements, and appending three suitably chosen equations, we arrive at the following problem: find an approximation $(\mathbf{L}_h, \boldsymbol{u}_h, p_h, \hat{\boldsymbol{u}}_h, \bar{\rho}_h) \in \mathbf{Y}_h\times\boldsymbol{V}_h\times P_h\times\boldsymbol{M}_h(\boldsymbol{g})\times\overline{\Psi}_h$ such that

$$\begin{aligned}
&(\mathbf{L}_h,\mathbf{G})_{\mathscr{T}_h} + (\boldsymbol{u}_h, \nabla\cdot\mathbf{G})_{\mathscr{T}_h} - \langle\hat{\boldsymbol{u}}_h, \mathbf{G}\boldsymbol{n}\rangle_{\partial\mathscr{T}_h} = 0,\\
&(\nu\mathbf{L}_h - p_h\mathbf{I}, \nabla\boldsymbol{v})_{\mathscr{T}_h} + \left\langle(-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I})\boldsymbol{n}, \mathrm{v}\right\rangle_{\partial\mathscr{T}_h} = (\boldsymbol{f},\boldsymbol{v})_{\mathscr{T}_h},\\
&-(\boldsymbol{u}_h, \nabla q)_{\mathscr{T}_h} + \langle\hat{\boldsymbol{u}}_h\cdot\boldsymbol{n}, q - \bar{q}\rangle_{\partial\mathscr{T}_h} = 0,\\
&\bar{p}_h = \bar{\rho}_h,\\
&\left\langle(-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} = 0,\\
&\left\langle\hat{\boldsymbol{u}}_h\cdot\boldsymbol{n}, \bar{\psi}\right\rangle_{\partial\mathscr{T}_h} = 0,\\
&(p_h, 1)_{\mathscr{T}_h} = 0.
\end{aligned} \tag{6}$$

for all $(\mathbf{G}, \boldsymbol{v}, q, \boldsymbol{\mu}, \bar{\psi}) \in \mathbf{Y}_h\times\boldsymbol{V}_h\times P_h\times\boldsymbol{M}_h(\mathbf{0})\times\overline{\Psi}_h$. Note that the Dirichlet boundary condition has been enforced by requiring that $\hat{\boldsymbol{u}}_h \in \boldsymbol{M}_h(\boldsymbol{g})$.

Let us briefly comment on the three added equations. The first enforces the continuity of the normal component of the numerical trace of the total flux $-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I}$ on the interelement boundaries. The second is needed for the consistency of the method and ensures that the velocity can be weakly, locally divergence-free. Finally, the third equation is just the average pressure constraint and is needed for the sake of well-posedness of the method.

Let us briefly comment on the conservative property of our numerical fluxes. We observe that $\hat{\boldsymbol{u}}_h$ is single-valued over $\mathscr{E}_h$ since $\hat{\boldsymbol{u}}_h$ belongs to $\boldsymbol{M}_h$. Furthermore, if $(-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I})\mathrm{n}$ belongs to $\boldsymbol{M}_h$, then the fifth equation in (6) simply states that $[\![(-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I})\boldsymbol{n}]\!] = 0$ pointwise over the interior faces $\mathscr{E}_h^o$; in other words, the normal component of the numerical trace $-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I}$ is single-valued. Hence, both $\hat{\boldsymbol{u}}_h$ and $-\nu\hat{\mathbf{L}}_h + \hat{p}_h\mathbf{I}$ are conservative fluxes according to the definition in [1].

Let us briefly motivate the hybridization of the HDG method. At first sight, the system (6) appears unattractive since it involves too many unknowns. However, it turns out that by appealing to the local solver (4) we can eliminate all the variables $\mathbf{L}_h$, $\boldsymbol{u}_h$, and p_h to obtain a weak formulation in terms of $(\hat{\boldsymbol{u}}_h, \bar{\rho}_h)$ only. After solving for $(\hat{\boldsymbol{u}}_h, \bar{\rho}_h)$, the approximate gradient, velocity, and pressure, $(\mathbf{L}_h, \boldsymbol{u}_h, p_h)$, can be inexpensively computed in an element-by-element fashion. Since $\hat{\boldsymbol{u}}_h$ is defined on element faces and since $\bar{\rho}_h$ is defined as a constant function on each element boundary, the hybridization approach reduces significantly the number of the globally coupled unknowns. This is precisely what we are going to accomplish in the next subsection.

2.3. Characterization of the numerical trace $\hat{\boldsymbol{u}}_h$ and the pressure mean $\bar{\rho}_h$

We begin by introducing the specific local solvers. The first local solver associates to $\boldsymbol{f} \in \boldsymbol{L}^2(\Omega)$ the function $(\mathbf{L}_h^f, \boldsymbol{u}_h^f, p_h^f) \in (\mathbf{Y}_h, \boldsymbol{V}_h, P_h)$

satisfying (4) when we set $\hat{\boldsymbol{u}}_h = 0$ and $\bar{\rho}_h = 0$. The second local solver associates to $\boldsymbol{\eta} \in \boldsymbol{M}_h$ the function $(\mathbf{L}_h^{\eta}, \boldsymbol{u}_h^{\eta}, p_h^{\eta}) \in (\mathbf{Y}_h, \boldsymbol{V}_h, P_h)$ satisfying (4) when we set $\boldsymbol{f} = 0, \bar{\rho}_h = 0$, and $\hat{\boldsymbol{u}}_h = \boldsymbol{\eta}$. And the third local solver associates to $\bar{\psi} \in \overline{\Psi}_h$ the function $(\mathbf{L}_h^{\bar{\psi}}, \boldsymbol{u}_h^{\bar{\psi}}, p_h^{\bar{\psi}}) \in (\mathbf{Y}_h, \boldsymbol{V}_h, P_h)$ satisfying (4) when we set $\boldsymbol{f} = 0, \hat{\mathrm{u}}_h = 0$, and $\bar{\rho}_h = \bar{\psi}$. More precisely, we have

$$
\begin{aligned}
(\mathbf{L}_h^{\boldsymbol{f}}, \boldsymbol{u}_h^{\boldsymbol{f}}, p_h^{\boldsymbol{f}}) &:= \mathscr{L}(\boldsymbol{f}, \mathbf{0}, 0),\\
(\mathbf{L}_h^{\boldsymbol{\eta}}, \boldsymbol{u}_h^{\boldsymbol{\eta}}, p_h^{\boldsymbol{\eta}}) &:= \mathscr{L}(\mathbf{0}, \boldsymbol{\eta}, 0),\\
(\mathbf{L}_h^{\bar{\psi}}, \boldsymbol{u}_h^{\bar{\psi}}, p_h^{\bar{\psi}}) &:= \mathscr{L}(\mathbf{0}, \mathbf{0}, \bar{\psi}),
\end{aligned}
$$

where $\mathscr{L}$ denotes the local solver (4) that maps $(\boldsymbol{f}, \hat{\boldsymbol{u}}_h, \bar{\rho}_h)$ into $(\mathbf{L}_h, \mathrm{u}_h, p_h)$.

We next introduce the numerical traces

$$
\begin{aligned}
-\nu\widehat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}}\mathbf{I} &= -\nu\mathbf{L}_h^{\boldsymbol{f}} + p_h^{\boldsymbol{f}}\mathbf{I} + \mathbf{S}(\mathrm{u}_h^{\boldsymbol{f}} \otimes \boldsymbol{n}),\\
-\nu\widehat{\mathbf{L}}_h^{\boldsymbol{\eta}} + \hat{p}_h^{\boldsymbol{\eta}}\mathbf{I} &= -\nu\mathbf{L}_h^{\boldsymbol{\eta}} + p_h^{\boldsymbol{\eta}}\mathbf{I} + \mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta}} - \boldsymbol{\eta}) \otimes \boldsymbol{n},\\
-\nu\widehat{\mathbf{L}}_h^{\bar{\psi}} + \hat{p}_h^{\bar{\psi}}\mathbf{I} &= -\nu\mathbf{L}_h^{\bar{\psi}} + p_h^{\bar{\psi}}\mathbf{I} + \mathbf{S}(\boldsymbol{u}_h^{\bar{\psi}} \otimes \boldsymbol{n}).
\end{aligned}
\tag{7}
$$

We obtain the following auxiliary result whose proof is given in Appendix A.

Lemma 2.1. *For any $\boldsymbol{\eta}, \boldsymbol{\mu} \in \boldsymbol{M}_h$ and $\bar{\psi} \in \overline{\Psi}_h$, we have*

$$
\begin{aligned}
-\left\langle(-\nu\widehat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}}\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} &= -(\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}})_{\mathscr{T}_h},\\
-\left\langle(-\nu\widehat{\mathbf{L}}_h^{\boldsymbol{\eta}} + \hat{p}_h^{\boldsymbol{\eta}}\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} &= (\nu\mathbf{L}_h^{\boldsymbol{\mu}}, \mathbf{L}_h^{\boldsymbol{\eta}})_{\mathscr{T}_h} + \left\langle\mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta}} - \boldsymbol{\eta}), (\boldsymbol{u}_h^{\boldsymbol{\mu}} - \boldsymbol{\mu})\right\rangle_{\partial\mathscr{T}_h},\\
-\left\langle(-\nu\widehat{\mathbf{L}}_h^{\bar{\psi}} + \hat{p}_h^{\bar{\psi}}\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} &= -\langle\bar{\psi}, \boldsymbol{\mu}\cdot\boldsymbol{n}\rangle_{\partial\mathscr{T}_h}.
\end{aligned}
\tag{8}
$$

We are ready to state the characterization result.

Theorem 2.1. *Let $(\mathbf{L}_h, \boldsymbol{u}_h, p_h, \hat{\boldsymbol{u}}_h, \bar{\rho}_h)$ be the solution of (6). We have that*

$$
\begin{aligned}
\mathbf{L}_h &= \mathbf{L}_h^{\boldsymbol{f}} + \mathbf{L}_h^{\boldsymbol{\lambda}},\\
\boldsymbol{u}_h &= \boldsymbol{u}_h^{\boldsymbol{f}} + \boldsymbol{u}_h^{\boldsymbol{\lambda}},\\
p_h &= p_h^{\boldsymbol{f}} + p_h^{\boldsymbol{\lambda}} + p_h^{\varrho},\\
\hat{\boldsymbol{u}}_h &= \boldsymbol{\lambda},\\
\bar{\rho}_h &= \varrho,
\end{aligned}
\tag{9}
$$

where $(\boldsymbol{\lambda}, \varrho) \in (\boldsymbol{M}_h(\boldsymbol{g}), \bar{\Psi}_h)$ satisfies

$$
\begin{aligned}
&a_h(\boldsymbol{\lambda}, \boldsymbol{\mu}) + b_h(\varrho, \boldsymbol{\mu}) = \ell_h(\boldsymbol{\mu}), \quad \forall\boldsymbol{\mu} \in \boldsymbol{M}_h(\mathbf{0}),\\
&b_h(\bar{\psi}, \boldsymbol{\lambda}) = 0, \qquad\qquad\quad \forall\bar{\psi} \in \bar{\Psi}_h,
\end{aligned}
\tag{10}
$$

and

$$
(p_h^{\boldsymbol{f}} + p_h^{\boldsymbol{\lambda}} + p_h^{\varrho}, 1)_{\mathscr{T}_h} = 0.
$$

Here the forms are given by

$$
\begin{aligned}
a_h(\boldsymbol{\eta}, \boldsymbol{\mu}) &= (\nu\mathbf{L}_h^{\boldsymbol{\eta}}, \mathbf{L}_h^{\boldsymbol{\mu}})_{\mathscr{T}_h} + \left\langle\mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta}} - \boldsymbol{\eta}), (\boldsymbol{u}_h^{\boldsymbol{\mu}} - \boldsymbol{\mu})\right\rangle_{\partial\mathscr{T}_h},\\
b_h(\bar{\psi}, \boldsymbol{\mu}) &= -\langle\bar{\psi}_h, \boldsymbol{\mu}\cdot\boldsymbol{n}\rangle_{\partial\mathscr{T}_h},\\
\ell_h(\boldsymbol{\mu}) &= (\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}})_{\mathscr{T}_h},
\end{aligned}
\tag{11}
$$

for all $\boldsymbol{\eta} \in \boldsymbol{M}_h$, $\boldsymbol{\mu} \in \boldsymbol{M}_h$, and $\bar{\psi} \in \overline{\Psi}_h$.

Proof. We first note from (6) and (7) that $(\mathbf{L}_h, \mathrm{u}_h, p_h, \hat{\boldsymbol{u}}_h, \bar{\rho}_h)$ satisfies

$$
\begin{aligned}
&\mathbf{L}_h = \mathbf{L}_h^{\boldsymbol{f}} + \mathbf{L}_h^{\boldsymbol{\lambda}},\\
&\boldsymbol{u}_h = \boldsymbol{u}_h^{\boldsymbol{f}} + \boldsymbol{u}_h^{\boldsymbol{\lambda}},\\
&p_h = p_h^{\boldsymbol{f}} + p_h^{\boldsymbol{\lambda}} + \varrho,\\
&\hat{\boldsymbol{u}}_h = \boldsymbol{\lambda},\\
&\bar{\rho}_h = \varrho,
\end{aligned}
$$

where $(\boldsymbol{\lambda}, \varrho) \in (\boldsymbol{M}_h(\boldsymbol{g}), \bar{\Psi}_h)$ is such that

$$
\begin{aligned}
&\left\langle(-\nu(\hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{\mathbf{L}}_h^{\boldsymbol{\lambda}} + \hat{\mathbf{L}}_h^{\varrho}) + (\hat{p}_h^{\mathrm{f}} + \hat{p}_h^{\boldsymbol{\lambda}} + \hat{p}_h^{\varrho})\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} = 0, \quad \forall\boldsymbol{\mu} \in \boldsymbol{M}_h(\mathbf{0}),\\
&-\langle\boldsymbol{\lambda}\cdot\boldsymbol{n}, \bar{\psi}_h\rangle_{\partial\mathscr{T}_h} = 0, \quad \forall\bar{\psi} \in \bar{\Psi}_h,\\
&(p_h^{\boldsymbol{f}} + p_h^{\boldsymbol{\lambda}} + p_h^{\varrho}, 1)_{\mathscr{T}_h} = 0.
\end{aligned}
\tag{12}
$$

The desired result then follows from Lemma 2.1 and (10)–(12). □

2.4. Existence and uniqueness of the numerical solution

A multi-valued tensor $\mathbf{S}$ is said to be *positive-definite* on a face F, if both branches, $\mathbf{S}^+$ and $\mathbf{S}^-$, of $\mathbf{S}$ are positive-definite, namely,

$$
\langle\mathbf{S}^{\pm}\boldsymbol{v}, \boldsymbol{v}\rangle_F > 0, \quad \forall\boldsymbol{v} \neq \mathbf{0}. \tag{13}
$$

When $\mathbf{S}$ is positive-definite on *all* faces of $\mathscr{E}_h$ we say that $\mathbf{S}$ is strictly positive-definite and indicate this by $\mathbf{S} > 0$ on $\mathscr{E}_h$.

