J Sci Comput (2010) 45: 215–237
DOI 10.1007/s10915-010-9359-0

A Comparison of HDG Methods for Stokes Flow

B. Cockburn · N.C. Nguyen · J. Peraire

Received: 1 June 2009 / Revised: 21 December 2009 / Accepted: 22 February 2010 /
Published online: 7 March 2010

Abstract In this paper, we compare hybridizable discontinuous Galerkin (HDG) methods for numerically solving the velocity-pressure-gradient, velocity-pressure-stress, and velocity-pressure-vorticity formulations of Stokes flow. Although they are defined by using different formulations of the Stokes equations, the methods share several common features. First, they use polynomials of degree k for all the components of the approximate solution. Second, they have the same globally coupled variables, namely, the approximate trace of the velocity on the faces and the mean of the pressure on the elements. Third, they give rise to a matrix system of the same size, sparsity structure and similar condition number. As a result, they have the same computational complexity and storage requirement. And fourth, they can provide, by means of an element-by element postprocessing, a new approximation of the velocity which, unlike the original velocity, is divergence-free and $\boldsymbol{H}(\mathrm{div})$-conforming. We present numerical results showing that each of the approximations provided by these three methods converge with the optimal order of $k+1$ in L^2 for any $k \geq 0$. We also display experiments indicating that the postprocessed velocity is a better approximation than the original approximate velocity. It converges with an additional order than the original velocity for the gradient-based HDG, and with the same order for the vorticity-based HDG methods. For the stress-based HDG methods, it seems to converge with an additional order for even polynomial degree approximations. Finally, the numerical results indicate that the method based on the velocity-pressure-gradient formulation provides the best approximations for similar computational complexity.

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

B. Cockburn
School of Mathematics, University of Minnesota, Minneapolis, MN 55455, USA

N.C. Nguyen (✉) · J. Peraire
Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
e-mail: cuongng@mit.edu

1 Introduction

In this paper, we compare three hybridizable discontinuous Galerkin (HDG) methods for numerically solving the Stokes system

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

where $\int_{\partial\Omega} \boldsymbol{g} \cdot \boldsymbol{n} = 0$. Here Ω is a bounded domain in $\mathbb{R}^d$ with Lipschitz boundary $\partial\Omega$. The HDG methods we consider are a vorticity-based HDG method introduced in [6], the gradient-based HDG method in [16] and a stress-based HDG method introduced in this paper.

To better describe our results, let us briefly review the previous work on hybridizable DG methods for Stokes flow. Hybridization for DG methods for Stokes flow was initially introduced in [3] as a technique that allows for the use of globally divergence-free velocity spaces without having to actually carry out their almost-impossible construction. A *velocity-pressure-vorticity* formulation in two-space dimensions was used and polynomials of degree k were taken for all the components of the velocity and polynomials of degree $k-1$ for the both the pressure and the vorticity. Optimal orders of convergence were proven for the vorticity, pressure and velocity.

The technique was then further developed, with a similar intention, for a mixed method for a *velocity-pressure-vorticity* formulation in two [4] and three space dimensions [5]. Indeed, a novel, global formulation for the method was obtained solely in terms of the tangential velocity and the pressure on the borders of the elements. As a further development of this approach, the first HDG method for the Stokes equations was introduced in [6] again for the *velocity-pressure-vorticity* formulation. This *vorticity-based* HDG method is one of the three HDG methods we shall compare in this paper.

Another is the HDG method based on the *velocity-pressure-gradient* formulation of the Stokes system proposed in [16]. This *gradient-based* HDG method was analyzed in [8] where it was proved that the approximate velocity, pressure, and gradient converge with the optimal order $k+1$ in L^2-norm for any $k \geq 0$. Moreover, it was shown how to obtain a new divergence-free approximate velocity lying in $\boldsymbol{H}(\mathrm{div})$ by means of an element-by-element postprocessing. The postprocessed velocity was then proven to converge with order $k+2$ for $k \geq 1$ and with order 1 for $k = 0$.

Note that there are DG methods which provide velocities that are divergence-free inside each of the element; however, the do not lie on $\boldsymbol{H}(\mathrm{div})$ since their normal component has no interelement continuity. Examples are the first DG method proposed for the Stokes system [1] and, more recently, and the DG method proposed in [14]. Note also that there are DG methods that do provide velocities that are divergence-free and belong to $\boldsymbol{H}(\mathrm{div})$. The first family DG methods with this property were introduced in [9] (see also the particular cases developed later in [10] and in [19]) and the DG method proposed in [3]. However, their velocities converge with order at most $k+1$ for $k \geq 1$.

The last HDG method we consider is introduced in this paper for the *velocity-pressure-stress* formulation of the Stokes system. This stress-based HDG method is an extension of the above-mentioned HDG methods to the velocity-pressure-stress formulation. It can also considered to be a variation of the HDG method proposed in [17] for compressible, linearly elastic bodies.

In this paper, we compare the above-mentioned vorticity-based, gradient-based, and stress-based HDG methods. Although these HDG methods are used to discretize three different formulations of the Stokes equations, they share several common features. Indeed, we show that when they use polynomials of degree k for all the components of the approximate solution, convergence with the optimal order of $k+1$ in L^2 for any $k \geq 0$ is achieved; this was known, as pointed out above, for the gradient-based HDG method, but was not known for the vorticity-based and the stress-based HDG methods. Second, we show that these three methods can have the same globally coupled variables, namely, the approximate trace of the velocity on the faces and the mean of the pressure on the elements. Third, we show that they give rise to a matrix system of the same size, structure and condition number. As a result, they have the same computational complexity and storage requirements, at least for the globally coupled unknowns. And fourth, we show that their solution can be postprocessed in an element-by element fashion [8] to yield a new approximation of the velocity. Unlike the original velocity, the postprocessed velocity is divergence-free and $\boldsymbol{H}(\mathrm{div})$-conforming.

Our numerical experiments show that the approximations provided by the gradient-based HDG method are more accurate than those provided by the stress-based and vorticity-based HDG methods. Moreover, they also show that the postprocessed velocity converges with order $k+2$ for the gradient-based HDG method when $k \geq 1$ and with order $k+1$ for the vorticity-based HDG method. The stress-based HDG method yields the postprocessed velocity which converges with order $k+1$ for $k=0,1$, but tends to converge with order $k+2$ for even order polynomial degrees $k \geq 2$. Since the three methods have a similar computational complexity, we conclude that the gradient-based HDG method is the method of choice.

Let us note that, compared to all known DG methods for the Stokes equations, the three HDG methods considered here have the following advantages. First, they result in the matrix system of smaller size and compactness since the globally coupled unknowns are defined on *the borders of the elements* and connected through neighboring elements only. Second, their numerical approximations converge with the optimal order $k+1$, whereas all other methods display the suboptimal order of convergence of k for the approximate gradient and pressure. And third, the HDG methods are somewhat simpler to implement.

The paper is organized as follows. In Sect. 2 we introduce the HDG methods for solving the Stokes system and define the element-by-element postprocessing to compute a new approximation of the velocity. In Sect. 3 we detail the implementation of the HDG methods. In Sect. 4 we present numerical results to assess the performance of the methods. Finally, in Sect. 5 we end with some concluding remarks.

2 The HDG Methods

2.1 Notation

Our notation is the one used in previous work [8, 16]. We denote by $\mathcal{T}_h$ a collection of disjoint regular elements K that partition Ω and set $\partial\mathcal{T}_h := \{\partial K : K \in \mathcal{T}_h\}$. For an element K of the collection $\mathcal{T}_h$, $F = \partial K \cap \partial\Omega$ is the boundary face if the $d-1$ Lebesgue measure of F is nonzero. For two elements K^+ and K^- of the collection $\mathcal{T}_h$, $F = \partial K^+ \cap \partial K^-$ is the interior face between K^+ and K^- if the $d-1$ Lebesgue measure of F is nonzero. We denote by $\mathcal{E}_h^o$ and $\mathcal{E}_h^\partial$ the set of interior and boundary faces, respectively. We set $\mathcal{E}_h = \mathcal{E}_h^o \cup \mathcal{E}_h^\partial$.