With a strictly positive-definite stabilization tensor $\mathbf{S}$ we can prove the existence and uniqueness of the HDG solution as follows. We first need to prove the well-posedness of the local solvers.

Lemma 2.2. *If the stabilization tensor $\mathbf{S}$ satisfies the condition*

$$
\mathbf{S} > 0 \quad \text{on } \mathscr{E}_h, \tag{14}
$$

we have that both $(\mathbf{L}_h^{\boldsymbol{f}}, \boldsymbol{u}_h^{\boldsymbol{f}}, p_h^{\boldsymbol{f}})$ and $(\mathbf{L}_h^{\boldsymbol{\eta}}, \mathrm{u}_h^{\boldsymbol{\eta}}, p_h^{\boldsymbol{\eta}})$ exist and are unique.

Proof. Substituting (7) into (49) we obtain

$$
(\mathbf{L}_h^{\boldsymbol{f}}, \mathbf{G})_K + (\boldsymbol{u}_h^{\boldsymbol{f}}, \nabla\cdot\mathbf{G})_K = 0, \tag{15a}
$$

$$
(\nabla\cdot(-\nu\mathbf{L}_h^{\boldsymbol{f}} + p_h^{\boldsymbol{f}}\mathbf{I}), \boldsymbol{v})_K + \left\langle\mathbf{S}\boldsymbol{u}_h^{\boldsymbol{f}}, \boldsymbol{v}\right\rangle_{\partial K} = (\boldsymbol{f}, \boldsymbol{v})_K, \tag{15b}
$$

$$
-(\boldsymbol{u}_h^{\boldsymbol{f}}, \nabla q)_K = 0, \tag{15c}
$$

$$
\bar{p}_h^{\boldsymbol{f}} = 0, \tag{15d}
$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{P}_k(K) \times P_k(K)$. Due to the linearity, finite dimensionality, and to the fact that this is a square system, it is sufficient to show that the only solution of the above system for $\boldsymbol{f} = 0$ is $(\mathbf{L}_h^{\boldsymbol{f}}, \boldsymbol{u}_h^{\boldsymbol{f}}, p_h^{\boldsymbol{f}}) = (\mathbf{0}, \mathbf{0}, 0)$. Indeed, taking $\mathbf{G} = \nu\mathbf{L}_h^{\boldsymbol{f}}$, $\mathrm{v} = \boldsymbol{u}_h^{\boldsymbol{f}}$, and $q = p_h^{\boldsymbol{f}}$ and adding the first three equations, we get

$$
(\nu\mathbf{L}_h^{\boldsymbol{f}}, \mathbf{L}_h^{\boldsymbol{f}})_K + \langle\mathbf{S}\boldsymbol{u}_h^{\boldsymbol{f}}, \boldsymbol{u}_h^{\boldsymbol{f}}\rangle_{\partial K} = 0.
$$

This equation implies that $\mathbf{L}_h^{\boldsymbol{f}} = \mathbf{0}$ over the simplex K and that $\boldsymbol{u}_h^{\boldsymbol{f}} = \mathbf{0}$ on ∂K since we assume $\mathbf{S} > 0$. It thus follows from (15a) that

$$
(\nabla\boldsymbol{u}_h^{\boldsymbol{f}}, \mathbf{G})_K = 0, \quad \forall\mathbf{G} \in \mathbf{P}_k(K).
$$

Since $\boldsymbol{u}_h^{\boldsymbol{f}} \in \boldsymbol{P}_k(K)$ the above equation implies $\nabla\boldsymbol{u}_h^{\boldsymbol{f}}$ is constant over K. As a consequence, $\boldsymbol{u}_h^{\boldsymbol{f}} = 0$ over K since $\mathrm{u}_h^{\boldsymbol{f}} = 0$ on ∂K. Hence, from (15b) we have

$$
(\nabla p_h^{\boldsymbol{f}}, \boldsymbol{v})_K = 0, \quad \forall\boldsymbol{v} \in \mathrm{P}_k(K).
$$

Since $p_h^{\boldsymbol{f}} \in P_k(K)$ the above equation implies $p_h^{\boldsymbol{f}}$ is constant over K. We thus obtain that $p_h^{\boldsymbol{f}} = 0$ since $\bar{p}_h^{\boldsymbol{f}} = 0$. The existence and uniqueness of $(\mathbf{L}_h^{\boldsymbol{\eta}}, \mathrm{u}_h^{\boldsymbol{\eta}}, p_h^{\boldsymbol{\eta}})$ can be shown in the same manner. This completes the proof. □

Theorem 2.2. *If the stabilization parameter $\mathbf{S}$ satisfies the condition (14) we have that the solution $(\boldsymbol{\lambda}, \varrho)$ of the variational formulation (10) given in Theorem 2.1 exists and is unique.*

Proof. The existence and uniqueness of $(\boldsymbol{\lambda}, \varrho)$ follows if we show that the only solution of the problem (10) for $\boldsymbol{f} = \mathbf{0}$ and $\boldsymbol{g} = \mathbf{0}$ is $\boldsymbol{\lambda} = \mathbf{0}$ and $\varrho = 0$. In (10) we choose $\boldsymbol{\mu} = \boldsymbol{\lambda}$ and $\bar{\psi} = \varrho$ and subtract the first equation from the second one to obtain

$$
(\nu\mathbf{L}_h^{\boldsymbol{\lambda}}, \mathbf{L}_h^{\boldsymbol{\lambda}})_{\mathscr{T}_h} + \langle\mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\lambda}} - \boldsymbol{\lambda}), (\mathrm{u}_h^{\boldsymbol{\lambda}} - \boldsymbol{\lambda})\rangle_{\partial\mathscr{T}_h} = 0.
$$

As a consequence, we can conclude that $\mathbf{L}_h^{\boldsymbol{\lambda}} = \mathbf{0}$ on Ω and $\boldsymbol{u}_h^{\boldsymbol{\lambda}} = \boldsymbol{\lambda}$ on $\mathscr{E}_h$ since we assume $\mathbf{S} > 0$.

After a simple integration by parts, the Eq. (50a), with η replaced by λ, now reads

$$(\nabla \boldsymbol{u}_h^{\lambda}, \mathbf{G})_{\mathscr{T}_h} = 0, \quad \forall \mathbf{G} \in \mathbf{Y}_h.$$

This implies that $\boldsymbol{u}_h^{\lambda}$ is a constant on $\mathscr{T}_h$. As a consequence, λ is also a constant on $\mathscr{E}_h$ and, since $\boldsymbol{\lambda} \in \boldsymbol{M}_h(\mathbf{0})$, we must have that $\boldsymbol{\lambda} = \mathbf{0}$ on $\mathscr{E}_h$. Finally, we insert $\boldsymbol{\lambda} = \mathbf{0}$ into the first equation of (10) to obtain

$$\langle \varrho, \boldsymbol{\mu} \cdot \mathbf{n} \rangle_{\partial \mathscr{T}_h} = 0, \quad \forall \boldsymbol{\mu} \in \boldsymbol{M}_h(\mathbf{0}),$$

which implies $\varrho = 0$. This completes the proof of Theorem 2.2. □

As a consequence of Lemma 2.2 and Theorem 2.2, the approximate solution $(\mathbf{L}_h, \boldsymbol{u}_h, p_h, \hat{\boldsymbol{u}}_h, \bar{\rho}_h)$ of the mixed formulation (6) exists and is unique.

2.5. The general form of the numerical traces

We have shown how to eliminate $(\mathbf{L}_h, \boldsymbol{u}_h, p_h)$ to obtain a weak formulation in terms of $(\hat{\boldsymbol{u}}_h, \bar{\rho}_h)$. The key elements are the local solver (4), the definition of the numerical trace (5), and the jump condition

$$\langle [\![(-\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I}) \boldsymbol{n}]\!], \boldsymbol{\mu} \rangle_{\mathscr{E}_h} = 0, \quad \forall \boldsymbol{\mu} \in \mathsf{M}_h(\mathbf{0}). \tag{16}$$

Here we show that the HDG method can be formally stated by a formulation in which the unknown variables are $\mathbf{L}_h$, $\boldsymbol{u}_h$, and p_h only. We proceed as follows.

First, we derive an explicit expression for the numerical traces in terms of $(\mathbf{L}_h, \boldsymbol{u}_h, p_h)$. By the choice of the space $\boldsymbol{M}_h$, and assuming that the stabilization tensor $\mathbf{S}$ is constant on each face in $\mathscr{E}_h$, the jump condition (16) implies that

$$[\![(-\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I}) \boldsymbol{n}]\!] = 0, \quad \text{on } \mathscr{E}_h^{\mathrm{o}}.$$

Inserting (5) into the above equation, we obtain

$$[\![(-\nu \mathbf{L}_h + p_h \mathbf{I}) \boldsymbol{n}]\!] + \mathbf{S}^{+} \boldsymbol{u}_h^{+} + \mathbf{S}^{-} \boldsymbol{u}_h^{-} - (\mathbf{S}^{+} + \mathbf{S}^{-}) \hat{\boldsymbol{u}}_h = 0, \quad \text{on } \mathscr{E}_h^{\mathrm{o}}.$$

Since both $\mathbf{S}^{-}$ and $\mathbf{S}^{+}$ are positive definite, and $\mathbf{S}^{-} + \mathbf{S}^{+}$ is invertible we obtain that, on $\mathscr{E}_h^{\mathrm{o}}$,

$$\hat{\boldsymbol{u}}_h = (\mathbf{S}^{-} + \mathbf{S}^{+})^{-1} (\mathbf{S}^{+} \mathsf{u}_h^{+} + \mathbf{S}^{-} \boldsymbol{u}_h^{-}) + (\mathbf{S}^{-} + \mathbf{S}^{+})^{-1} [\![(-\nu \mathbf{L}_h + p_h \mathbf{I}) \boldsymbol{n}]\!]. \tag{17}$$

Substituting this expression into (5) yields on $\mathscr{E}_h^{\mathrm{o}}$,

$$\begin{aligned} -\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I} &= \mathbf{S}^{-} (\mathbf{S}^{-} + \mathbf{S}^{+})^{-1} (-\nu \mathbf{L}_h^{+} + p_h^{+} \mathbf{I}) \\ &\quad + \mathbf{S}^{+} (\mathbf{S}^{-} + \mathbf{S}^{+})^{-1} (-\nu \mathbf{L}_h^{-} + p_h^{-} \mathbf{I}) + \mathbf{S}^{-} (\mathbf{S}^{-} + \mathbf{S}^{+})^{-1} \mathbf{S}^{+} [\![\boldsymbol{u}_h \otimes \boldsymbol{n}]\!]. \end{aligned} \tag{18}$$

Recall that since $\hat{\boldsymbol{u}}_h \in \boldsymbol{M}_h(\boldsymbol{g})$, on the boundary faces $\mathscr{E}_h^{\partial}$, we have

$$\begin{aligned} &\hat{\boldsymbol{u}}_h = \mathsf{P} \boldsymbol{g}, \\ &-\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I} = -\nu \mathbf{L}_h + p_h \mathbf{I} + \mathbf{S} (\boldsymbol{u}_h - \mathsf{P} \boldsymbol{g}) \otimes \boldsymbol{n}. \end{aligned} \tag{19}$$

Thus, we can view the HDG method as: find an approximate solution $(\mathbf{L}_h, \boldsymbol{u}_h, p_h) \in (\mathbf{Y}_h, \boldsymbol{V}_h, P_h)$ such that

$$\begin{aligned} &(\mathbf{L}_h, \mathbf{G})_{\mathscr{T}_h} + (\boldsymbol{u}_h, \nabla \cdot \mathbf{G})_{\mathscr{T}_h} - \langle \hat{\boldsymbol{u}}_h, \mathbf{G} \boldsymbol{n} \rangle_{\partial \mathscr{T}_h} = \mathbf{0}, \\ &-(-\nu \mathbf{L}_h + p \mathbf{I}, \nabla \boldsymbol{v})_{\mathscr{T}_h} + \left\langle (-\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I}) \boldsymbol{n}, \mathsf{v} \right\rangle_{\partial \mathscr{T}_h} = (\boldsymbol{f}, \boldsymbol{v})_{\mathscr{T}_h}, \\ &-(\boldsymbol{u}_h, \nabla q)_{\mathscr{T}_h} + \langle \hat{\boldsymbol{u}}_h \cdot \boldsymbol{n}, q \rangle_{\partial \mathscr{T}_h} = 0, \\ &(p_h, 1)_{\mathscr{T}_h} = 0, \end{aligned} \tag{20}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in (\mathbf{Y}_h, \boldsymbol{V}_h, P_h)$, where the numerical traces, $\hat{\boldsymbol{u}}_h$ and $-\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I}$, are given by Eqs. (17)–(19).

Note that this is nothing but the weak formulation of the DG method proposed in [12]. The spaces are also identical to ours but, as we pointed out in the Introduction, the numerical traces are different. The difference lies in the definition of the numerical trace for the velocity. Indeed, in contrast with our choice, the numerical trace of the velocity used in [12] has *two* components: one for the first equation and another for the third equation. The one for the first equation lacks the term involving the jump of the total flux and the one for the third equation lacks the term involving the velocity gradient. This subtle difference is responsible for the huge difference in the approximation properties of the methods, as we are going to see in the Section of numerical experiments.

2.6. Local postprocessing of the velocity

We use the local postprocessing proposed in [17] to obtain a new approximate velocity, $\boldsymbol{u}_h^{*}$ of $\boldsymbol{u}$, which may converge at faster rate than the original approximation u_h. We define the postprocessed approximate velocity $\boldsymbol{u}_h^{*}$ on $K \in \mathscr{T}_h$ as the element of $\mathscr{P}_{k+1}(K)$ such that

$$\begin{aligned} &(\nu \nabla \boldsymbol{u}_h^{*}, \nabla \boldsymbol{v})_K = -(\mathbf{L}_h, \nabla \boldsymbol{v})_K, \quad \forall \boldsymbol{v} \in \mathscr{P}_{k+1}(K), \\ &(\boldsymbol{u}_h^{*}, 1)_K = (\boldsymbol{u}_h, 1)_K. \end{aligned} \tag{21}$$

To compute $\boldsymbol{u}_h^{*}$ we need only to invert a matrix of size equal to the dimension of $\mathscr{P}_{k+1}(K)$ for each element K of the triangulation $\mathscr{T}_h$. Therefore, the postprocessed velocity is less expensive to compute than the original approximate velocity.