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 $F \in \mathcal{E}_h^o$, we set

$$[\![\mathbf{G}\boldsymbol{n}]\!] = \mathbf{G}^+\boldsymbol{n}^+ + \mathbf{G}^-\boldsymbol{n}^-,$$
$$[\![\boldsymbol{v} \odot \boldsymbol{n}]\!] = \boldsymbol{v}^+ \odot \boldsymbol{n}^+ + \boldsymbol{v}^- \odot \boldsymbol{n}^-,$$
$$[\![q\boldsymbol{n}]\!] = q^+\boldsymbol{n}^+ + q^-\boldsymbol{n}^-.$$

Here $\odot$ can be one of $\cdot$, $\times$, and $\otimes$; in this order, they denote the usual dot product, cross product, and tensor product. For $F \in \mathcal{E}_h^\partial$, the set of boundary edges on which $\mathbf{G}$, $\boldsymbol{v}$ and q are single valued, we set

$$[\![\mathbf{G}\boldsymbol{n}]\!] = \mathbf{G}\boldsymbol{n},$$
$$[\![\boldsymbol{v} \odot \boldsymbol{n}]\!] = \boldsymbol{v} \odot \boldsymbol{n},$$
$$[\![q\boldsymbol{n}]\!] = q\boldsymbol{n},$$

where $\boldsymbol{n}$ is the unit outward normal to $\partial\Omega$.

Let $\mathcal{P}_k(D)$ 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 $\boldsymbol{\mathcal{P}}_k(D) = [\mathcal{P}_k(D)]^d$, $\mathbf{P}_k(D) = [\mathcal{P}_k(D)]^{d\times d}$, $\boldsymbol{L}^2(D) = [L^2(D)]^d$, and $\boldsymbol{\mathfrak{L}}^2(D) = [L^2(D)]^{d(d+1)/2}$. We introduce discontinuous finite element approximation spaces for the gradient, velocity, and pressure as

$$\mathbf{G}_h = \{\mathbf{G} \in \mathbf{L}^2(\mathcal{T}_h) \,:\, \mathbf{G}|_K \in \mathbf{P}_k(K),\ \forall\, K \in \mathcal{T}_h\},$$
$$\boldsymbol{\Sigma}_h = \{\boldsymbol{\varsigma} \in \mathbf{G}_h \,:\, \boldsymbol{\varsigma} \text{ is symmetric}\},$$
$$\boldsymbol{V}_h = \{\boldsymbol{v} \in \boldsymbol{L}^2(\mathcal{T}_h) \,:\, \boldsymbol{v}|_K \in \boldsymbol{\mathcal{P}}_k(K),\ \forall K \in \mathcal{T}_h\},$$
$$P_h = \{q \in L^2(\mathcal{T}_h) \,:\, q|_K \in \mathcal{P}_k(K),\ \forall K \in \mathcal{T}_h\}.$$

To define the approximation space for the vorticity we have to distinguish between two and three dimensions. In the two dimensional case, $\mathbf{w} = \partial u_2/\partial x_1 - \partial u_1/\partial x_2$, we define $\mathbf{P}_k(K) := \mathcal{P}_k(K)$ and $\mathbf{W}_h := P_h$. In the three-dimensional case, $\mathbf{w} = (\partial u_3/\partial x_2 - \partial u_2/\partial x_3, \partial u_1/\partial x_3 - \partial u_3/\partial x_1, \partial u_2/\partial x_1 - \partial u_1/\partial x_2)$, we define $\mathbf{P}_k(K) := \boldsymbol{\mathcal{P}}_k(K)$ and $\mathbf{W}_h := \boldsymbol{V}_h$.

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

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

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) $F \in \mathcal{E}_h$ and discontinuous at their borders. We further denote by $\overline{\Psi}_h$ the set of functions in $L^2(\partial\mathcal{T}_h)$ that are constant on each ∂K for all elements K

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

The mean of our approximate pressure will belong to this space. The mean of a given function q in $L^2(\partial\mathcal{T}_h)$ is denoted $\overline{q}$ and is set to be on the boundary ∂K of an element K equal to $\overline{q}|_{\partial K} = \frac{1}{|\partial K|}\int_{\partial K} q$.

Finally, we define various inner products for our finite element spaces. We write $(\eta,\zeta)_{\mathcal{T}_h} := \sum_{K\in\mathcal{T}_h}(\eta,\zeta)_K$, where $(\eta,\zeta)_D$ denotes the integral of $\eta\,\zeta$ over the domain $D\subset\mathbb{R}^d$. We also write $(\boldsymbol{\eta},\boldsymbol{\zeta})_{\mathcal{T}_h} := \sum_{i=1}^d(\eta_i,\zeta_i)_{\mathcal{T}_h}$ and $(\mathbf{N},\mathbf{Z})_{\mathcal{T}_h} := \sum_{i,j=1}^d(\mathrm{N}_{ij},\mathrm{Z}_{ij})_{\mathcal{T}_h}$. Finally, we write $(\eta,\zeta)_{\partial\mathcal{T}_h} := \sum_{K\in\mathcal{T}_h}(\eta,\zeta)_{\partial K}$ and $(\boldsymbol{\eta},\boldsymbol{\zeta})_{\partial\mathcal{T}_h} := \sum_{i=1}^d\langle\eta_i,\zeta_i\rangle_{\partial\mathcal{T}_h}$, where $\langle\eta,\zeta\rangle_D$ denotes the integral of $\eta\,\zeta$ over the domain $D\subset\mathbb{R}^{d-1}$.

2.2 Definition of the HDG Methods

2.2.1 The Gradient-Based Formulation

We are now ready to define the HDG methods. First, we consider the velocity-pressure-gradient formulation of the Stokes system (1):

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

Here $\mathbf{L}$ is the gradient tensor and $\mathbf{I}$ is the second-order identity tensor. The HDG method for the velocity-pressure-gradient formulation seeks an approximation $(\mathbf{L}_h,\boldsymbol{u}_h,p_h,\widehat{\boldsymbol{u}}_h)\in\mathbf{G}_h\times\boldsymbol{V}_h\times P_h\times\boldsymbol{M}_h(\boldsymbol{g})$ such that

$$\begin{aligned}
(\mathbf{L}_h,\mathbf{G})_{\mathcal{T}_h}+(\boldsymbol{u}_h,\nabla\cdot\mathbf{G})_{\mathcal{T}_h}-\langle\widehat{\boldsymbol{u}}_h,\mathbf{G}\boldsymbol{n}\rangle_{\partial\mathcal{T}_h} &= 0,\\
(\nu\mathbf{L}_h-p_h\mathbf{I},\nabla\boldsymbol{v})_{\mathcal{T}_h}+\big\langle(-\nu\widehat{\mathbf{L}}_h+\widehat{p}_h\mathbf{I})\boldsymbol{n},\boldsymbol{v}\big\rangle_{\partial\mathcal{T}_h} &= (\boldsymbol{f},\boldsymbol{v})_{\mathcal{T}_h},\\
-(\boldsymbol{u}_h,\nabla q)_{\mathcal{T}_h}+\langle\widehat{\boldsymbol{u}}_h\cdot\boldsymbol{n},q\rangle_{\partial\mathcal{T}_h} &= 0,\\
\big\langle(-\nu\widehat{\mathbf{L}}_h+\widehat{p}_h\mathbf{I})\boldsymbol{n},\boldsymbol{\mu}\big\rangle_{\partial\mathcal{T}_h} &= 0,\\
(p_h,1)_{\mathcal{T}_h} &= 0,
\end{aligned}\tag{2a}$$

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

$$\big(-\nu\widehat{\mathbf{L}}_h+\widehat{p}_h\mathbf{I}\big)\boldsymbol{n}=(-\nu\mathbf{L}_h+p_h\mathbf{I})\,\boldsymbol{n}+\mathbf{S}(\boldsymbol{u}_h-\widehat{\boldsymbol{u}}_h).\tag{2b}$$

Here $\mathbf{S}$ is the *stabilization tensor*. A detailed analysis of how to choose it to obtain optimal convergence properties of the method can be found in [8]. One such choice is to set $\mathbf{S}:=\nu\tau\,\mathbf{I}$, where τ is a constant on $\partial\mathcal{T}_h$ and independent of the diameter of the elements. It can then be shown that the resulting HDG method is a "traditional" DG method whose numerical traces are

$$\widehat{\boldsymbol{u}}_h=\frac{1}{2}\boldsymbol{u}_h^+ +\frac{1}{2}\boldsymbol{u}_h^- -\frac{1}{2\nu\tau}[\![(\nu\mathbf{L}_h-p_h\mathbf{I})\boldsymbol{n}]\!],$$

$$\nu\widehat{\mathbf{L}}_h-\widehat{p}_h\mathbf{I}=\frac{1}{2}(\nu\mathbf{L}_h^+-p_h^+\mathbf{I})+\frac{1}{2}(\nu\mathbf{L}_h^--p_h^-\mathbf{I})-\frac{\nu\tau}{2}[\![\boldsymbol{u}_h\otimes\boldsymbol{n}]\!].$$

See in [16] the details of this computation as well as a brief comparison with the DG method proposed in [11].