2.7. Neumann boundary condition

Let us end this section by extending the method to the case when the Neumann boundary condition $(-\nu \mathbf{L} + p \mathbf{I}) \boldsymbol{n} = \mathsf{g}_N$ is enforced in part of the boundary $\partial\Omega, \partial\Omega_N$. First, we require that the approximate trace $\boldsymbol{\lambda}_h$ belongs to

$$\boldsymbol{M}_h(\boldsymbol{g}) = \{ \boldsymbol{\mu} \in \boldsymbol{M}_h : \boldsymbol{\mu} = \boldsymbol{g} \text{ on } \partial\Omega_D \}, \tag{22}$$

where $\partial\Omega_D = \partial\Omega \setminus \partial\Omega_N$ is the Dirichlet boundary. We then replace the jump condition with

$$\langle (-\nu \hat{\mathbf{L}}_h + \hat{p}_h \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \rangle_{\partial \mathscr{T}_h} = \langle \boldsymbol{g}_N, \boldsymbol{\mu} \rangle_{\partial\Omega_N}, \quad \forall \boldsymbol{\mu} \in \mathsf{M}_h(0). \tag{23}$$

As a result, the bilinear forms a_h and b_h of the weak formulation (10) remain unchanged, while the linear functional ℓ_h is now given by

$$\ell_h(\boldsymbol{\mu}) = (\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}}) + \langle \boldsymbol{g}_N, \boldsymbol{\mu} \rangle_{\partial\Omega_N}, \quad \forall \boldsymbol{\mu} \in \mathsf{M}_h. \tag{24}$$

Hence, in order to incorporate the Neumann condition $(-\nu \mathbf{L} + p \mathbf{I}) \boldsymbol{n} = \boldsymbol{g}_N$ on $\partial\Omega_N$ we need only to redefine the space $\boldsymbol{M}_h(\boldsymbol{g})$ according to (22) and modify ℓ_h according to (24).

3. Implementation aspects of the HDG method

In this section, we describe in detail how to efficiently implement the HDG method via an augmented Lagrangian approach; see [13] and the references therein. Towards this end, we introduce a time derivative of the pressure into the continuity equation. In this way, we can express the pressure in term of the velocity, thereby eliminating the mean of the pressure from the local solver. Thus, we arrive at a system in terms of the approximate trace of the velocity only. The efficiency of this implementation strategy lies in the fact that the new system has less degrees of freedom than the original system which involves both the approximate trace of the velocity and the mean of the pressure. Although the HDG method can also be implemented by using the Uzawa method, we choose the augmented Lagrangian method because it is more efficient than Uzawa method for solving the saddle point system associated with the Stokes problem. We refer to [13] for a detailed discussion.

3.1. Motivation of the method

The idea of the method is to introduce an *evolution* problem whose limit, as time goes to infinity, is nothing but the solution of the original problem. Let us show that for the continuous problem. For a given initial pressure $p_0 \in L_0^2(\Omega) := \{ q \in L^2(\Omega) : (q, 1)_{\Omega} = 0 \}$, the evolution problem is

$$\begin{aligned} &\tfrac{\partial p(t)}{\partial t} + \nabla\cdot\boldsymbol{u}(t) = 0, \quad \text{in } \Omega\times(0,\infty),\\ &p(t=0) = p_0, \qquad\quad \text{on } \partial\Omega, \end{aligned} \tag{25}$$

where $\boldsymbol{u}(t)$ is a function of $p(t)$ and defined as the solution of

$$\begin{aligned} &\mathbf{L}(t) - \nabla\boldsymbol{u}(t) = \mathbf{0}, \qquad \text{in}\Omega\times(0,\infty),\\ &\nabla\cdot(-\nu\mathbf{L}(t) + p(t)\mathbf{I}) = \boldsymbol{f}, \quad \text{in } \Omega\times(0,\infty),\\ &\boldsymbol{u}(t) = \boldsymbol{g}, \qquad\qquad \text{on } \partial\Omega\times(0,\infty). \end{aligned} \tag{26}$$

The system (25) and (26) is the time-dependent version of the original problem (1).

We proceed as follows. We first set $\delta^{\mathbf{L}}(t) := \mathbf{L}(t) - \mathbf{L}$, $\delta^{\boldsymbol{u}}(t) := \boldsymbol{u}(t) - \boldsymbol{u}$ and $\delta^{p}(t) := p(t) - p$, where we recall that $(\mathbf{L}, \boldsymbol{u}, p)$ is the solution of the steady-state original problem (1). It then follows from (25)-(26), and (1) that

$$\begin{aligned} &\tfrac{\partial \delta^p}{\partial t} + \nabla\cdot\delta^{\boldsymbol{u}} = 0, \qquad \text{in } \Omega\times(0,\infty),\\ &\delta^p(t=0) = p_0 - p, \quad \text{on } \partial\Omega, \end{aligned} \tag{27}$$

where $\delta^{\boldsymbol{u}}$ is a function of δ^p and defined as the solution of

$$\begin{aligned} &\delta^{\mathbf{L}} - \nabla\delta^{\boldsymbol{u}} = \mathbf{0}, \qquad \text{in } \Omega\times(0,\infty),\\ &-\nabla\cdot\delta^{\mathbf{L}} = -\nu^{-1}\,\nabla\cdot(\delta^p\mathbf{I}), \quad \text{in } \Omega\times(0,\infty),\\ &\delta^{\boldsymbol{u}} = \mathbf{0}, \qquad\qquad \text{on } \partial\Omega\times(0,\infty). \end{aligned} \tag{28}$$

Multiplying the first equation (27) by $\delta^p(t)$ and integrating on Ω, we get

$$\frac{1}{2}\frac{d}{dt}\|\delta^p\|^2 + \Theta = 0,$$

where $\|\cdot\|$ is the $L^2(\Omega)$-norm and $\Theta := (\nabla\cdot\delta^{\boldsymbol{u}}, \delta^p)_\Omega$. It follows from the equations (28) and integration by parts that

$$\Theta = -(\delta^{\boldsymbol{u}}, \nabla\cdot(\delta^p\mathbf{I}))_\Omega = -\nu(\delta^{\boldsymbol{u}}, \nabla\cdot(\delta^{\mathbf{L}}))_\Omega = \nu(\nabla\delta^{\boldsymbol{u}}, \delta^{\mathbf{L}})_\Omega = \nu\|\delta^{\mathbf{L}}\|^2.$$

Now, by the equations (28), we have that $\delta^{\mathbf{L}}(t) = \nu^{-1}\mathscr{I}(\delta^p(t)\mathbf{I})$, where $\mathscr{I} := \nabla(\Delta)^{-1}\nabla\cdot$. Since $\mathscr{I}$ is a self-adjoint, strongly elliptic operator, its smallest eigenvalue λ is strictly positive and we can write that

$$\Theta \geqslant \frac{\lambda^2}{\nu}\|\delta^p\|^2.$$

This implies that

$$\frac{1}{2}\frac{d}{dt}\|\delta^p\|^2 + \frac{\lambda^2}{\nu}\|\delta^p\|^2 \leqslant 0,$$

and, as a consequence, that

$$\|\delta^p(t)\| \leqslant e^{-t\lambda^2/\nu}\|\delta^p(0)\|.$$

This shows that as time goes to infinity, $p(t)$ converges exponentially fast in time to p. It follows from this result and from the equations (28) that $\mathbf{L}(t)$ and $\boldsymbol{u}(t)$ also converge exponentially fast in time to $\mathbf{L}$ and $\boldsymbol{u}$, respectively.

3.2. Augmented Lagrangian approach

The augmented Lagrangian method we use is obtained by discretizing equations (25) and (26) in time by using the backward-Euler method and in space by using the HDG formulation described Section 2 for the spatial discretization. The main difference, however, is that the local problem is now given by

$$\begin{aligned} &\tfrac{\partial p(t)}{\partial t} + \nabla\cdot\boldsymbol{u}(t) = 0, \qquad \text{in } K\times(0,\infty),\\ &p(t=0) = p_0, \qquad\qquad \text{on } \partial K,\\ &\mathbf{L}(t) - \nabla\boldsymbol{u}(t) = \mathbf{0}, \qquad \text{in } K\times(0,\infty),\\ &\nabla\cdot(-\nu\mathbf{L}(t) + p(t)\mathbf{I}) = \boldsymbol{f}, \quad \text{in } K\times(0,\infty),\\ &\boldsymbol{u}(t) = \hat{\boldsymbol{u}}_h(t), \qquad\qquad \text{on } \partial K\times(0,\infty). \end{aligned} \tag{29}$$

Note that we no longer need to use the mean of the pressure $\bar{p}_h(t)$ since this is now a time-dependent problem.

Thus, we begin defining the iterative method by providing the following initial guess for the approximate pressure $p_h^0 \in P_h$:

$$(p_h^0, q)_{\mathscr{T}_h} = (p_0, q)_{\mathscr{T}_h}, \tag{30}$$

for all $q \in P_h$. Next, given a constant time step Δt and a pressure p_h^{n-1} for $n \geqslant 1$, we define the iterate $p_h^n \in P_h$ as an approximation to $p(n\Delta t)$ such that

$$\frac{1}{\Delta t}(p_h^n, q)_{\mathscr{T}_h} - (\boldsymbol{u}_h^n, \nabla q)_{\mathscr{T}_h} + \langle \hat{\boldsymbol{u}}_h^n\cdot\boldsymbol{n}, q\rangle_{\partial\mathscr{T}_h} = \frac{1}{\Delta t}(p_h^{n-1}, q)_{\mathscr{T}_h}, \tag{31}$$

Here the functions $\boldsymbol{u}_h^n$ and $\hat{\boldsymbol{u}}_h^n$ are components of the function $(\mathbf{L}_h^n, \boldsymbol{u}_h^n, \hat{\mathrm{u}}_h^n) \in \mathbf{Y}_h\times\boldsymbol{V}_h\times\boldsymbol{M}_h(\boldsymbol{g})$ determined by the equations

$$\begin{aligned} &(\mathbf{L}_h^n, \mathbf{G})_{\mathscr{T}_h} + (\boldsymbol{u}_h^n, \nabla\cdot\mathbf{G})_{\mathscr{T}_h} - \langle \hat{\boldsymbol{u}}_h^n, \mathbf{G}\mathsf{n}\rangle_{\partial\mathscr{T}_h} = 0,\\ &-(-\nu\mathbf{L}_h^n + p_h^n\mathbf{I}, \nabla\boldsymbol{v})_{\mathscr{T}_h} + \left\langle (-\nu\hat{\mathbf{L}}_h^n + \hat{p}_h^n\mathbf{I})\boldsymbol{n}, \boldsymbol{v}\right\rangle_{\partial\mathscr{T}_h} = (\boldsymbol{f}, \mathsf{v})_{\mathscr{T}_h},\\ &\left\langle (-\nu\hat{\mathbf{L}}_h^n + \hat{p}_h^n\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} = 0, \end{aligned} \tag{32}$$

for all $(\mathbf{G}, \boldsymbol{v}, \boldsymbol{\mu}) \in \mathbf{Y}_h\times\boldsymbol{V}_h\times\boldsymbol{M}_h(\mathbf{0})$, where

$$-\nu\hat{\mathbf{L}}_h^n + \hat{p}_h^n\mathbf{I} = -\nu\mathbf{L}_h^n + p_h^n\mathbf{I} + \mathbf{S}(\boldsymbol{u}_h^n - \hat{\boldsymbol{u}}_h^n)\otimes\mathsf{n} \quad \text{on } \partial\mathscr{T}_h. \tag{33}$$

Here $\mathbf{L}_h^n$ and $\boldsymbol{u}_h^n$ are approximations to $\mathbf{L}(n\Delta t)$ and $\boldsymbol{u}(n\Delta t)$, respectively. The system of Eqs. ()()()(31)–(33) is nothing but the backward-Euler method for the temporal discretization and the HDG method described in Section 2 for the spatial discretization of the continuous equations (25) and (26).

3.3. Convergence to the original HDG approximation

In a similar manner as the exact solution, we show that as n goes to infinity, $(\mathbf{L}_h^n, \boldsymbol{u}_h^n, p_h^n)$ converges exponentially in time to the original HDG approximation $(\mathbf{L}_h, \boldsymbol{u}_h, p_h)$ introduced in Section 2. The proof of the following result can be found in Appendix B.

Lemma 3.1. *Let $\delta_h^{\mathbf{L},n} := \mathbf{L}_h^n - \mathbf{L}_h$, $\delta_h^{u,n} := \boldsymbol{u}_h^n - \boldsymbol{u}_h$, $\delta_h^{p,n} := p_h^n - p_h$ and $\delta_h^{\hat{\boldsymbol{u}},n} := \hat{\boldsymbol{u}}_h^n - \hat{\boldsymbol{u}}_h$. Then the error $\delta_h^{p,n} := p_h^n - p_h$ satisfies*

$$\frac{1}{2\Delta t}\left(\|\delta_h^{p,n}\|^2 - \|\delta_h^{p,n-1}\|^2\right) + \Theta_h^n \leqslant 0, \tag{34}$$

where

$$\Theta_h^n = \nu(\delta_h^{\mathbf{L},n}, \delta_h^{\mathbf{L},n})_{\mathscr{T}_h} + \langle \mathbf{S}(\delta_h^{\boldsymbol{u},n} - \delta_h^{\hat{\boldsymbol{u}},n}), (\delta_h^{\boldsymbol{u},n} - \delta_h^{\hat{\boldsymbol{u}},n})\rangle_{\partial\mathscr{T}_h}. \tag{35}$$

It now follows from Lemma 3.1 and Section 3.1 that

$$\Theta_h^n \leqslant \frac{\lambda_h^2}{\nu}\|\delta_h^{p,n}\|^2, \tag{36}$$

where λ_h is an approximation to the smallest eigenvalue λ. We thus obtain

$$\frac{1}{2\Delta t}(\|\delta_h^{p,n}\|^2 - \|\delta_h^{p,n-1}\|^2) + \frac{\lambda_h^2}{\nu}\|\delta_h^{p,n}\|^2 \leqslant 0, \tag{37}$$

which yields

$$\|\delta_h^{p,n}\| \leqslant \sigma^n\|\delta_h^{p,0}\|, \tag{38}$$

where $\sigma := (1 + 2\lambda_h^2\Delta t/\nu)^{-1/2}$. This result implies that as n goes to infinity, p_h^n converges to p_h. The same conclusion holds for the remaining components of the approximation.

In practice, we stop the iterations when the relative error of the pressure is less than a prescribed tolerance $\varepsilon_{\mathrm{tol}}$, that is, when $n = n_{\mathrm{iter}}$ such that

$$\frac{\|p_h^{n_{\mathrm{iter}}} - p_h^{n_{\mathrm{iter}}-1}\|_{\mathscr{T}_h}}{\|p_h^{n_{\mathrm{iter}}}\|_{\mathscr{T}_h}} < \varepsilon_{\mathrm{tol}}. \tag{39}$$

We note from (38) that n_{iter} depends on the artificial time step Δt in such a way that n_{iter} decreases as Δt increases.