2.2.2 The Vorticity-Based Formulation

Next, we consider the velocity-pressure-vorticity formulation of the Stokes system (1):

$$\begin{aligned} \mathbf{w} - \nabla \times \boldsymbol{u} &= \mathbf{0}, && \text{in } \Omega, \\ \nu \nabla \times \mathbf{w} + \nabla p &= \boldsymbol{f}, && \text{in } \Omega, \\ \nabla \cdot \boldsymbol{u} &= 0, && \text{in } \Omega, \\ \boldsymbol{u} &= \boldsymbol{g}, && \text{on } \partial\Omega, \\ \int_\Omega p &= 0. \end{aligned}$$

The HDG method for the velocity-pressure-vorticity formulation, [6], seeks an approximation $(\mathbf{w}_h, \boldsymbol{u}_h, p_h, \widehat{u}_h) \in \mathbf{W}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h(\boldsymbol{g})$ that satisfies

$$\begin{aligned} (\mathbf{w}_h, \mathbf{r})_{\mathcal{T}_h} - (\boldsymbol{u}_h, \nabla \times \mathbf{r})_{\mathcal{T}_h} - \langle \widehat{\boldsymbol{u}}_h, \mathbf{r} \times \boldsymbol{n} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (\nu \mathbf{w}_h, \nabla \times \boldsymbol{v})_{\mathcal{T}_h} - (p_h, \nabla \cdot \boldsymbol{v})_{\mathcal{T}_h} + \langle \nu \boldsymbol{n} \times \widehat{\mathbf{w}}_h + \widehat{p}_h \boldsymbol{n}, \boldsymbol{v} \rangle_{\partial\mathcal{T}_h} &= (\boldsymbol{f}, \boldsymbol{v})_{\mathcal{T}_h}, \\ -(\boldsymbol{u}_h, \nabla q)_{\mathcal{T}_h} + \langle \widehat{\boldsymbol{u}}_h \cdot \boldsymbol{n}, q \rangle_{\partial\mathcal{T}_h} &= 0, \\ \langle \nu \boldsymbol{n} \times \widehat{\mathbf{w}}_h + \widehat{p}_h \boldsymbol{n}, \boldsymbol{\mu} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (p_h, 1)_{\mathcal{T}_h} &= 0, \end{aligned} \tag{3a}$$

for all $(\mathbf{r}, \boldsymbol{v}, q, \boldsymbol{\mu}) \in \mathbf{W}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h(\mathbf{0})$, where

$$\nu \boldsymbol{n} \times \widehat{\mathbf{w}}_h + \widehat{p}_h \boldsymbol{n} = \nu \boldsymbol{n} \times \mathbf{w}_h + p_h \boldsymbol{n} + \mathbf{S}(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h). \tag{3b}$$

For simplicity of presentation we have adopted here the same notation used earlier to denote the approximate velocity, pressure and the stabilization tensor.

Note that when we take the stabilization tensor $\mathbf{S}$ as in the gradient-based formulation, the HDG method becomes a DG method whose numerical traces are

$$\begin{aligned} \widehat{\boldsymbol{u}}_h &= \frac{1}{2}\left(\boldsymbol{u}_h^+ + \boldsymbol{u}_h^-\right) + \frac{1}{2\nu\tau}[\![\nu \boldsymbol{n} \times \mathbf{w}_h + p_h \boldsymbol{n}]\!], \\ \widehat{\mathbf{w}}_h &= \frac{1}{2}\left(\mathbf{w}_h^+ + \mathbf{w}_h^-\right) + \frac{\nu\tau}{2}[\![\boldsymbol{u}_h \times \boldsymbol{n}]\!], \\ \widehat{p}_h &= \frac{1}{2}\left(p_h^+ + p_h^-\right) + \frac{\nu\tau}{2}[\![\boldsymbol{u}_h \cdot \boldsymbol{n}]\!]. \end{aligned}$$

2.2.3 The Stress-Based Formulation

Finally, we consider the velocity-pressure-stress formulation of the Stokes system (1):

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

The HDG method for the velocity-pressure-stress formulation seeks an approximation $(\boldsymbol{\sigma}_h, \boldsymbol{u}_h, p_h, \widehat{\boldsymbol{u}}_h) \in \boldsymbol{\Sigma}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h(\boldsymbol{g})$ that satisfies

$$\begin{aligned}
(\boldsymbol{\sigma}_h, \boldsymbol{\varsigma})_{\mathcal{T}_h} + 2(\boldsymbol{u}_h, \nabla\cdot\boldsymbol{\varsigma})_{\mathcal{T}_h} - 2\langle \widehat{\boldsymbol{u}}_h, \boldsymbol{\varsigma}\boldsymbol{n}\rangle_{\partial\mathcal{T}_h} &= 0,\\
(\nu\boldsymbol{\sigma}_h - p_h\mathbf{I}, \nabla\boldsymbol{v})_{\mathcal{T}_h} + \langle(-\nu\widehat{\boldsymbol{\sigma}}_h + \widehat{p}_h\mathbf{I})\boldsymbol{n}, \boldsymbol{v}\rangle_{\partial\mathcal{T}_h} &= (\boldsymbol{f}, \boldsymbol{v})_{\mathcal{T}_h},\\
-(\boldsymbol{u}_h, \nabla q)_{\mathcal{T}_h} + \langle\widehat{\boldsymbol{u}}_h\cdot\boldsymbol{n}, q\rangle_{\partial\mathcal{T}_h} &= 0,\\
\langle(-\nu\widehat{\boldsymbol{\sigma}}_h + \widehat{p}_h\mathbf{I})\boldsymbol{n}, \boldsymbol{\mu}\rangle_{\partial\mathcal{T}_h} &= 0,\\
(p_h, 1)_{\mathcal{T}_h} &= 0,
\end{aligned} \tag{4a}$$

for all $(\boldsymbol{\varsigma}, \boldsymbol{v}, q, \boldsymbol{\mu}) \in \boldsymbol{\Sigma}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h(\mathbf{0})$, where

$$(-\nu\widehat{\boldsymbol{\sigma}}_h + \widehat{p}_h\mathbf{I})\,\boldsymbol{n} = (-\nu\boldsymbol{\sigma}_h + p_h\mathbf{I})\,\boldsymbol{n} + \mathbf{S}(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h). \tag{4b}$$

This HDG method is strongly related to the HDG method proposed in [17] for linear elasticity. The methods are not quite the same however. Indeed, the elastic materials considered in [17] cannot taken to be exactly incompressible. Moreover, here we take $\boldsymbol{\Sigma_h}$ to be a space of symmetric matrices motivated by the fact that $\boldsymbol{\sigma}$ is a symmetric tensor.

Again, when the stabilization tensor is $\mathbf{S} := \nu\tau\,\mathbf{I}$, where τ is a constant on $\partial\mathcal{T}_h$, the resulting HDG method is a "traditional" DG method whose numerical traces are

$$\widehat{\boldsymbol{u}}_h = \frac{1}{2}\boldsymbol{u}_h^+ + \frac{1}{2}\boldsymbol{u}_h^- - \frac{1}{2\nu\tau}[\![(\nu\boldsymbol{\sigma}_h - p_h\mathbf{I})\boldsymbol{n}]\!],$$

$$\nu\widehat{\boldsymbol{\sigma}}_h - \widehat{p}_h\mathbf{I} = \frac{1}{2}(\nu\boldsymbol{\sigma}_h^+ - p_h^+\mathbf{I}) + \frac{1}{2}(\nu\boldsymbol{\sigma}_h^- - p_h^-\mathbf{I}) - \frac{\nu\tau}{2}[\![\boldsymbol{u}_h\otimes\boldsymbol{n}]\!].$$

2.3 Hybridization of the HDG Methods

The primary motivation for the hybridization is the reduction in the number of global degrees of freedom achieved by the elimination of volumetric unknowns in favor of facet unknowns. The hybridization is carried out in two steps. First, we introduce *local* Stokes problems at the element level and define associated *local solvers* by using a DG discretization. The local problems use the velocity trace and the pressure mean as boundary conditions and thus parametrize the velocity, gradient, and pressure in terms these variables. Second, we impose a conservativity condition by requiring that the numerical flues have to be *conservative* in the sense that the normal component of the numerical fluxes is single-valued across the interior faces. The conservativity condition results in a variational formulation in terms of the approximate trace of the velocity and the mean of the pressure only.