3.4. Characterization of the approximate solution

Having established that the iterates of the augmented Lagrangian method converge to the approximation solution of the HDG method of Section 2, we now show how to efficiently implement the method. Towards this end, we begin by characterizing each of the iterates by means of a hybridization technique similar to that used to hybridize the original HDG method.

We first introduce three local solvers. The first local solver maps $\boldsymbol{f} \in \boldsymbol{L}^2(\Omega)$ to $(\mathbf{L}_h^{\boldsymbol{f},\Delta t}, \boldsymbol{u}_h^{\boldsymbol{f},\Delta t}, p_h^{\boldsymbol{f},\Delta t}) \in \mathbf{Y}_h \times \boldsymbol{V}_h \times P_h$ satisfying

$$\begin{aligned}
&(\mathbf{L}_h^{\boldsymbol{f},\Delta t}, \mathbf{G}^{\boldsymbol{f},\Delta t})_K + (\mathrm{u}_h^{\boldsymbol{f},\Delta t}, \nabla\cdot\mathbf{G})_K = 0,\\
&(\nabla\cdot(-\nu\mathbf{L}_h^{\boldsymbol{f},\Delta t} + p_h^{\boldsymbol{f},\Delta t}\mathbf{I}), \boldsymbol{v})_K + \left\langle \mathbf{S}\boldsymbol{u}_h^{\mathrm{f},\Delta t}, \boldsymbol{v}\right\rangle_{\partial K} = (\boldsymbol{f}, \mathrm{v})_K,\\
&\tfrac{1}{\Delta t}(p_h^{\boldsymbol{f},\Delta t}, q)_K - (\boldsymbol{u}_h^{\mathrm{f},\Delta t}, \nabla q)_K = 0,
\end{aligned} \tag{40}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{P}_k(K) \times P_k(K)$. The second local solver maps $\pi_h \in P_h$ to $(\mathbf{L}_h^{\pi_h,\Delta t}, \boldsymbol{u}_h^{\pi_h,\Delta t}, p_h^{\pi_h,\Delta t}) \in \mathbf{Y}_h \times \boldsymbol{V}_h \times P_h$ satisfying

$$\begin{aligned}
&(\mathbf{L}_h^{\pi_h,\Delta t}, \mathbf{G}^{\pi_h,\Delta t})_K + (\mathrm{u}_h^{\pi_h,\Delta t}, \nabla\cdot\mathbf{G})_K = 0,\\
&(\nabla\cdot(-\nu\mathbf{L}_h^{\pi_h,\Delta t} + p_h^{\pi_h,\Delta t}\mathbf{I}), \boldsymbol{v})_K + \left\langle \mathbf{S}\boldsymbol{u}_h^{\pi_h,\Delta t}, \boldsymbol{v}\right\rangle_{\partial K} = 0,\\
&\tfrac{1}{\Delta t}(p_h^{\pi_h,\Delta t}, q)_K - (\mathrm{u}_h^{\pi_h,\Delta t}, \nabla q)_K = \tfrac{1}{\Delta t}(\pi_h, q)_{\mathscr{T}_h},
\end{aligned} \tag{41}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{P}_k(K) \times P_k(K)$. The last local solver maps $\boldsymbol{\eta} \in \boldsymbol{M}_h$ to $(\mathbf{L}_h^{\boldsymbol{\eta},\Delta t}, \boldsymbol{u}_h^{\boldsymbol{\eta},\Delta t}, p_h^{\boldsymbol{\eta},\Delta t}) \in \mathbf{Y}_h \times \boldsymbol{V}_h \times P_h$ satisfying

$$\begin{aligned}
&(\mathbf{L}_h^{\boldsymbol{\eta},\Delta t}, \mathbf{G})_K + (\mathrm{u}_h^{\boldsymbol{\eta},\Delta t}, \nabla\cdot\mathbf{G})_K = \langle \boldsymbol{\eta}, \mathbf{G}\boldsymbol{n}\rangle_{\partial K},\\
&(\nabla\cdot(-\nu\mathbf{L}_h^{\boldsymbol{\eta},\Delta t} + p_h^{\boldsymbol{\eta},\Delta t}\mathbf{I}), \boldsymbol{v})_K + \left\langle \mathbf{S}\boldsymbol{u}_h^{\boldsymbol{\eta},\Delta t}, \boldsymbol{v}\right\rangle_{\partial K} = \langle \mathbf{S}\boldsymbol{\eta}, \boldsymbol{v}\rangle_{\partial K},\\
&\tfrac{1}{\Delta t}(p_h^{\boldsymbol{\eta},\Delta t}, q)_K - (\mathrm{u}_h^{\boldsymbol{\eta},\Delta t}, \nabla q)_K = -\langle \boldsymbol{\eta}\cdot\boldsymbol{n}, q\rangle_{\partial K},
\end{aligned} \tag{42}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{P}_k(K) \times P_k(K)$. Note that the local solvers here are different from the ones introduced in Section 2.

The following theorem characterizes the numerical trace $\hat{\mathrm{u}}_h^n$ as the solution of a variational formulation. The proof of this theorem follows almost directly from those of Theorems 2.1 and 2.2 and is thus omitted here.

Theorem 3.1. *Let $(\mathbf{L}_h^n, \boldsymbol{u}_h^n, p_h^n, \hat{\boldsymbol{u}}_h^n)$ be the solution of (31)–(33). We have that*

$$\begin{aligned}
\mathbf{L}_h^n &= \mathbf{L}_h^{\boldsymbol{f},\Delta t} + \mathbf{L}_h^{p_h^{n-1},\Delta t} + \mathbf{L}_h^{\boldsymbol{\lambda}^n,\Delta t},\\
\boldsymbol{u}_h^n &= \boldsymbol{u}_h^{\boldsymbol{f},\Delta t} + \boldsymbol{u}_h^{p_h^{n-1},\Delta t} + \boldsymbol{u}_h^{\boldsymbol{\lambda}^n,\Delta t},\\
p_h^n &= p_h^{\boldsymbol{f},\Delta t} + p_h^{p_h^{n-1},\Delta t} + p_h^{\boldsymbol{\lambda}^n,\Delta t},\\
\hat{\boldsymbol{u}}_h^n &= \boldsymbol{\lambda}^n,
\end{aligned} \tag{43}$$

where $\boldsymbol{\lambda}^n$ is the only function in $\boldsymbol{M}_h(\boldsymbol{g})$ satisfying

$$a_h^{\Delta t}(\boldsymbol{\lambda}^n, \boldsymbol{\mu}) = \ell_h^{\Delta t}(\boldsymbol{\mu}; p_h^{n-1}), \quad \forall \boldsymbol{\mu} \in \boldsymbol{M}_h(\mathbf{0}). \tag{44}$$

Here the forms are given by

$$\begin{aligned}
a_h^{\Delta t}(\boldsymbol{\eta}, \boldsymbol{\mu}) &= \left(\nu\mathbf{L}_h^{\boldsymbol{\eta},\Delta t}, \mathbf{L}_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathscr{T}_h} + \left\langle \mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta},\Delta t} - \boldsymbol{\eta}), (\boldsymbol{u}_h^{\boldsymbol{\mu},\Delta t} - \boldsymbol{\mu})\right\rangle_{\partial\mathscr{T}_h}\\
&\quad + \frac{1}{\Delta t}\left(p_h^{\boldsymbol{\eta},\Delta t}, p_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathscr{T}_h},\\
\ell_h^{\Delta t}(\boldsymbol{\mu}; p_h^{n-1}) &= (\boldsymbol{f}, \mathrm{u}_h^{\boldsymbol{\mu},\Delta t})_{\mathscr{T}_h} - \frac{1}{\Delta t}\left(p_h^{n-1}, p_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathscr{T}_h},
\end{aligned} \tag{45}$$

for all $\boldsymbol{\eta}, \boldsymbol{\mu} \in \boldsymbol{M}_h$.

We note that the bilinear form $a_h^{\Delta t}$ is similar to a_h defined in (11) of the HDG method of Section 2 except that $a_h^{\Delta t}$ has an additional term due to the pressure. Similarly, the functional $\ell_h^{\Delta t}$ has an additional term due to the pressure. These pressure terms result from introducing the time derivative into the continuity equation.

3.5. Implementation considerations

The characterization result in Theorem 3.1 allows for an efficient implementation of the HDG method, which we shall articulate as follows. First, for each element $K \in \mathscr{T}_h$, we compute the function $(\mathbf{L}_h^{\boldsymbol{f},\Delta t}, \boldsymbol{u}_h^{\boldsymbol{f},\Delta t}, p_h^{\boldsymbol{f},\Delta t})$ by solving the local solver (40); the function $(\mathbf{L}_h^{\varphi,\Delta t}, \mathrm{u}_h^{\varphi,\Delta t}, p_h^{\varphi,\Delta t})$ by solving the local solver (41) for all elements φ of a basis of $\mathscr{P}_k(K)$; and the function $(\mathbf{L}_h^{\boldsymbol{\eta},\Delta t}, \mathrm{u}_h^{\boldsymbol{\eta},\Delta t}, p_h^{\boldsymbol{\eta},\Delta t})$ by solving the local solver (42) for all elements $\boldsymbol{\eta}$ of a basis of $\mathscr{P}_k(F),\ \forall F \in \partial K$. We then need to compute $\boldsymbol{\lambda}^n$ for each time level n.

To compute $\boldsymbol{\lambda}^n$, we note that the matrix equation of the weak formulation (44) is of the form

$$\mathbb{A}\Lambda^n = \mathbb{F}^n \tag{46}$$

where Λ^n represents the degrees of freedom for $\boldsymbol{\lambda}^n$. The matrix $\mathbb{A}$ and vector $\mathbb{F}^n$ can be formed by the usual finite element assembly procedure once the elemental matrices and vectors are computed as follows.

Let $\boldsymbol{\mu}_i \in \mathscr{P}_k(\{F : F \in \partial K\}),\ 1 \leqslant i \leqslant N,$ where $N = (d+1)\dim \mathscr{P}_k(F)$, be the set of basis functions on the faces of the boundary ∂K of an element $K \in \mathscr{T}_h$. (Note that these basis functions are constructed from polynomials of degree k which are defined on the faces of ∂K.) The elemental matrix and vector are then given by

$$\begin{aligned}
&\mathbb{A}_{ij}^K = \left(\nu\mathbf{L}_K^{\boldsymbol{\mu}_i,\Delta t}, \mathbf{L}_K^{\boldsymbol{\mu}_j,\Delta t}\right)_K + \tfrac{1}{\Delta t}(p_K^{\boldsymbol{\mu}_i,\Delta t}, p_K^{\boldsymbol{\mu}_j,\Delta t})_K + \left\langle \mathbf{S}(\boldsymbol{u}_K^{\boldsymbol{\mu}_i,\Delta t} - \boldsymbol{\mu}_i), (\boldsymbol{u}_K^{\boldsymbol{\mu}_j,\Delta t} - \boldsymbol{\mu}_j)\right\rangle_{\partial K}, \quad 1 \leqslant i,j \leqslant N,\\
&\mathbb{F}_i^{K,n} = (\boldsymbol{f}, \boldsymbol{u}_K^{\boldsymbol{\mu}_i,\Delta t})_K - \tfrac{1}{\Delta t}(p_h^{n-1}, p_K^{\boldsymbol{\mu}_i,\Delta t})_K, \qquad 1 \leqslant i \leqslant N.
\end{aligned} \tag{47}$$

where $(\mathbf{L}_K^{\boldsymbol{\mu}_i,\Delta t}, \boldsymbol{u}_K^{\boldsymbol{\mu}_i,\Delta t}, p_K^{\boldsymbol{\mu}_i,\Delta t}) \in \mathbf{P}_k(K) \times \mathrm{P}_k(K) \times P_k(K)$ is the solution of the second local solver (42) on the element K for $\boldsymbol{\eta} := \boldsymbol{\mu}_i$.

To verify the stopping criterion (39) we need to update the pressure p_h^n as

$$p_h^n = p_h^{\boldsymbol{f},\Delta t} + p_h^{p_h^{n-1},\Delta t} + p_h^{\boldsymbol{\lambda}^n,\Delta t}, \tag{48}$$

where $p_h^{\boldsymbol{f},\Delta t}$, $p_h^{p_h^{n-1},\Delta t}$, and $p_h^{\boldsymbol{\lambda}^n,\Delta t}$ are calculated from (40), (41) for $\pi_h = p_h^{n-1}$, and (42) for $\boldsymbol{\eta} = \boldsymbol{\lambda}^n$, respectively. If (39) does not hold, we increase $n := n+1$, assemble the right-hand side of (46), solve the system (46) for Λ^n, and update the pressure p_h^n according to (48). When (39) holds, we terminate the process and compute

Table 1
Implementation of the HDG method via the augmented Lagrangian approach.

Implementation steps
Step 1. Given ε_{tol}, pick Δt, and set $p_h^0 = 0$ and $n := 1$
Step 2. For all of K of $\mathscr{T}_h$, compute $(\mathbf{L}_h^{\boldsymbol{f},\Delta t}, \boldsymbol{u}_h^{\boldsymbol{f},\Delta t}, p_h^{\boldsymbol{f},\Delta t})$ by solving (40); $(\mathbf{L}_h^{\varphi_j,\Delta t}, \boldsymbol{u}_h^{\varphi_j,\Delta t}, p_h^{\varphi_j,\Delta t})$ by solving (41) for $\pi_h = \varphi_j$ for all shape functions $\varphi_j \in \mathscr{P}_k(K)$, $1 \leqslant j \leqslant M$; and $(\mathbf{L}_K^{\boldsymbol{\mu}_i}, \boldsymbol{u}_K^{\boldsymbol{\mu}_i}, p_K^{\boldsymbol{\mu}_i})$ by solving (42) for $\boldsymbol{\eta} = \boldsymbol{\mu}_i$ for all shape functions $\boldsymbol{\mu}_i \in \mathscr{P}_k(\{F : F \in \partial K\}),\ 1 \leqslant i \leqslant N$
Step 3. Calculate the elemental matrix $\mathbb{A}^K$ according to (47) and form the stiffness matrix $\mathbb{A}$ by applying the finite element assembly procedure
Step 4. Calculate the elemental vector $\mathbb{F}^{K,n}$ according to (47) and form the vector $\mathbb{F}^n$ by applying the finite element assembly procedure
Step 5. Solve $\mathbb{A}\boldsymbol{\Lambda}^n = \mathbb{F}^n$, where $\boldsymbol{\Lambda}^n$ represents the degrees of freedom of $\boldsymbol{\lambda}^n$
Step 6. Compute p_h^n according to (48)
Step 7. If (39) does not hold, set $n := n+1$ and go to **Step 4**
Step 8. If it does, compute $(\mathbf{L}_h^n, \boldsymbol{u}_h^n)$ according to (43) and stop

$(\mathbf{L}_h^n, \boldsymbol{u}_h^n)$ from (43). The implementation of the HDG method is summarized in Table 1.