2.3.1 Hybridization of the Gradient-Based HDG Method

We first describe how to hybridize the HDG method for the velocity-pressure-gradient formulation. To begin, we introduce the local problem for all $K \in \mathcal{T}_h$,

$$\begin{aligned}
\mathbf{L} - \nabla\boldsymbol{u} &= \mathbf{0}, \quad \text{in } K,\\
\nabla\cdot(-\nu\mathbf{L} + p\mathbf{I}) &= \boldsymbol{f}, \quad \text{in } K,\\
\nabla\cdot\boldsymbol{u} &= 0, \quad \text{in } K,\\
\boldsymbol{u} &= \boldsymbol{\eta}, \quad \text{on } \partial K,\\
\overline{p} &= \rho, \quad \text{on } \partial K,
\end{aligned}$$

where $(\boldsymbol{f},\boldsymbol{\eta},\rho)\in \boldsymbol{L}^2(\Omega)\times \boldsymbol{M}_h(\boldsymbol{g})\times\overline{\Psi}_h$ is the data for the local problem. We then define the local solver $\mathcal{L}^{\mathrm{G}}$ that maps $(\boldsymbol{f},\boldsymbol{\eta},\rho)$ to the solution $(\mathbf{L}_h,\boldsymbol{u}_h,p_h)\in \mathbf{G}_h\times \boldsymbol{V}_h\times P_h$ of the problem

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

for all $(\mathbf{G},\boldsymbol{v},q)\in\mathbf{P}_k(K)\times\boldsymbol{P}_k(K)\times P_k(K)$ and each $K\in\mathcal{T}_h$. We then set

$$\begin{aligned}(\mathbf{L}_h^{\boldsymbol{f}},\boldsymbol{u}_h^{\boldsymbol{f}},p_h^{\boldsymbol{f}})&:=\mathcal{L}^{\mathrm{G}}(\boldsymbol{f},\mathbf{0},0),\\ (\mathbf{L}_h^{\boldsymbol{\eta}},\boldsymbol{u}_h^{\boldsymbol{\eta}},p_h^{\boldsymbol{\eta}})&:=\mathcal{L}^{\mathrm{G}}(\mathbf{0},\boldsymbol{\eta},0),\\ (\mathbf{L}_h^{\rho},\boldsymbol{u}_h^{\rho},p_h^{\rho})&:=\mathcal{L}^{\mathrm{G}}(\mathbf{0},\mathbf{0},\rho).\end{aligned}$$

We can now state the following result.

Proposition 2.1 ([16]) *Suppose that* $(\mathbf{L}_h,\boldsymbol{u}_h,p_h)$ *is the solution of* (2). *Let* $(\boldsymbol{\lambda},\varrho)\in(\boldsymbol{M}_h(\boldsymbol{g}),\overline{\Psi}_h)$ *satisfy*

$$\begin{aligned}\big\langle(\nu\mathbf{L}_h^{\boldsymbol{\lambda}}-p_h^{\boldsymbol{\lambda}}\mathbf{I})\boldsymbol{n}-\mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\lambda}}-\boldsymbol{\lambda}),\boldsymbol{\mu}\big\rangle_{\partial\mathcal{T}_h}-\langle\varrho,\boldsymbol{\mu}\cdot\boldsymbol{n}\rangle_{\partial\mathcal{T}_h}&=\Big\langle(-\nu\mathbf{L}_h^{\boldsymbol{f}}+p_h^{\boldsymbol{f}}\mathbf{I})\boldsymbol{n}+\mathbf{S}\boldsymbol{u}_h^{\boldsymbol{f}},\boldsymbol{\mu}\Big\rangle_{\partial\mathcal{T}_h},\\ -\big\langle\boldsymbol{\lambda}\cdot\boldsymbol{n},\overline{\psi}\big\rangle_{\partial\mathcal{T}_h}&=0,\end{aligned}\tag{5}$$

for all $(\boldsymbol{\mu},\overline{\psi})\in\boldsymbol{M}_h(\mathbf{0})\times\overline{\Psi}_h$, *and*

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

Then $\mathbf{L}_h=\mathbf{L}_h^{\boldsymbol{f}}+\mathbf{L}_h^{\boldsymbol{\lambda}}$, $\boldsymbol{u}_h=\boldsymbol{u}_h^{\boldsymbol{f}}+\boldsymbol{u}_h^{\boldsymbol{\lambda}}$, $p_h=p_h^{\boldsymbol{f}}+p_h^{\boldsymbol{\lambda}}+p_h^{\varrho}$, $\widehat{\boldsymbol{u}}_h=\boldsymbol{\lambda}$, *and* $\overline{p}_h=\varrho$.

Moreover, it is shown in [16] that the problem for $(\boldsymbol{\lambda},\varrho)$ as determined by (5) can be rewritten as a weak formulation giving rise to a matrix system typical of saddle point problems.

Proposition 2.2 ([16]) *The pair* $(\boldsymbol{\lambda},\varrho)\in(\boldsymbol{M}_h(\boldsymbol{g}),\overline{\Psi}_h)$ *as determined by* (5) *satisfies*

$$\begin{aligned}a_h(\boldsymbol{\lambda},\boldsymbol{\mu})+b_h(\varrho,\boldsymbol{\mu})&=f_h(\boldsymbol{\mu}),&&\forall\,\boldsymbol{\mu}\in\boldsymbol{M}_h(\mathbf{0}),\\ b_h(\overline{\psi},\boldsymbol{\lambda})&=0,&&\forall\,\overline{\psi}\in\overline{\Psi}_h,\end{aligned}\tag{6}$$

and

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

Here the forms are given by

$$\begin{aligned}a_h(\boldsymbol{\eta},\boldsymbol{\mu})&=(\nu\mathbf{L}_h^{\boldsymbol{\eta}},\mathbf{L}_h^{\boldsymbol{\mu}})_{\mathcal{T}_h}+\big\langle\mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta}}-\boldsymbol{\eta}),(\boldsymbol{u}_h^{\boldsymbol{\mu}}-\boldsymbol{\mu})\big\rangle_{\partial\mathcal{T}_h},\\ b_h(\overline{\psi},\boldsymbol{\mu})&=-\big\langle\overline{\psi},\boldsymbol{\mu}\cdot\boldsymbol{n}\big\rangle_{\partial\mathcal{T}_h},\\ f_h(\boldsymbol{\mu})&=\big(\boldsymbol{f},\boldsymbol{u}_h^{\boldsymbol{\mu}}\big)_{\mathcal{T}_h},\end{aligned}$$

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

Let us emphasize that this result is useful for the analysis of the method and its associated matrix equations, but not for implementation. For that purpose, Proposition 2.1 can be used.

2.3.2 Hybridization of the Vorticity-Based HDG Method

In a similar way, the hybridization of the HDG method for the velocity-pressure-vorticity formulation can be achieved. For simplicity of presentation we adapt the same notations used above. For the given data $(\boldsymbol{f},\boldsymbol{\eta},\rho)\in \boldsymbol{L}^2(\Omega)\times \boldsymbol{M}_h(\boldsymbol{g})\times \overline{\Psi}_h$ we introduce the local problem for all $K\in\mathcal{T}_h$,

$$\begin{aligned}
\mathbf{w}-\nabla\times\boldsymbol{u}&=\mathbf{0}, && \text{in } K,\\
\nabla\times\mathbf{w}+\nabla p&=\boldsymbol{f}, && \text{in } K,\\
\nabla\cdot\boldsymbol{u}&=0, && \text{in } K,\\
\boldsymbol{u}&=\boldsymbol{\eta}, && \text{on } \partial K,\\
\overline{p}&=\rho, && \text{on } \partial K.
\end{aligned}\tag{7}$$

Then we define the local solver as an operator $\mathcal{L}^{\mathrm{V}}$ that maps $(\boldsymbol{f},\boldsymbol{\eta},\rho)$ to the solution $(\mathbf{w}_h,\boldsymbol{u}_h,p_h)\in \mathbf{W}_h\times \boldsymbol{V}_h\times P_h$ of the problem,

$$\begin{aligned}
(\mathbf{w}_h,\mathbf{r})_K-(\boldsymbol{u}_h,\nabla\times\mathbf{r})_K&=\langle\boldsymbol{\eta},\mathbf{r}\times\boldsymbol{n}\rangle_{\partial K},\\
(\nu\nabla\times\mathbf{w}_h+\nabla p_h\mathbf{I}),\boldsymbol{v})_K+\langle\mathbf{S}\boldsymbol{u}_h,\boldsymbol{v}\rangle_{\partial K}&=(\boldsymbol{f},\boldsymbol{v})_K+\langle\mathbf{S}\boldsymbol{\eta},\boldsymbol{v}\rangle_{\partial K},\\
-(\boldsymbol{u}_h,\nabla q)_K&=-\langle\boldsymbol{\eta}\cdot\boldsymbol{n},q-\overline{q}\rangle_{\partial K},\\
\overline{p}_h&=\rho,
\end{aligned}$$

for all $(\mathbf{r},\boldsymbol{v},q)\in \mathbf{P}_k(K)\times \boldsymbol{P}_k(K)\times P_k(K)$ and each $K\in\mathcal{T}_h$. We then set

$$\begin{aligned}
(\mathbf{w}_h^f,\boldsymbol{u}_h^f,p_h^f)&:=\mathcal{L}^{\mathrm{V}}(\boldsymbol{f},\mathbf{0},0),\\
(\mathbf{w}_h^\eta,\boldsymbol{u}_h^\eta,p_h^\eta)&:=\mathcal{L}^{\mathrm{V}}(\mathbf{0},\boldsymbol{\eta},0),\\
(\mathbf{w}_h^\rho,\boldsymbol{u}_h^\rho,p_h^\rho)&:=\mathcal{L}^{\mathrm{V}}(\mathbf{0},\mathbf{0},\rho).
\end{aligned}$$

We can now state the following results.