Finally, we point out the degrees of freedom and sparsity structure of the discrete system (46), restricting our attention to the case of a conforming triangulation $\mathscr{T}_h$ (no hanging nodes). It is clear that the matrix $\mathbb{A}$ has a block structure with square blocks of order equal to the dimension of $(d+1)\mathscr{P}_k(F)$ for each face F. The number of block rows and block columns is equal to the number of interior faces of the triangulation N_F. Furthermore, on each block row, there are at most $(2d+1)$ blocks that are not equal to

Table 2
History of convergence of the HDG method for $v\tau = h$.

Degree k	Mesh h^{-1}	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathscr{T}_h}$		$\|p-p_h\|_{\mathscr{T}_h}$		$\|\mathbf{L}-\mathbf{L}_h\|_{\mathscr{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^*\|_{\mathscr{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order
0	2	3.40e−0	–	1.16e−0	–	1.70e−9	–	3.81e−0	–
	4	4.10e−0	−0.27	5.42e−1	1.10	1.33e−9	0.35	4.19e−0	−0.14
	8	3.94e−0	0.06	3.41e−1	0.67	8.24e−0	0.70	3.96e−0	0.08
	16	3.87e−0	0.03	1.94e−1	0.81	4.60e−0	0.84	3.87e−0	0.03
	32	3.85e−0	0.01	1.11e−1	0.80	2.54e−0	0.85	3.85e−0	0.01
1	2	1.45e−0	–	8.75e−1	–	7.33e−0	–	4.44e−1	–
	4	6.84e−1	1.08	2.69e−1	1.70	2.57e−0	1.51	9.34e−2	2.25
	8	3.45e−1	0.99	7.34e−2	1.87	7.34e−1	1.81	1.39e−2	2.75
	16	1.73e−1	1.00	1.86e−2	1.98	1.95e−1	1.92	1.92e−3	2.86
	32	8.65e−2	1.00	4.65e−3	2.00	4.99e−2	1.96	2.53e−4	2.92
2	2	3.20e−1	–	2.07e−1	–	1.97e−0	–	9.00e−2	–
	4	8.23e−2	1.96	3.38e−2	2.62	3.13e−1	2.65	8.09e−3	3.48
	8	2.13e−2	1.95	4.59e−3	2.88	4.38e−2	2.84	5.51e−4	3.88
	16	5.38e−3	1.98	5.86e−4	2.97	5.66e−3	2.95	3.54e−5	3.96
	32	1.35e−3	2.00	7.34e−5	3.00	7.15e−4	2.98	2.24e−6	3.99

Table 3
History of convergence of the HDG method for $v\tau = 1$.

Degree k	Mesh h^{-1}	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathscr{T}_h}$		$\|p-p_h\|_{\mathscr{T}_h}$		$\|\mathbf{L}-\mathbf{L}_h\|_{\mathscr{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^*\|_{\mathscr{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order
0	2	2.06e−0	–	1.35e−0	–	1.47e−9	–	2.60e−0	–
	4	1.56e−0	0.40	5.75e−1	1.23	1.05e−9	0.48	1.67e−0	0.64
	8	7.19e−1	1.12	4.82e−1	0.25	6.75e−0	0.64	7.46e−1	1.16
	16	3.34e−1	1.10	2.66e−1	0.86	4.14e−0	0.71	3.40e−1	1.14
	32	1.58e−1	1.08	1.44e−1	0.89	2.45e−0	0.76	1.59e−1	1.10
1	2	9.55e−1	–	9.36e−1	–	6.97e−0	–	4.17e−1	–
	4	2.51e−1	1.93	2.87e−1	1.71	2.34e−0	1.57	8.92e−2	2.22
	8	6.61e−2	1.93	7.85e−2	1.87	7.48e−1	1.65	1.47e−2	2.60
	16	1.62e−2	2.03	2.01e−2	1.97	2.08e−1	1.85	2.11e−3	2.80
	32	3.98e−3	2.02	5.04e−3	1.99	5.51e−2	1.92	2.86e−4	2.89
2	2	2.31e−1	–	2.27e−1	–	2.12e−0	–	9.53e−2	–
	4	3.47e−2	2.74	3.77e−2	2.59	3.50e−1	2.60	9.98e−3	3.26
	8	4.21e−3	3.04	5.10e−3	2.89	4.89e−2	2.84	6.79e−4	3.88
	16	5.26e−4	3.00	6.50e−4	2.97	6.56e−3	2.90	4.56e−5	3.90
	32	6.54e−5	3.01	8.14e−5	3.00	8.49e−4	2.95	2.96e−6	3.94

Table 4
History of convergence of the HDG method for $v\tau = 1/h$.

Degree k	Mesh h^{-1}	$\|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathscr{T}_h}$		$\|p-p_h\|_{\mathscr{T}_h}$		$\|\mathbf{L}-\mathbf{L}_h\|_{\mathscr{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^*\|_{\mathscr{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order
0	2	1.51e−0	–	1.77e−0	–	1.32e−9	–	2.13e−0	–
	4	1.11e−0	0.44	1.08e−0	0.71	8.78e−0	0.59	1.19e−0	0.84
	8	5.30e−1	1.07	2.32e−0	−1.11	5.72e−0	0.62	5.45e−1	1.12
	16	3.88e−1	0.45	2.53e−0	−0.12	4.76e−0	0.27	3.91e−1	0.48
	32	3.44e−1	0.18	2.64e−0	−0.06	4.39e−0	0.11	3.44e−1	0.18
1	2	7.66e−1	–	1.09e−0	–	7.08e−0	–	4.45e−1	–
	4	2.04e−1	1.91	4.75e−1	1.19	2.88e−0	1.30	1.34e−1	1.74
	8	5.67e−2	1.84	2.00e−1	1.25	1.75e−0	0.72	3.83e−2	1.80
	16	1.43e−2	1.99	9.25e−2	1.11	8.85e−1	0.98	9.80e−3	1.97
	32	3.60e−3	1.99	4.39e−2	1.08	4.46e−1	0.99	2.49e−3	1.98
2	2	2.10e−1	–	2.84e−1	–	2.50e−0	–	1.14e−1	–
	4	3.23e−2	2.70	7.20e−2	1.98	6.13e−1	2.03	1.96e−2	2.55
	8	3.82e−3	3.08	1.62e−2	2.15	1.40e−1	2.13	2.33e−3	3.07
	16	4.85e−4	2.98	3.68e−3	2.14	3.56e−2	1.97	3.03e−4	2.94
	32	6.12e−5	2.99	8.65e−4	2.09	9.02e−3	1.98	3.89e−5	2.96

Table 5
Comparison of the hybridized globally divergence-free LDG method [4] and the HDG method for $v\tau = 1$.

Degree k	Mesh h^{-1}	Hybridized LDG method [4]				HDG method					
		$\|\|\boldsymbol{u}-\boldsymbol{u}_h\|\|_{\mathcal{F}_h}$		$\|\|p-p_h\|\|_{\mathcal{F}_h}$		$\|\|p-p_h\|\|_{\mathcal{F}_h}$		$\|\|\boldsymbol{u}-\boldsymbol{u}_h\|\|_{\mathcal{F}_h}$		$\|\|\boldsymbol{u}-\boldsymbol{u}_h^*\|\|_{\mathcal{F}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
1	2	7.30e−1	–	2.20e−0	–	9.36e−1	–	9.55e−1	–	6.41e−1	–
	4	3.90e−1	0.93	1.30e−0	0.70	2.87e−1	1.71	2.51e−1	1.93	1.36e−1	2.24
	8	6.60e−2	2.50	7.30e−1	0.90	7.85e−2	1.87	6.61e−2	1.93	2.03e−2	2.74
	16	1.40e−2	2.25	3.70e−1	0.97	2.01e−2	1.97	1.62e−2	2.03	2.76e−3	2.88
	32	3.10e−3	2.10	1.80e−1	0.99	5.04e−3	1.99	3.98e−3	2.02	3.62e−4	2.93
2	2	2.90e−1	–	7.40e−1	–	2.27e−1	–	2.31e−1	–	1.22e−1	–
	4	7.00e−2	2.00	2.40e−1	1.59	3.77e−2	2.59	3.47e−2	2.74	1.18e−2	3.38
	8	1.20e−2	2.61	6.70e−2	1.86	5.10e−3	2.89	4.21e−3	3.04	8.08e−4	3.86
	16	1.70e−3	2.71	1.70e−2	1.96	6.50e−4	2.97	5.26e−4	3.00	5.33e−5	3.92
	32	2.40e−4	2.87	4.30e−3	1.99	8.14e−5	3.00	6.45e−5	3.01	3.42e−6	3.96

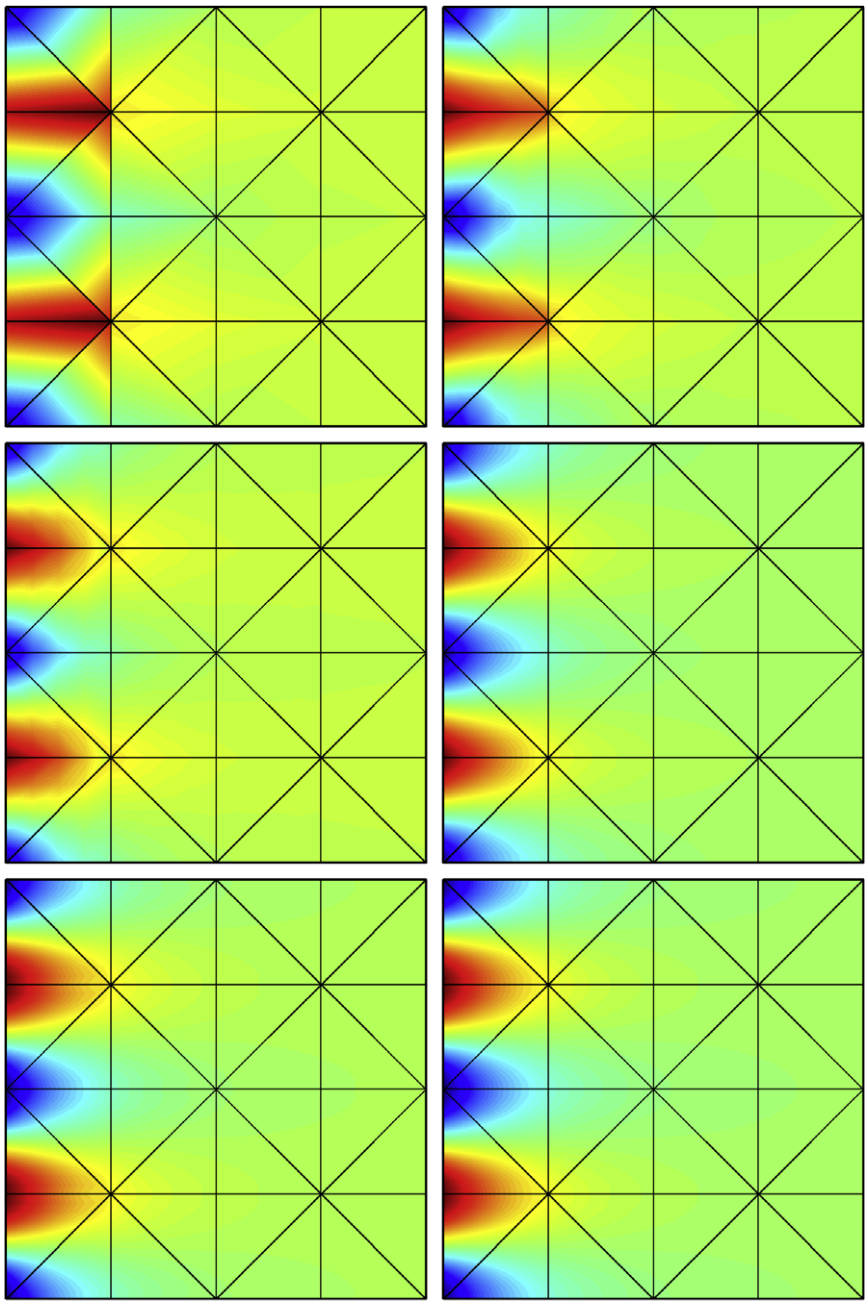

Fig. 1. Horizontal component of the original velocity (left) and the postprocessed velocity (right) for polynomial degree $k = 1$ (top), $k = 2$ (middle) and $k = 3$ (bottom) over the original mesh $\ell = 0$.

zero. Hence, the size of the matrix $\mathbb{A}$ is $N_A \times N_A$, where $N_A = N_F(d+1)\ \dim \mathscr{P}_k(F)$.

4. Numerical experiments

We consider the Stokes problem whose exact solution coincides with the analytical solution of the incompressible Navier–Stokes equations obtained by Kovasznay in [14], namely,

$$\mathbf{u}_1 = 1 - \exp(\lambda x_1)\cos(2\pi x_2),$$
$$\mathbf{u}_2 = \frac{\lambda}{2\pi}\exp(\lambda x_1)\sin(2\pi x_2),$$
$$p = \frac{1}{2}\exp(2\lambda x_1),$$

where $\lambda = \frac{Re}{2} - \sqrt{\frac{Re^2}{4} + 4\pi^2}$ and $Re = \frac{1}{\nu}$ is the Reynolds number. The Kovasznay flow is also a solution of the Stokes problem (1) with the source term $\boldsymbol{f} = -(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}$. We take Dirichlet boundary conditions for the velocity as the restriction of the exact solution to the domain boundary. Here the computational domain is $\Omega = (0,2)\times(-0.5,1.5)$ and $\nu = 0.1$ so that the Reynolds number is $Re = 10$.