Proposition 2.3 ([6]) *Suppose that* $(\mathbf{w}_h,\boldsymbol{u}_h,p_h)$ *is the solution of* (3). *Let* $(\boldsymbol{\lambda},\varrho)\in(\boldsymbol{M}_h(\boldsymbol{g}),\overline{\Psi}_h)$ *be such that*

$$\begin{aligned}
-\left\langle\nu\boldsymbol{n}\times\mathbf{w}_h^{\boldsymbol{\lambda}}+p_h^{\boldsymbol{\lambda}}\boldsymbol{n}-\mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\lambda}}-\boldsymbol{\lambda}),\boldsymbol{\mu}\right\rangle_{\partial\mathcal{T}_h}-\langle\varrho,\boldsymbol{\mu}\cdot\boldsymbol{n}\rangle_{\partial\mathcal{T}_h}&=\left\langle\nu\boldsymbol{n}\times\mathbf{w}_h^f+p_h^f\boldsymbol{n}+\mathbf{S}\boldsymbol{u}_h^f,\boldsymbol{\mu}\right\rangle_{\partial\mathcal{T}_h},\\
-\left\langle\boldsymbol{\lambda}\cdot\boldsymbol{n},\overline{\psi}\right\rangle_{\partial\mathcal{T}_h}&=0,
\end{aligned}\tag{8}$$

for all $(\boldsymbol{\mu},\overline{\psi})\in \boldsymbol{M}_h(\mathbf{0})\times\overline{\Psi}_h$, *and*

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

Then $\mathbf{w}_h=\mathbf{w}_h^f+\mathbf{w}_h^{\boldsymbol{\lambda}}$, $\boldsymbol{u}_h=\boldsymbol{u}_h^f+\boldsymbol{u}_h^{\boldsymbol{\lambda}}$, $p_h=p_h^f+p_h^{\boldsymbol{\lambda}}+p_h^\varrho$, $\widehat{\boldsymbol{u}}_h=\boldsymbol{\lambda}$, *and* $\overline{p}_h=\varrho$.

Proposition 2.4 ([6]) *The pair* $(\boldsymbol{\lambda},\varrho)\in(\boldsymbol{M}_h(\boldsymbol{g}),\overline{\Psi}_h)$ *as determined by* (8) *satisfies*

$$\begin{aligned}
c_h(\boldsymbol{\lambda},\boldsymbol{\mu})+d_h(\varrho,\boldsymbol{\mu})&=g_h(\boldsymbol{\mu}), && \forall\boldsymbol{\mu}\in\boldsymbol{M}_h(\mathbf{0}),\\
d_h(\overline{\psi},\boldsymbol{\lambda})&=0, && \forall\overline{\psi}\in\overline{\Psi}_h,
\end{aligned}\tag{9}$$

and

$$(p_h^f + p_h^\lambda + p_h^o, 1)_{\mathcal{T}_h} = 0.$$

Here the forms are given by

$$c_h(\boldsymbol{\eta}, \boldsymbol{\mu}) = (\nu \mathbf{w}_h^{\boldsymbol{\eta}}, \mathbf{w}_h^{\boldsymbol{\mu}})_{\mathcal{T}_h} + \left\langle \mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta}} - \boldsymbol{\eta}), (\boldsymbol{u}_h^{\boldsymbol{\mu}} - \boldsymbol{\mu}) \right\rangle_{\partial \mathcal{T}_h},$$

$$d_h(\overline{\psi}, \boldsymbol{\mu}) = -\left\langle \overline{\psi}, \boldsymbol{\mu} \cdot \boldsymbol{n} \right\rangle_{\partial \mathcal{T}_h},$$

$$g_h(\boldsymbol{\mu}) = \left(\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu}} \right)_{\mathcal{T}_h},$$

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

2.3.3 Hybridization of the Stress-Based HDG Method

As the hybridization of the stress-based HDG method can be carried out in the same fashion as the gradient-based HDG method, we do not provide the detailed procedure to save space. Instead we make two observations. First, the system of equations (8) is similar to the system (5) associated with the HDG method for the velocity-pressure-gradient formulation. In fact, the only difference between the two systems lies in the second equation due to the definition of the numerical fluxes. Second, the weak formulation (9) is also similar to the previous formulation (6). The difference between the two formulations lies in the first term of their first bilinear form.

Finally, we emphasize that the weak formulations (6) and (9) have significantly less globally coupled degrees of freedom than the original formulations (2) and (3), since the approximate trace of the velocity is defined on the element faces and the mean of the pressure is piecewise-constant. This huge advantage comes with the additional cost of solving the local problems on all elements of the triangulation. However, this additional cost is negligible as compared to the cost of solving the global system; see Sect. 3 for a detailed discussion.

2.4 Local Postprocessing

We use the element-by-element postprocessing proposed in [8] to obtain a new approximate velocity which is exactly divergence-free and $\boldsymbol{H}(\mathrm{div})$-conforming. This postprocessing is a modification of the BDM projection [2] which uses the numerical trace of the velocity and the optimally convergent approximate gradient.

We first consider the HDG solution of the velocity-pressure-gradient formulation. We define the new approximate velocity $\boldsymbol{u}_h^\star$ as the element of $\boldsymbol{\mathcal{P}}_{k+1}(K)$ such that for all $K \in \mathcal{T}_h$,

$$\langle (\boldsymbol{u}_h^\star - \widehat{\boldsymbol{u}}_h) \cdot \boldsymbol{n}, \mu \rangle_F = 0, \quad \forall \mu \in \mathcal{P}_k(F), \tag{10a}$$

$$\langle (\boldsymbol{n} \times \nabla)(\boldsymbol{u}_h^\star \cdot \boldsymbol{n}) - \boldsymbol{n} \times (\{\{\mathbf{L}_h^T\}\}\boldsymbol{n}), (\boldsymbol{n} \times \nabla)\mu \rangle_F = 0 \quad \forall \mu \in \mathcal{P}_{k+1}(F)^\perp, \tag{10b}$$

for all faces F of K, and such that

$$(\boldsymbol{u}_h^\star - \boldsymbol{u}_h, \nabla w)_K = 0, \quad \forall w \in \mathcal{P}_k(K), \tag{10c}$$

$$(\nabla \times \boldsymbol{u}_h^\star - \boldsymbol{\omega}_h, \nabla \times (\boldsymbol{v}\, \mathbf{B}_K))_K = 0, \quad \forall \boldsymbol{v} \in \boldsymbol{S}_k(K). \tag{10d}$$

Here $\{\{\mathbf{L}_h^T\}\}$ is the single-valued function on $\mathcal{E}_h$ equal to $((\mathbf{L}_h^T)^+ + (\mathbf{L}_h^T)^-)/2$ on the set $\mathcal{E}_h \setminus \partial\Omega$ and equal to $\mathbf{L}_h^T$ on $\partial\Omega$. We also have that

$$\mathcal{P}_k(F)^\perp := \{w \in \mathcal{P}_k(F) : \langle w, \zeta\rangle_F = 0,\ \forall \zeta \in \mathcal{P}_{k-1}(F)\}.$$

Moreover, $\boldsymbol{\omega}_h := (\mathrm{L}_{32h} - \mathrm{L}_{23h}, \mathrm{L}_{13h} - \mathrm{L}_{31h}, \mathrm{L}_{21h} - \mathrm{L}_{12h})$ is the approximation to the vorticity and $\mathbf{B}_K$ is the so-called *symmetric bubble matrix* introduced in [7], namely,

$$\mathbf{B}_K := \sum_{\ell=0}^{3} \lambda_{\ell-3}\lambda_{\ell-2}\lambda_{\ell-1}\nabla\lambda_\ell \otimes \nabla\lambda_\ell,$$

where λ_i are the barycentric coordinates associated with the tetrahedron K, the subindices being computed mod 4. The set $\boldsymbol{S}_k(K)$ is the space of vector-valued homogeneous polynomials $\boldsymbol{v}$ of degree k such that $\boldsymbol{v}\cdot\boldsymbol{x} = 0$ [15].

In the two dimensional case, the postprocessing is defined by the above equations if $\nabla \times \boldsymbol{u}$ is replaced by $\nabla \times \boldsymbol{u} := \partial_1 u_2 - \partial_2 u_1$, and if (10d) is replaced by

$$(\nabla \times \boldsymbol{u}_h^\star - \omega_h, w b_K)_K = 0, \quad \forall w \in \mathcal{P}_{k-1}(K),$$

where $b_K := \lambda_0\lambda_1\lambda_2$ and $\omega_h := \mathbf{L}_{21h} - \mathbf{L}_{12h}$.

We use the same postprocessing for the HDG solution of the velocity-pressure-vorticity formulation and of the velocity-pressure-stress formulation. To this end, we only need to define an approximate gradient $\mathbf{L}_h$. On each simplex $K \in \mathcal{T}_h$, we take $\mathbf{L}_h$ to be the element of $\mathbf{P}_k(K)$ defined by

$$(\mathbf{L}_h, \mathbf{G})_K = -(\boldsymbol{u}_h, \nabla\cdot\mathbf{G})_K + \langle \widehat{\boldsymbol{u}}_h, \mathbf{G}\boldsymbol{n}\rangle_{\partial K}, \quad \text{for all } \mathbf{G} \in \mathbf{P}_k(K). \tag{11}$$

Here the approximate velocity $\boldsymbol{u}_h$ and its numerical trace $\widehat{\boldsymbol{u}}_h$ are computed by using the vorticity-based HDG method or the stress-based HDG method.

For the three methods, the postprocessed velocity $\boldsymbol{u}_h^\star$ is well defined, belongs to $\boldsymbol{H}(div, \Omega)$, and is divergence-free; see [8].

3 Practical Implementation of the HDG Methods

In this section, we describe in detail how to implement the HDG methods via the augmented Lagrangian approach; see [12] and the references therein. The augmented Lagrangian approach renders the implementation of the HDG methods simple and efficient. The main advantage of this approach is that it eliminates the mean of pressure. However, this advantage comes with solving iteratively for the degrees of freedom of the velocity.

We shall focus our attention on the HDG method for the velocity-pressure-vorticity formulation. The implementation procedure for the other HDG methods can be carried out in a similar way; see [16] for additional details.

3.1 Augmented Lagrangian Approach

The augmented Lagrangian [12] is widely used for solving the steady incompressible Navier-Stokes equations. In this method the steady solution is computed as the asymptotic

limit of a time-dependent solution of an evolution problem as time goes to infinity. We first introduce an artificial time derivative to the continuity equation as follows

$$\begin{aligned} \frac{\partial p(t)}{\partial t} + \nabla \cdot \boldsymbol{u}(t) = 0, \quad & \text{in } \Omega \times (0,\infty), \\ p(t=0) = 0, \quad & \text{in } \Omega, \end{aligned}$$

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

$$\begin{aligned} \mathbf{w} - \nabla \times \boldsymbol{u} = 0, \quad & \text{in } \Omega \\ \nu \nabla \times \mathbf{w} + \nabla p = \boldsymbol{f}, \quad & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{g}, \quad & \text{on } \partial\Omega. \end{aligned}$$

Note that, since $\in_{\partial\Omega} \boldsymbol{g} \cdot \boldsymbol{n} = 0$, the mean of $p(t)$ over Ω is equal to 0 at all times $t \geq 0$.

Next, given a constant time step Δt and a pressure p_h^{n-1} for $n \geq 1$ with $p_h^0 = 0$, we define the iterate $(\mathbf{w}_h^n, \boldsymbol{u}_h^n, p_h^n, \widehat{\boldsymbol{u}}_h^n) \in \mathbf{W}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h(\boldsymbol{g})$ as the solution of

$$\begin{aligned} (\mathbf{w}_h^n, \mathbf{r})_{\mathcal{T}_h} - (\boldsymbol{u}_h^n, \nabla \times \mathbf{r})_{\mathcal{T}_h} - \langle \widehat{\boldsymbol{u}}_h^n, \mathbf{r} \times \boldsymbol{n} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (\nu \mathbf{w}_h^n, \nabla \times \boldsymbol{v})_{\mathcal{T}_h} - (p_h^n, \nabla \cdot \boldsymbol{v})_{\mathcal{T}_h} + \langle \nu \boldsymbol{n} \times \widehat{\mathbf{w}}_h^n + \widehat{p}_h^n \boldsymbol{n}, \boldsymbol{v} \rangle_{\partial\mathcal{T}_h} &= (\boldsymbol{f}, \boldsymbol{v})_{\mathcal{T}_h}, \\ \frac{1}{\Delta t}(p_h^n, q)_{\mathcal{T}_h} - (\boldsymbol{u}_h^n, \nabla q)_{\mathcal{T}_h} + \langle \widehat{\boldsymbol{u}}_h^n \cdot \boldsymbol{n}, q \rangle_{\partial\mathcal{T}_h} &= \frac{1}{\Delta t}(p_h^{n-1}, q)_{\mathcal{T}_h}, \\ \langle \nu \boldsymbol{n} \times \widehat{\mathbf{w}}_h^n + \widehat{p}_h^n \boldsymbol{n}, \boldsymbol{\mu} \rangle_{\partial\mathcal{T}_h} &= 0, \end{aligned} \tag{12a}$$

for all $(\mathbf{r}, \boldsymbol{v}, q, \boldsymbol{\mu}) \in \mathbf{W}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h(\mathbf{0})$, where

$$\nu \boldsymbol{n} \times \widehat{\mathbf{w}}_h^n + \widehat{p}_h^n \boldsymbol{n} = \nu \boldsymbol{n} \times \mathbf{w}_h^n + p_h^n \boldsymbol{n} + \mathbf{S}(\boldsymbol{u}_h^n - \widehat{\boldsymbol{u}}_h^n), \quad \text{on } \partial\mathcal{T}_h. \tag{12b}$$

We stop the iterations when the relative error of the pressure is less than a prescribed tolerance ε_{tol}, that is, when

$$\frac{\|p_h^n - p_h^{n-1}\|_{\mathcal{T}_h}}{\|p_h^n\|_{\mathcal{T}_h}} < \varepsilon_{\text{tol}}. \tag{13}$$

It is easy to show, see [16], that the sequence $(\mathbf{w}_h^n, \boldsymbol{u}_h^n, p_h^n, \widehat{\boldsymbol{u}}_h^n)$ converges exponentially in time to the original HDG approximation $(\mathbf{w}_h, \boldsymbol{u}_h, p_h, \widehat{\boldsymbol{u}}_h)$ introduced in Sect. 2. It remains to describe how to solve the above system (12).

3.2 Implementation Considerations

Here, we show how the only globally coupled variable needed to solve for $(\mathbf{w}_h^n, \boldsymbol{u}_h^n, p_h^n, \widehat{\boldsymbol{u}}_h^n)$ is $\widehat{\boldsymbol{u}}_h^n$. We proceed as in Sect. 2.3. For any given positive number Δt, we define the local solver $\mathcal{L}^{\Delta t}$ as the operator that maps $(\boldsymbol{f}, \boldsymbol{\eta}, \theta) \in \boldsymbol{L}^2(\Omega) \times \boldsymbol{M}_h(\boldsymbol{g}) \times P_h$ to $(\mathbf{w}_h^{\ell,\Delta t}, \boldsymbol{u}_h^{\ell,\Delta t}, p_h^{\ell,\Delta t}) \in \mathbf{W}_h \times \boldsymbol{V}_h \times P_h$, solution of

$$\begin{aligned} &(\mathbf{w}_h^{\ell,\Delta t}, \mathbf{r})_K - (\boldsymbol{u}_h^{\ell,\Delta t}, \nabla \times \mathbf{r})_K = \langle \boldsymbol{\eta}, \mathbf{r} \times \boldsymbol{n} \rangle_{\partial K}, \\ &(\nu \nabla \times \mathbf{w}_h^{\ell,\Delta t} + \nabla p_h^{\ell,\Delta t} \mathbf{I}), \boldsymbol{v})_K \\ &\quad + \left\langle \mathbf{S}\boldsymbol{u}_h^{\ell,\Delta t}, \boldsymbol{v} \right\rangle_{\partial K} = (\boldsymbol{f}, \boldsymbol{v})_K + \langle \mathbf{S}\boldsymbol{\eta}, \boldsymbol{v} \rangle_{\partial K}, \\ &\frac{1}{\Delta t}(p_h^{\ell,\Delta t}, q)_K - (\boldsymbol{u}_h^{\ell,\Delta t}, \nabla q)_K = \frac{1}{\Delta t}(\theta, q)_{\mathcal{T}_h} - \langle \boldsymbol{\eta} \cdot \boldsymbol{n}, q \rangle_{\partial K}, \end{aligned} \tag{14}$$

for all $(\mathbf{r}, \boldsymbol{v}, q) \in \mathbf{P}_k(K) \times \boldsymbol{\mathcal{P}}_k(K) \times \mathcal{P}_k(K)$ and each $K \in \mathcal{T}_h$. We then set

$$(\mathbf{w}_h^{\boldsymbol{f},\Delta t}, \boldsymbol{u}_h^{\boldsymbol{f},\Delta t}, p_h^{\boldsymbol{f},\Delta t}) := \mathcal{L}^{\Delta t}(\boldsymbol{f}, \mathbf{0}, 0),$$

$$(\mathbf{w}_h^{\boldsymbol{\eta},\Delta t}, \boldsymbol{u}_h^{\boldsymbol{\eta},\Delta t}, p_h^{\boldsymbol{\eta},\Delta t}) := \mathcal{L}^{\Delta t}(\mathbf{0}, \boldsymbol{\eta}, 0),$$

$$(\mathbf{w}_h^{\theta,\Delta t}, \boldsymbol{u}_h^{\theta,\Delta t}, p_h^{\theta,\Delta t}) := \mathcal{L}^{\Delta t}(\mathbf{0}, \mathbf{0}, \theta).$$

We then have, see Sect. 2.3, the following result.