In our experiments, we consider meshes that are refinements of a uniform mesh of 32 ($h = 1/2$) congruent triangles. Each refinement is obtained by subdividing each triangle into four congruent triangles. We say that the mesh has level ℓ ($h = 1/2^{\ell+1}$) if it is obtained from the original mesh by ℓ of these refinements. On these meshes, we consider polynomials of degree k to represent all the approximate variables using a nodal basis within each element, with the nodes uniformly distributed. The numerical example

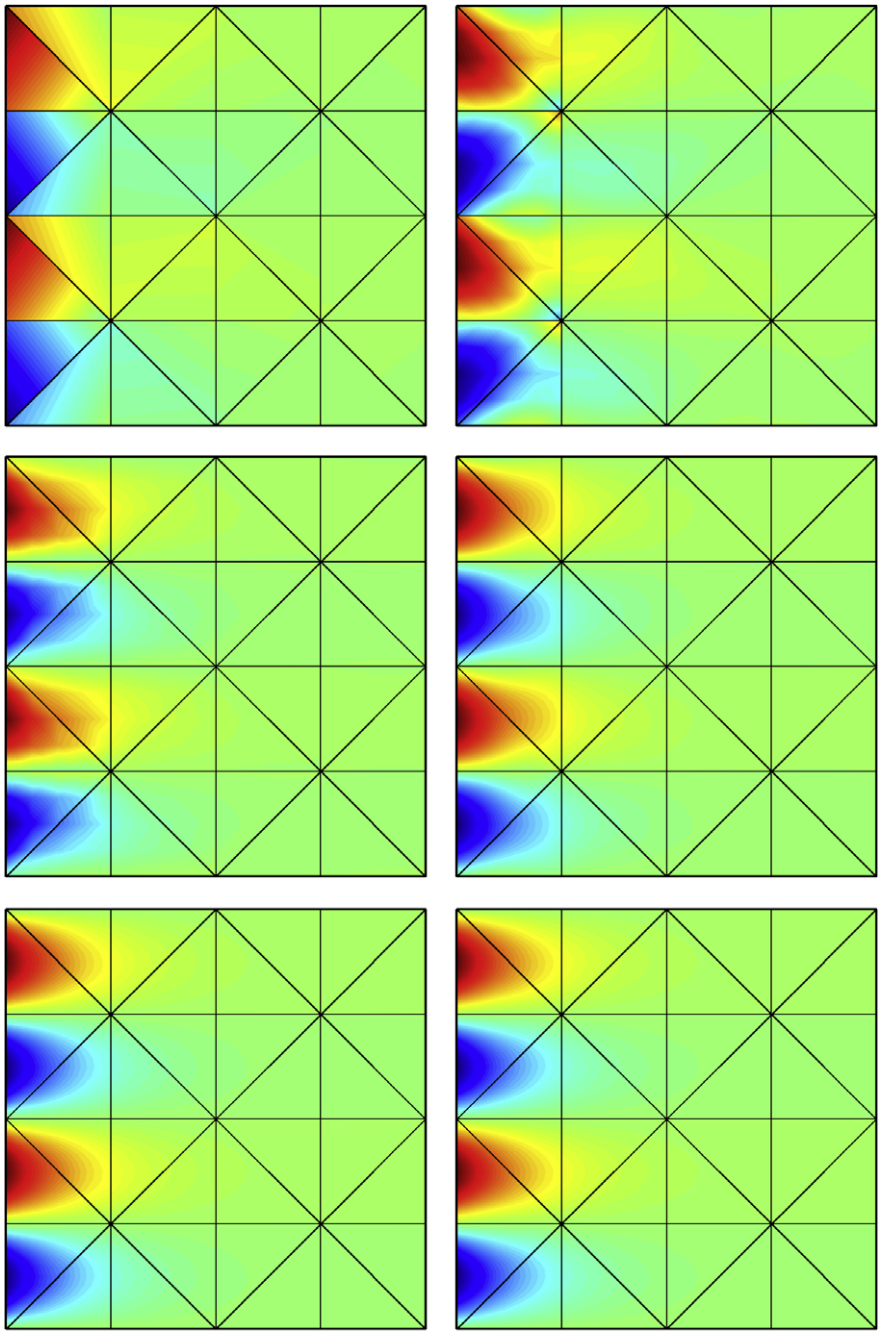

Fig. 2. Vertical component of the original velocity (left) and the postprocessed velocity (right) for polynomial degree $k = 1$ (top), $k = 2$ (middle) and $k = 3$ (bottom) over the original mesh $\ell = 0$.

and meshes are taken from [4] to permit comparison. In all test cases, the stabilization tensor $\mathbf{S}$ is chosen as

$$\mathbf{S} = \nu \begin{pmatrix} \tau & 0 \\ 0 & \tau \end{pmatrix},$$

where τ is a positive constant function defined on $\mathscr{E}_h$. In our implementation, we set $\varepsilon_{\text{tol}} = 10^{-8}$ for the error tolerance.

Below we present numerical results to assess the convergence and accuracy of the HDG method. We first explore the effect of the stabilization parameter τ on the convergence of the HDG method and compare the results of the HDG method with those of the hybridized globally divergence-free LDG method [4]. We then demonstrate the effectiveness of the local postprocessing in improving the approximate solution. Finally, we study the effect of the artificial time step Δt on the condition number of the discrete matrix and the required number of iterations.

4.1. Convergence of the HDG method

We first present the convergence results of the HDG method in Table 2 for $\nu\tau = h$, Table 3 for $\nu\tau = 1$, and Table 4 for $\nu\tau = 1/h$. We can clearly see the effect of the stabilization parameter τ on the accuracy and convergence of the numerical solution. In particular, when the stabilization parameter is chosen as $\nu\tau = h$ the approximate velocity converges with the suboptimal order k, while both the approximate pressure and gradient converge with the optimal order $k+1$. On the other hand, when we set $\nu\tau = 1/h$ the approximate velocity converges with the optimal order $k+1$; however, in

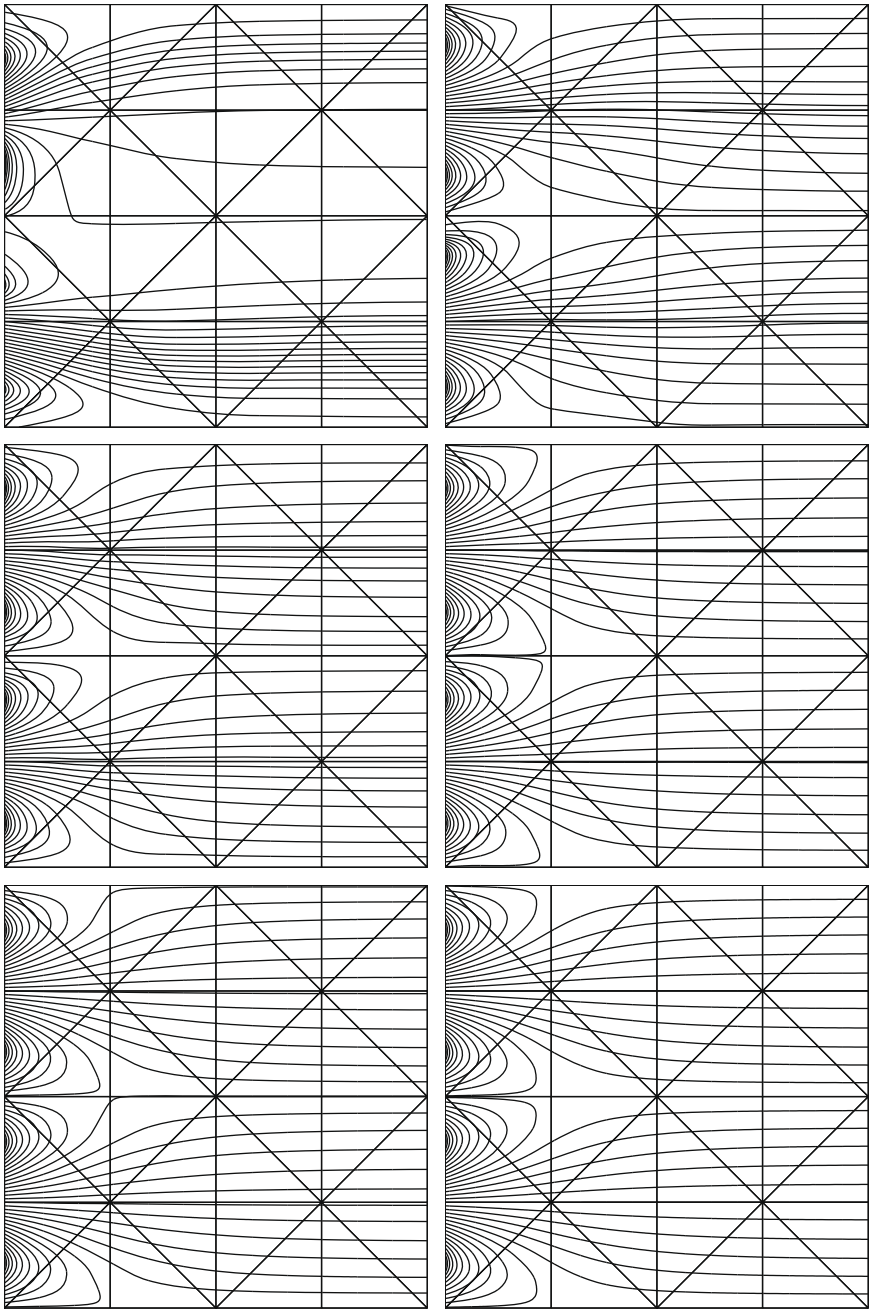

Fig. 3. Streamline of the original velocity (left) and the postprocessed velocity (right) for polynomial degree $k = 1$ (top), $k = 2$ (middle) and $k = 3$ (bottom) over the original mesh $\ell = 0$.

this case, both the approximate pressure and gradient seem to converge with order k. Setting the stabilization parameter to $v\tau = 1$, all the variables converge at the optimal rate of $k+1$. These results indicate that the optimal value of the stabilization parameter is in order of unity. It is interesting to note that the HDG method result in approximations which converge optimally for $k = 0$ when $v\tau = 1$, while some other DG methods such as the LDG method may not produce optimal convergent approximations in this case. Moreover, the approximate rotational tensor $\mathbf{W}_h := \mathbf{L}_h - (\mathbf{L}_h)^T$ converges with the same order $k+1$ as $\mathbf{L}_h$ for $v\tau = h$ and $v\tau = 1$.

We next plot the approximate solution for different meshes and polynomial degrees. Figs. 2–6 show the two components of the approximate velocity and the streamline over the original mesh $\ell = 0$ and the mesh $\ell = 1$ for different values of $k = 1$, 2, and 3. We see that the approximate solution can be significantly improved by increasing the polynomial degree or refining the mesh.

We now compare our results with those obtained the hybridized globally divergence-free LDG method [4]. We note that for other DG methods such as the LDG method [4,12] and Bassi–Rebay method [2,3] the approximate pressure, gradient, and vorticity converge suboptimally with order k. To wit, we display in Table 5 the error and order of convergence of the hybridized globally divergence-free LDG method and the HDG method with $v\tau = 1$ for the same Stokes problem on the same meshes. The results from this table are taken from [4]. We see that the approximate velocity of the HDG method has considerably smaller errors than that of the hybridized globally divergence-free LDG method although they both converge with the same order $k+1$. The approximate pressure of the HDG method converges with order $k+1$ (one order

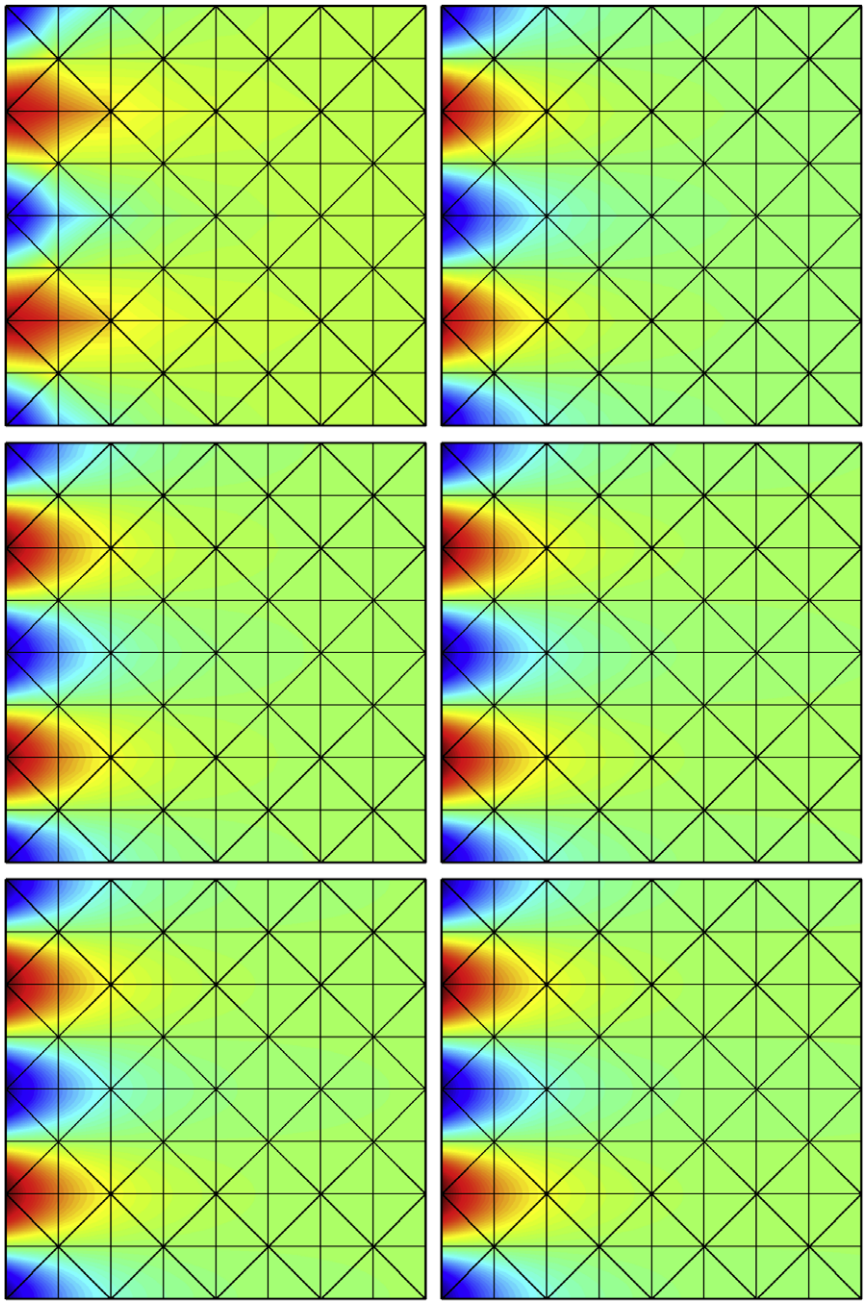

Fig. 4. Horizontal component of the original velocity (left) and the postprocessed velocity (right) for polynomial degree $k = 1$ (top), $k = 2$ (middle) and $k = 3$ (bottom) over the mesh $\ell = 1$.

higher) and thus has significantly smaller errors than that of the hybridized globally divergence-free LDG method. The postprocessed velocity of the HDG method for $k = 1$ has roughly the same magnitude errors as the approximate velocity of the hybridized globally divergence-free LDG method for $k = 2$. These results indicate that the HDG method can provide the same accuracy with far less computational cost.

4.2. Effectiveness of the local postprocessing

The error and order of convergence of the postprocessed velocity are also displayed in Tables 2–4. For the case $v\tau = 1/h$ the postprocessed velocity $\boldsymbol{u}_h^*$ appears to have the same order of convergence as the original velocity u_h. For the case $v\tau = 1$ we observe that $\boldsymbol{u}_h^*$ superconverges with order $k+2$ for $k \geqslant 1$, which is one order higher than $\boldsymbol{u}_h$. It is interesting to note that for the case $v\tau = h$ the postprocessed velocity superconverges with order $k+2$ for $k \geqslant 1$, while the original velocity converges with order k only. For $k = 0$, however, $\boldsymbol{u}_h^*$ converges with the same order as $\boldsymbol{u}_h$ in all cases.