Proposition 3.1 *Let* $(\mathbf{w}_h^n, \boldsymbol{u}_h^n, p_h^n, \widehat{\boldsymbol{u}}_h^n)$ *be the solution of* (12). *Then we have that*

$$\begin{aligned}
\mathbf{w}_h^n &= \mathbf{w}_h^{\boldsymbol{f},\Delta t} + \mathbf{w}_h^{p_h^{n-1},\Delta t} + \mathbf{w}_h^{\boldsymbol{\lambda}^n,\Delta t},\\
\boldsymbol{u}_h^n &= \boldsymbol{u}_h^{\boldsymbol{f},\Delta t} + \boldsymbol{u}_h^{p_h^{n-1},\Delta t} + \boldsymbol{u}_h^{\boldsymbol{\lambda}^n,\Delta t},\\
p_h^n &= p_h^{\boldsymbol{f},\Delta t} + p_h^{p_h^{n-1},\Delta t} + p_h^{\boldsymbol{\lambda}^n,\Delta t},\\
\widehat{\boldsymbol{u}}_h^n &= \boldsymbol{\lambda}^n,
\end{aligned} \tag{15}$$

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

$$c_h^{\Delta t}(\boldsymbol{\lambda}^n, \boldsymbol{\mu}) = g_h^{\Delta t}(\boldsymbol{\mu}; p_h^{n-1}), \quad \forall \boldsymbol{\mu} \in \boldsymbol{M}_h(\mathbf{0}), \tag{16a}$$

where

$$\begin{aligned}
c_h^{\Delta t}(\boldsymbol{\eta}, \boldsymbol{\mu}) = &\left(\nu \mathbf{w}_h^{\boldsymbol{\eta},\Delta t}, \mathbf{w}_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathcal{T}_h} + \left\langle \mathbf{S}(\boldsymbol{u}_h^{\boldsymbol{\eta},\Delta t} - \boldsymbol{\eta}), (\boldsymbol{u}_h^{\boldsymbol{\mu},\Delta t} - \boldsymbol{\mu}) \right\rangle_{\partial \mathcal{T}_h}\\
&+ \frac{1}{\Delta t}\left(p_h^{\boldsymbol{\eta},\Delta t}, p_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathcal{T}_h},
\end{aligned} \tag{16b}$$

$$g_h^{\Delta t}(\boldsymbol{\mu}; p_h^{n-1}) = \left(\boldsymbol{f}, \boldsymbol{u}_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathcal{T}_h} - \frac{1}{\Delta t}\left(p_h^{n-1}, p_h^{\boldsymbol{\mu},\Delta t}\right)_{\mathcal{T}_h},$$

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

The weak formulation (16) gives rise to a system of equations of the form

$$\mathbb{A}\,\Lambda^n = R^n,$$

where Λ^n represents the degrees of freedom for $\boldsymbol{\lambda}^n$. To form $\mathbb{A}$ and R^n, we compute the elemental matrices and vectors for all elements $K \in \mathcal{T}_h$ as follows

$$\begin{aligned}
\mathbb{A}_{ij}^K &= (\nu \mathbf{w}_K^{\boldsymbol{\mu}_i,\Delta t}, \mathbf{w}_K^{\boldsymbol{\mu}_j,\Delta t})_K + \frac{1}{\Delta t}(p_K^{\boldsymbol{\mu}_i,\Delta t}, p_K^{\boldsymbol{\mu}_j,\Delta t})_K\\
&\quad + \left\langle \mathbf{S}(\boldsymbol{u}_K^{\boldsymbol{\mu}_i,\Delta t} - \boldsymbol{\mu}_i), (\boldsymbol{u}_K^{\boldsymbol{\mu}_j,\Delta t} - \boldsymbol{\mu}_j) \right\rangle_{\partial K}, \qquad 1 \le i, j \le N,\\
R_i^{K,n} &= (\boldsymbol{f}, \boldsymbol{u}_K^{\boldsymbol{\mu}_i,\Delta t})_K - \frac{1}{\Delta t}(p_h^{n-1}, p_K^{\boldsymbol{\mu}_i,\Delta t})_K, \qquad 1 \le i \le N,
\end{aligned} \tag{17}$$

where $(\mathbf{w}_K^{\boldsymbol{\mu}_i,\Delta t}, \boldsymbol{u}_K^{\boldsymbol{\mu}_i,\Delta t}, p_K^{\boldsymbol{\mu}_i,\Delta t}) := \mathcal{L}^{\Delta t}(\mathbf{0}, \boldsymbol{\mu}_i, 0)$ and $N = (1+d)\,n$, where $n := (k+d-1)!/(k!(d-1)!)$. Here, $\{\boldsymbol{\mu}_i\}_{i=1+n\,m}^{n\,m+n}$ is a basis of $\boldsymbol{\mathcal{P}}_k(F_m)$ where F_m is the $m-th$ face of K, for $m = 1, \ldots, d+1$. The solution procedure is summarized Table 1.

Table 1 Implementation of the HDG method for the velocity-pressure-vorticity formulation

Implementation Steps
Step 1. Given ε_{tol}, pick Δt and set $n := 1$.
Step 2. For every element K of $\mathcal{T}_h$, solve the local solver (14) $(\mathbf{w}_K^{\boldsymbol{f},\Delta t}, \boldsymbol{u}_K^{\boldsymbol{f},\Delta t}, p_K^{\boldsymbol{f},\Delta t}) := \mathcal{L}^{\Delta t}(\boldsymbol{f}, \mathbf{0}, 0)$, $(\mathbf{w}_K^{\boldsymbol{\mu}_i}, \boldsymbol{u}_K^{\boldsymbol{\mu}_i}, p_K^{\boldsymbol{\mu}_i}) := \mathcal{L}^{\Delta t}(\mathbf{0}, \boldsymbol{\eta}, 0)$ for $\boldsymbol{\eta} := \boldsymbol{\mu}_i$, $1 \le i \le (d+1)\dim \boldsymbol{\mathcal{P}}_k(F)$, $(\mathbf{w}_K^{\varphi_j}, \boldsymbol{u}_K^{\varphi_j}, p_K^{\varphi_j}) := \mathcal{L}^{\Delta t}(\mathbf{0}, \mathbf{0}, \theta)$ for $\theta := \varphi_j \in \mathcal{P}_k(K)$, $1 \le j \le \dim \mathcal{P}_k(K)$.
Step 3. Calculate $\mathbb{A}^K$ from (17) for every $K \in \mathcal{T}_h$ to assemble the matrix $\mathbb{A}$.
Step 4. Calculate $R^{K,n}$ from (17) for every $K \in \mathcal{T}_h$ to assemble the vector R^n.
Step 5. Solve $\mathbb{A}\boldsymbol{\Lambda}^n = R^n$, where $\boldsymbol{\Lambda}^n$ is the vector of degrees of freedom of $\boldsymbol{\lambda}^n$.
Step 6. Compute $(\mathbf{w}_h^n, \boldsymbol{u}_h^n, p_h^n)$ according to (15) .
Step 7. If (13) does not hold, set $n := n+1$ and go to **Step 4**.
Step 8. If it does, then stop.

3.3 Computational Complexity and Storage Requirement

Let us discuss the computational complexity and memory storage required by the HDG method. The cost of the local solver per element is $O(N_u^3)$, where $N_u = \dim \boldsymbol{\mathcal{P}}_k(K)$. Hence, the total cost of the local solver is $O(N_K N_u^3)$, where N_K is the number of elements of the triangulation.