Furthermore, we present in Fig. 1 the horizontal component, in Fig. 2 the vertical component, and in Fig. 3 the streamline of the original velocity and the postprocessed velocity over the original mesh $\ell = 0$. We also plot in Figs. 4–6 the same quantities over the mesh $\ell = 1$. In all these of figures, we see a clear improvement of the approximations as we increase from $k = 0$ to $k = 2$. Most

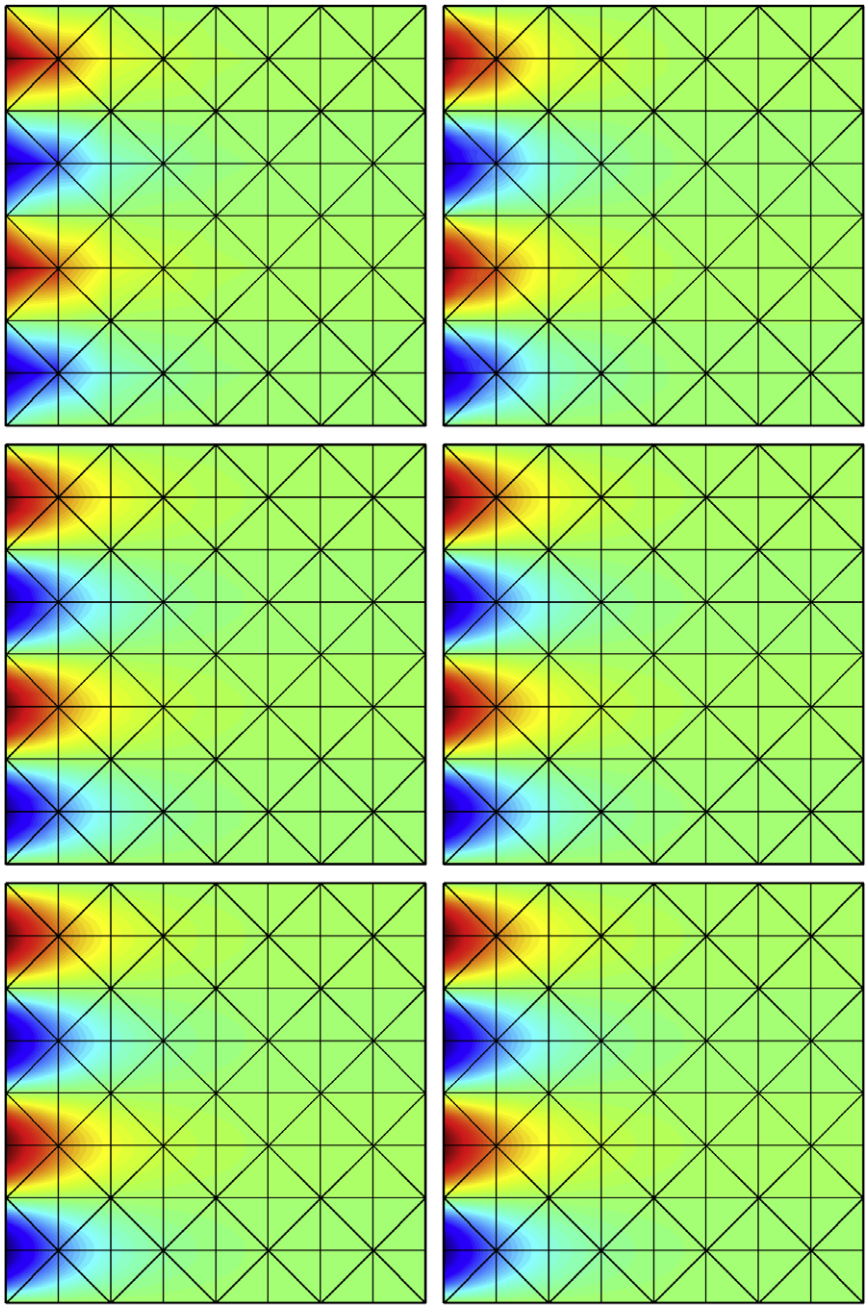

Fig. 5. Vertical component of the original velocity (left) and the postprocessed velocity (right) for polynomial degree $k = 1$ (top), $k = 2$ (middle) and $k = 3$ (bottom) over the mesh $\ell = 1$.

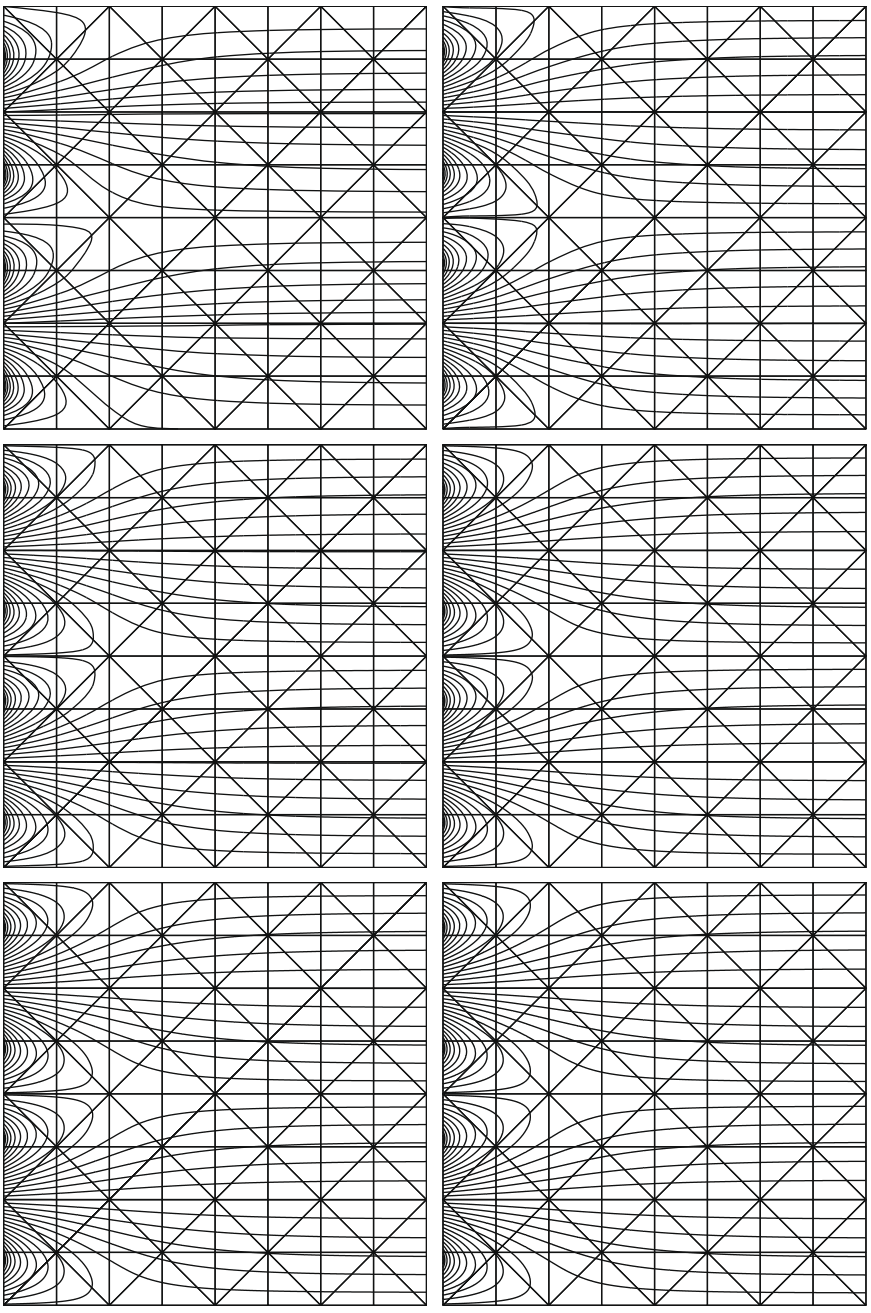

Fig. 6. Streamline of the original velocity (left) and the postprocessed velocity (right) for polynomial degree $k = 1$ (top), $k = 2$ (middle) and $k = 3$ (bottom) over the mesh $\ell = 1$.

notably, the local postprocessing improves the approximation of the velocity significantly for $k = 1$ and $k = 2$, since the postprocessed velocity $\boldsymbol{u}_h^*$ is clearly superior to the original velocity u_h.

4.3. Effect of the artificial time step

Finally, we examine how the artificial time step Δt affects the condition number of the matrix $\mathbb{A}$ and the number of iterations required to reach the error tolerance $\varepsilon_{\mathrm{tol}} = 10^{-8}$. (Recall that the solution procedure is started with zero pressure and terminated successfully when the relative error of the pressure is less than $\varepsilon_{\mathrm{tol}}$.) We define the condition number ratio R as

$$R = \frac{C}{(1 + \Delta t/\nu)(k+1)h^{-2}},$$

where C is the condition number of the matrix $\mathbb{A}$. Here the condition number is defined as the ratio of the largest singular value of $\mathbb{A}$ to the smallest singular value, which are computed by a singular value decomposition of $\mathbb{A}$.

We report in Tables 6 and 7 the condition number ratio R and the number of iterations for convergence, respectively, as a function of h and k for several values of Δt. In Table 6, we see that the ratio R remains remarkably close to 1/2. As a consequence, we have that the condition number of $\mathbb{A}$ is close to

$$\frac{1}{2}(1 + \Delta t/\nu)(k+1)h^{-2}.$$

In Table 7, we see that the number of iterations for convergence is relatively small and independent of the mesh size h and polynomial degree k. Hence, the augmented Lagrangian approach is attractive

Table 6
The condition number ratio R as a function of h, k, and Δt.

Degree k	Mesh h^{-1}	Artificial time step				
		$\Delta t = 1$	$\Delta t = 2$	$\Delta t = 4$	$\Delta t = 8$	$\Delta t = 16$
1	2	.54	.52	.50	.50	.49
	4	.51	.50	.50	.50	.50
	8	.48	.47	.47	.47	.47
	16	.47	.46	.46	.46	.46
	32	.46	.46	.46	.46	.46
2	2	.57	.54	.53	.52	.52
	4	.52	.50	.49	.49	.48
	8	.49	.48	.47	.47	.47
	16	.49	.47	.47	.46	.46
	32	.48	.47	.47	.46	.46

Table 7
The number of iterations required for convergence as a function of h, k, and Δt.

Degree k	Mesh h^{-1}	Artificial time step				
		$\Delta t = 1$	$\Delta t = 2$	$\Delta t = 4$	$\Delta t = 8$	$\Delta t = 16$
1	2	16	12	9	7	6
	4	16	12	9	7	6
	8	16	12	9	7	6
	16	16	12	9	8	6
	32	17	12	9	8	6
2	2	16	12	9	7	6
	4	16	12	9	7	6
	8	16	12	9	8	6
	16	17	12	9	8	6
	32	17	12	9	8	6

for solving the discrete Stokes system arising from the HDG discretization.

5. Conclusions

In this paper, we present a hybridizable DG (HDG) method for Stokes flow. We also use a local postprocessing to improve the numerical approximations. The main features of our approach and the results of our numerical experiments can be summarized as follows.

- All the approximate variables converge with the optimal order $k+1$ for an appropriate choice of the stabilization parameter, which in our particular example is of order unity. The approximate solution can be postprocessed to yield a new approximate velocity which converges with an additional order $k+2$ for $k \geqslant 1$. Note that these results are only observed for smooth problems.
- The approximate pressure and postprocessed velocity of the HDG method using polynomials of degree $k \geqslant 1$ have accuracy comparable to the approximate pressure and velocity of the hybridized globally divergence-free LDG method [4] using polynomials of degree $k+1$.
- Although the global coupled unknowns are the approximate trace of the velocity and the mean of pressure, the method can be better implemented by using the augmented Lagrangian approach since the mean of pressure is eliminated. Numerical results indicate that the number of iterations required for convergence is independent of both the mesh size and the polynomial degree.

The extension of this work to the incompressible Navier–Stokes equations constitute the subject of ongoing research. We end this paper by pointing out that the *a priori* error analysis of the HDG method presented here is provided in [10].

Appendix A. Proof of Lemma 2.1

Proof. We integrate by parts the local solver (4) to obtain

$$(\mathbf{L}_h^{\boldsymbol{f}}, \mathbf{G})_K + (\boldsymbol{u}_h^{\boldsymbol{f}}, \nabla \cdot \mathbf{G})_K = 0, \tag{49a}$$

$$(\nabla \cdot (-\nu \mathbf{L}_h^{\boldsymbol{f}} + p_h^{\boldsymbol{f}} \mathbf{I}), \boldsymbol{v})_K + \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{f}} - p_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{v} \right\rangle_{\partial K} = (\boldsymbol{f}, \mathrm{v})_K, \tag{49b}$$

$$-(\boldsymbol{u}_h^{\boldsymbol{f}}, \nabla q)_K = 0, \tag{49c}$$

$$\bar{p}_h^{\boldsymbol{f}} = 0, \tag{49d}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{P}_k(K) \times P_k(K)$ when we consider the data $(\boldsymbol{f}, \mathbf{0}, 0)$. Similarly, we obtain

$$(\mathbf{L}_h^{\boldsymbol{\eta}}, \mathbf{G})_K + (\boldsymbol{u}_h^{\boldsymbol{\eta}}, \nabla \cdot \mathbf{G})_K = \langle \boldsymbol{\eta}, \mathbf{G}\boldsymbol{n} \rangle_{\partial K}, \tag{50a}$$

$$(\nabla \cdot (-\nu \mathbf{L}_h^{\boldsymbol{\eta}} + p_h^{\boldsymbol{\eta}} \mathbf{I}), \boldsymbol{v})_K + \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{\eta}} + \hat{p}_h^{\boldsymbol{\eta}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{\eta}} - p_h^{\boldsymbol{\eta}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{v} \right\rangle_{\partial K} = 0, \tag{50b}$$

$$-(\boldsymbol{u}_h^{\boldsymbol{\eta}}, \nabla q)_K = -\langle \boldsymbol{\eta} \cdot \boldsymbol{n}, q - \bar{q} \rangle_{\partial K}, \tag{50c}$$

$$\bar{p}_h^{\boldsymbol{\eta}} = 0, \tag{50d}$$

for all $(\mathbf{G}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{P}_k(K) \times P_k(K)$ when we set the data to be $(\mathbf{0}, \boldsymbol{\eta}, 0)$. Moreover, we note that $(\mathbf{L}_h^{\bar{\psi}}, \boldsymbol{u}_h^{\bar{\psi}}, p_h^{\bar{\psi}}) = (\mathbf{0}, \mathbf{0}, \bar{\psi})$ for the data $(\mathbf{0}, \mathbf{0}, \bar{\psi})$. The first identity in (8) can be derived as follows:

$$\begin{aligned}
-\left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} &= -\left\langle -\nu \mathbf{L}_h^{\boldsymbol{f}} \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} - \left\langle p_h^{\boldsymbol{f}}, \boldsymbol{\mu} \cdot \boldsymbol{n} \right\rangle_{\partial \mathcal{T}_h} \\
&\quad - \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{f}} - p_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} \\
&= (\nu \mathbf{L}_h^{\boldsymbol{\mu}}, \mathbf{L}_h^{\boldsymbol{f}})_{\mathcal{T}_h} + (\mathrm{u}_h^{\boldsymbol{\mu}}, \nabla \cdot \nu \mathbf{L}_h^{\boldsymbol{f}})_{\mathcal{T}_h} \\
&\quad - \left(\nabla p_h^{\boldsymbol{f}}, \boldsymbol{u}_h^{\boldsymbol{\mu}} \right)_{\mathcal{T}_h} - \left\langle \bar{p}_h^{\boldsymbol{f}}, \boldsymbol{\mu} \cdot \boldsymbol{n} \right\rangle_{\partial \mathcal{T}_h} \\
&\quad - \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I} + \nu \mathbf{L}_h^{\mathrm{f}} - p_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h},
\end{aligned}$$

by (50a) with $\boldsymbol{\eta} := \boldsymbol{\mu}$ and $\mathbf{G} := \nu \mathbf{L}_h^{\boldsymbol{f}}$, and (50c) with $\boldsymbol{\eta} := \boldsymbol{\mu}$ and $q := p_h^{\boldsymbol{f}}$. Then

$$\begin{aligned}
-\left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} &= -(\mathrm{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}})_{\mathcal{T}_h} + (\nu \mathbf{L}_h^{\boldsymbol{\mu}}, \mathbf{L}_h^{\boldsymbol{f}})_{\mathcal{T}_h} \\
&\quad + \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I} + \nu \mathbf{L}_h^{\mathrm{f}} - p_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{u}_h^{\boldsymbol{\mu}} - \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h},
\end{aligned}$$

by (49b) with $\boldsymbol{v} := \boldsymbol{u}_h^{\boldsymbol{\mu}}$ and (49d). By (49a) with $\mathbf{G} := \nu \mathbf{L}_h^{\boldsymbol{\mu}}$ and (50b) with $\boldsymbol{v} := \boldsymbol{u}_h^{\mathrm{f}}$,

$$\begin{aligned}
-\left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} &= -(\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}})_{\mathcal{T}_h} - (\nabla \cdot (p_h^{\boldsymbol{\mu}} \mathbf{I}), \boldsymbol{u}_h^{\boldsymbol{f}})_{\mathcal{T}_h} \\
&\quad - \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{\mu}} + \hat{p}_h^{\boldsymbol{\mu}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{\mu}} - p_h^{\boldsymbol{\mu}} \mathbf{I}) \mathrm{n}, \boldsymbol{u}_h^{\boldsymbol{f}} \right\rangle_{\partial \mathcal{T}_h} \\
&\quad + \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{f}} + \hat{p}_h^{\boldsymbol{f}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{f}} - p_h^{\boldsymbol{f}} \mathbf{I}) \boldsymbol{n}, \mathrm{u}_h^{\boldsymbol{\mu}} - \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} \\
&= -(\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}})_{\mathcal{T}_h},
\end{aligned}$$

by (49c) with $q := p_h^{\boldsymbol{\mu}}$ and (7). The second identity in (8) can be derived as follows:

$$\begin{aligned}
-\left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{\eta}} + \hat{p}_h^{\boldsymbol{\eta}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} &= -\langle -\nu \mathbf{L}_h^{\boldsymbol{\eta}} \boldsymbol{n}, \boldsymbol{\mu} \rangle_{\partial \mathcal{T}_h} - \langle p_h^{\boldsymbol{\eta}}, \boldsymbol{\mu} \cdot \boldsymbol{n} \rangle_{\partial \mathcal{T}_h} \\
&\quad - \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{\eta}} + \hat{p}_h^{\boldsymbol{\eta}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{\eta}} - p_h^{\boldsymbol{\eta}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h} \\
&= (\nu \mathbf{L}_h^{\boldsymbol{\mu}}, \mathbf{L}_h^{\boldsymbol{\eta}})_{\mathcal{T}_h} + (\boldsymbol{u}_h^{\boldsymbol{\mu}}, \nabla \cdot \nu \mathbf{L}_h^{\boldsymbol{\eta}})_{\mathcal{T}_h} \\
&\quad - (\nabla p_h^{\boldsymbol{\eta}}, \boldsymbol{u}_h^{\boldsymbol{\mu}})_{\mathcal{T}_h} - \langle \bar{p}_h^{\boldsymbol{\eta}}, \boldsymbol{\mu} \cdot \boldsymbol{n} \rangle_{\partial \mathcal{T}_h} \\
&\quad - \left\langle (-\nu \hat{\mathbf{L}}_h^{\boldsymbol{\eta}} + \hat{p}_h^{\boldsymbol{\eta}} \mathbf{I} + \nu \mathbf{L}_h^{\boldsymbol{\eta}} - p_h^{\boldsymbol{\eta}} \mathbf{I}) \boldsymbol{n}, \boldsymbol{\mu} \right\rangle_{\partial \mathcal{T}_h},
\end{aligned}$$

by (50a) with $\boldsymbol{\eta} := \boldsymbol{\mu}$ and $\mathbf{G} := \nu\mathbf{L}_h^{\eta}$, and (50c) with $\boldsymbol{\eta} := \boldsymbol{\mu}$ and $q := p_h^{\eta}$. Then

$$
\begin{aligned}
-\left\langle(-\nu\hat{\mathbf{L}}_h^{\eta}+\hat{p}_h^{\eta}\mathbf{I})\boldsymbol{n},\boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h} &= (\nu\mathbf{L}_h^{\mu},\mathbf{L}_h^{\eta})_{\mathscr{T}_h}\\
&\quad+\left\langle(-\nu\hat{\mathbf{L}}_h^{\eta}+\hat{p}_h^{\eta}\mathbf{I}+\nu\mathbf{L}_h^{\eta}-p_h^{\eta}\mathbf{I})\boldsymbol{n},\mathsf{u}_h^{\mu}-\boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h}\\
&= (\nu\mathbf{L}_h^{\mu},\mathbf{L}_h^{\eta})_{\mathscr{T}_h}+\langle\mathbf{S}(\boldsymbol{u}_h^{\eta}-\boldsymbol{\eta}),\mathsf{u}_h^{\mu}-\boldsymbol{\mu}\rangle_{\partial\mathscr{T}_h},
\end{aligned}
$$

by (50b) with $\boldsymbol{\nu} := \boldsymbol{u}_h^{\mu}$, (50d), and (7). Finally, the last identity in (8) follows from the identity $(\mathbf{L}_h^{\psi},\boldsymbol{u}_h^{\psi},p_h^{\psi}) = (\mathbf{0},\mathbf{0},\bar{\psi})$. This completes the proof. □

Appendix B. Proof of Lemma 3.1

Proof. We first note that

$$(\delta_h^{p,0},q)_{\mathscr{T}_h} = (p_0-p_h,q)_{\mathscr{T}_h},$$

for all $q \in P_h$. Given the pressure error $\delta_h^{p,n-1}$ for $n \geqslant 1$, we have from (31) and (6) that $\delta_h^{p,n} \in P_h$ satisfies

$$\frac{1}{\Delta t}(\delta_h^{p,n},q)_{\mathscr{T}_h}-(\delta_h^{\mathsf{u},n},\nabla q)_{\mathscr{T}_h}+\langle\delta_h^{\hat{\mathsf{u}},n}\cdot\boldsymbol{n},q\rangle_{\partial\mathscr{T}_h}=\frac{1}{\Delta t}(\delta_h^{p,n-1},q)_{\mathscr{T}_h},\quad(51)$$

for all $q \in P_h$. Here $(\delta_h^{\mathbf{L},n},\delta_h^{\mathsf{u},n},\delta_h^{\dot{\boldsymbol{u}},n})$ is the element of $\mathbf{Y}_h\times\boldsymbol{V}_h\times\boldsymbol{M}_h(\mathbf{0})$ such that

$$
\begin{aligned}
&(\delta_h^{\mathbf{L},n},\mathbf{G})_{\mathscr{T}_h}+(\delta_h^{\boldsymbol{u},n},\nabla\cdot\mathbf{G})_{\mathscr{T}_h}-\langle\delta_h^{\hat{\mathsf{u}},n},\mathbf{G}\boldsymbol{n}\rangle_{\partial\mathscr{T}_h}=0,\\
&-(-\nu\,\delta_h^{\mathbf{L},n}+\delta_h^{p,n}\mathbf{I},\nabla\mathsf{v})_{\mathscr{T}_h}+\left\langle(-\nu\,\delta_h^{\hat{\mathbf{L}},n}+\delta_h^{\hat{p},n}\mathbf{I})\boldsymbol{n},\boldsymbol{\nu}\right\rangle_{\partial\mathscr{T}_h}=0,\\
&\left\langle(-\nu\,\delta_h^{\hat{\mathbf{L}},n}+\delta_h^{\hat{p},n}\mathbf{I})\boldsymbol{n},\boldsymbol{\mu}\right\rangle_{\partial\mathscr{T}_h}=0,
\end{aligned}\quad(52)
$$

for all $(\mathbf{G},\boldsymbol{\nu},q,\boldsymbol{\mu}) \in \mathbf{Y}_h\times\mathsf{V}_h\times\boldsymbol{M}_h(\mathbf{0})$, where

$$-\nu\,\delta_h^{\hat{\mathbf{L}},n}+\delta_h^{\hat{p},n}\mathbf{I}=-\nu\delta_h^{\mathbf{L},n}+\delta_h^{p,n}\mathbf{I}+\mathbf{S}(\delta_h^{\mathsf{u},n}-\delta_h^{\hat{\mathsf{u}},n})\otimes\boldsymbol{n}\quad\text{on }\partial\mathscr{T}_h.$$

Taking $q := \delta_h^{p,n}$ in (51), we obtain that

$$\frac{1}{\Delta t}(\delta_h^{p,n}-\delta_h^{p,n-1},\delta_h^{p,n})_{\mathscr{T}_h}+\Theta_h^n=0,$$

where $\Theta_h^n := -(\delta_h^{\boldsymbol{u},n},\nabla\delta_h^{p,n})_{\mathscr{T}_h}+\langle\delta_h^{\hat{\boldsymbol{u}},n}\cdot\boldsymbol{n},\delta_h^{p,n}\rangle_{\partial\mathscr{T}_h}$. Then application of the Cauchy–Schwartz inequality yields

$$\frac{1}{2\Delta t}(\|\delta_h^{p,n}\|^2-\|\delta_h^{p,n-1}\|^2)+\Theta_h^n\leqslant 0.$$

Moreover, by choosing $\mathbf{G} = \nu\delta_h^{\mathbf{L},n}$, $\boldsymbol{\nu} = \delta_h^{\boldsymbol{u},n}$, and $\boldsymbol{\mu} = \delta_h^{\hat{\boldsymbol{u}},n}$ in (52) and summing the three equations up, we obtain

$$\Theta_h^n = \nu(\delta_h^{\mathbf{L},n},\delta_h^{\mathbf{L},n})_{\mathscr{T}_h}+\langle\mathbf{S}(\delta_h^{\boldsymbol{u},n}-\delta_h^{\hat{\boldsymbol{u}},n}),(\delta_h^{\boldsymbol{u},n}-\delta_h^{\hat{\boldsymbol{u}},n})\rangle_{\partial\mathscr{T}_h}.$$

The desired result follows from the last two equations. This completes the proof. □

References

[1] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (5) (2001) 1749–1779.

[2] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, J. Comput. Phys. 131 (1997) 267–279.

[3] F. Bassi, A. Crivellini, D.A. Di Pietro, S. Rebay, An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations, J. Comput. Phys. 218 (2) (2006) 794–815.

[4] J. Carrero, B. Cockburn, D. Schötzau, Hybridized globally divergence-free LDG methods. I. The Stokes problem, Math. Comput. 75 (2006) 533–563.

[5] B. Cockburn, B. Dong, J. Guzmán, A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems, Math. Comput. 32 (2) (2007) 233–262.

[6] B. Cockburn, J. Gopalakrishnan, Incompressible finite elements via hybridization. Part I: the Stokes system in two space dimensions, SIAM J. Numer. Anal. 43 (4) (2005) 1627–1650.

[7] B. Cockburn, J. Gopalakrishnan, Incompressible finite elements via hybridization. Part II: the Stokes system in three space dimensions, SIAM J. Numer. Anal. 43 (4) (2005) 1651–1672.

[8] B. Cockburn, J. Gopalakrishnan, The derivation of hybridizable discontinuous Galerkin methods for Stokes flow, SIAM J. Numer. Anal. 47 (2009) 1092–1125.

[9] B. Cockburn, J. Gopalakrishnan, R. Lazarov, Unified hybridization of discontinuous Galerkin, mixed and continuous Galerkin methods for second-order elliptic problems, SIAM J. Numer. Anal. 47 (2009) 1319–1365.

[10] B. Cockburn, J. Gopalakrishnan, N.C. Nguyen, J. Peraire, F.-J. Sayas, Analysis of an HDG method for Stokes flow, Math. Comput., submitted for publication.

[11] B. Cockburn, J. Guzmán, H. Wang, Superconvergent discontinuous Galerkin methods for second-order elliptic problems, Math. Comput. 78 (2009) 1–24.

[12] B. Cockburn, G. Kanschat, D. Schötzau, C. Schwab, Local discontinuous Galerkin methods for the Stokes system, SIAM J. Numer. Anal. 40 (1) (2002) 319–343.

[13] M. Fortin, R. Glowinski, Augmented Lagrangian methods, Studies in Mathematics and Its Applications, vol. 15, North-Holland Publishing Co., Amsterdam, 1983. Applications to the numerical solution of boundary value problems, Translated from the French by B. Hunt and D.C. Spicer.

[14] L.I.G. Kovasznay, Laminar flow behind two-dimensional grid, Proc. Cambridge Philos. Soc. 44 (1948) 58–62.

[15] N.C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for linear convection–diffusion equations, J. Comput. Phys. 228 (2009) 3232–3254.

[16] N.C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection–diffusion equations, J. Comput. Phys. 228 (2009) 8841–8855.

[17] Rolf Stenberg, Some new families of finite elements for the Stokes equations, Numer. Math. 56 (1990) 827–838.