Next, let us compute the number of degrees of freedom and describe the sparsity structure of the linear system (3.2), restricting our attention to the case of a conforming triangulation $\mathcal{T}_h$ (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 $\boldsymbol{\mathcal{P}}_k(F)$ for each face F. The number of block rows and block columns is equal to N_F, where N_F is the number of interior faces of the triangulation. Furthermore, on each block row, there are at most $(2d+1)$ blocks that are not equal to zero. Hence, the size of $\mathbb{A}$ is $N_{\text{dof}} \times N_{\text{dof}}$ and the number of nonzero entries of $\mathbb{K}$ is $N_F(2d+1)\dim \boldsymbol{\mathcal{P}}_k(F)$, where $N_{\text{dof}} = N_F(d+1)\dim \boldsymbol{\mathcal{P}}_k(F)$. In general, the solution of the linear system (3.2) will cost $O(N_{\text{dof}}^{\gamma})$ with $\gamma \sim 2$ typically. Therefore, the computational complexity of the HDG method will be dominated by the cost of solving the linear system (3.2) since the operation count of the local solver scales linearly with N_{dof}.

We end this section by noting that the gradient-based and stress-based HDG methods produce a global matrix which has the same structure and size as the matrix of the vorticity-based HDG method. Therefore, the three HDG methods have the same computational complexity and storage requirement.

4 Numerical Results

4.1 Example with Smooth Solution

We consider the Stokes problem whose exact solution coincides with the analytical solution of the incompressible Navier-Stokes equations obtained by Kovasznay in [13], namely,

$$u_1 = 1 - \exp(\lambda x_1)\cos(2\pi x_2),$$
$$u_2 = \frac{\lambda}{2\pi}\exp(\lambda x_1)\sin(2\pi x_2),$$
$$p = \frac{1}{2}\exp(2\lambda x_1),$$

where $\lambda = \frac{Re}{2} - \sqrt{\frac{Re^2}{4} + 4\pi^2}$ and $Re = \frac{1}{\nu}$ is the Reynolds number. We take Dirichlet boundary conditions for the velocity as the restriction of the exact solution to the domain boundary. Here the computational domain is $\Omega = (-0.5, 1.5) \times (0, 2)$ 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. In all cases, the stabilization tensor $\mathbf{S}$ is chosen as

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

where τ is some positive constant defined on $\mathcal{E}_h$. Here we choose $\nu\tau = 1$. Below we compare the convergence and accuracy properties of the HDG methods. We use superscripts G, S, and V to indicate the approximate solution computed by the gradient-based, stress-based, and vorticity-based HDG methods, respectively.

We present a history of convergence of the HDG methods in Tables 2 to 4. We observe from Tables 2 and 3 that all the approximate variables converge with the optimal order $k+1$ for all the methods. However, the gradient-based HDG method is superior since it yields smaller errors than the other two methods. Moreover, as shown in Table 4, the postprocessed velocity of the gradient-based HDG method converges with order $k+2$ for $k \geq 1$, while that of the vorticity HDG method converges with order $k+1$ only. It is interesting to note from Table 4 that the postprocessed velocity of the stress-based HDG method converges with order $k+1$ for $k = 0, 1, 3$, but tends to converge with order $k+2$ for $k = 2, 4$.

To visualize the effect of the local postprocessing, we show in Fig. 1 the original and postprocessed horizontal velocities of the three HDG methods for $k = 2$ on the same mesh. Figures 2–4 display the original, postprocessed, and exact horizontal velocities along the line $x = -0.4$. We observe that our local postprocessing is not effective for the vorticity-based HDG method since it does not really improve the approximation of the velocity in this case. However, our local postprocessing is very effective for the stress-based and gradient-based HDG methods since the postprocessed velocity is clearly superior to the original velocity. We should also note that the postprocessed velocity is divergence-free and $\boldsymbol{H}(\mathrm{div})$-conforming, whereas the original velocity is not.

Finally, we look at the effect of the artificial time step Δt on the condition number of the stiffness matrix and the number of iterations required to reach the error tolerance $\varepsilon_{\mathrm{tol}} = 10^{-8}$. We define the condition number ratio R as

$$R := \frac{C}{(1 + \Delta t/\nu)(k+1)h^{-2}}, \tag{18}$$

where C denotes the condition number of the stiffness matrix, which is the ratio of the largest singular value of the matrix to the smallest singular value. We report in Table 5 the

Table 2 History of convergence of the approximate velocity and pressure for the Stokes problem with smooth solution

Degree	Mesh	$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{G}}\|_{\mathcal{T}_h}$		$\|p-p_h^{\mathrm{G}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{S}}\|_{\mathcal{T}_h}$		$\|p-p_h^{\mathrm{S}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{V}}\|_{\mathcal{T}_h}$		$\|p-p_h^{\mathrm{V}}\|_{\mathcal{T}_h}$	
k	h^{-1}	error	order	error	order	error	order	error	order	error	order	error	order
0	2	2.06e-0	–	1.35e-0	–	1.94e-0	–	1.46e-0	–	2.43e-0	–	1.14e-0	–
	4	1.56e-0	0.40	5.75e-1	1.23	1.43e-0	0.44	8.42e-1	0.79	2.03e-0	0.26	3.79e-1	1.59
	8	7.19e-1	1.12	4.82e-1	0.25	6.57e-1	1.12	7.33e-1	0.20	1.10e-0	0.88	2.17e-1	0.81
	16	3.34e-1	1.10	2.66e-1	0.86	3.17e-1	1.05	3.90e-1	0.91	5.71e-1	0.95	1.21e-1	0.84
	32	1.58e-1	1.08	1.44e-1	0.89	1.54e-1	1.04	1.97e-1	0.98	2.89e-1	0.98	7.17e-2	0.76
1	2	9.55e-1	–	9.36e-1	–	9.61e-1	–	1.08e-0	–	1.08e-0	–	1.04e-0	–
	4	2.51e-1	1.93	2.87e-1	1.71	2.46e-1	1.97	3.28e-1	1.72	4.81e-1	1.16	4.14e-1	1.33
	8	6.61e-2	1.93	7.85e-2	1.87	6.54e-2	1.91	9.14e-2	1.85	1.36e-1	1.82	1.28e-1	1.70
	16	1.62e-2	2.03	2.01e-2	1.97	1.62e-2	2.01	2.42e-2	1.92	3.55e-2	1.94	3.67e-2	1.80
	32	3.98e-3	2.02	5.04e-3	1.99	4.03e-3	2.01	6.24e-3	1.95	9.07e-3	1.97	1.02e-2	1.85
2	2	2.31e-1	–	2.27e-1	–	2.32e-1	–	2.54e-1	–	4.46e-1	–	4.22e-1	–
	4	3.47e-2	2.74	3.77e-2	2.59	3.46e-2	2.74	4.34e-2	2.55	4.69e-2	3.25	4.33e-2	3.28
	8	4.21e-3	3.04	5.10e-3	2.89	4.21e-3	3.04	6.24e-3	2.80	5.56e-3	3.08	6.02e-3	2.85
	16	5.26e-4	3.00	6.50e-4	2.97	5.26e-4	3.00	8.35e-4	2.90	6.92e-4	3.01	7.72e-4	2.96
	32	6.54e-5	3.01	8.14e-5	3.00	6.54e-5	3.01	1.08e-4	2.95	8.72e-5	2.99	9.67e-5	3.00

Table 3 History of convergence of the approximate stress and vorticity for the Stokes problem with smooth solution. Note for the gradient-based HDG method that $\boldsymbol{\sigma}_h^{\mathrm{G}} = \mathbf{L}_h^{\mathrm{G}} + (\mathbf{L}_h^{\mathrm{G}})^T$ and $\mathbf{w}_h^{\mathrm{G}} = \mathbf{L}_h^{\mathrm{G}} - (\mathbf{L}_h^{\mathrm{G}})^T$

Degree	Mesh	$\|\mathbf{w}-\mathbf{w}_h^{\mathrm{G}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{\sigma}-\boldsymbol{\sigma}_h^{\mathrm{G}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{\sigma}-\boldsymbol{\sigma}_h^{\mathrm{S}}\|_{\mathcal{T}_h}$		$\|\mathbf{w}-\mathbf{w}_h^{\mathrm{V}}\|_{\mathcal{T}_h}$	
k	h^{-1}	error	order	error	order	error	order	error	order
0	2	1.15e-9	–	1.71e-9	–	1.68e-9	–	1.11e-9	–
	4	7.12e-0	0.70	1.16e-9	0.56	1.26e-9	0.41	7.64e-0	0.54
	8	3.28e-0	1.12	7.92e-0	0.55	7.77e-0	0.70	3.60e-0	1.08
	16	1.97e-0	0.74	4.82e-0	0.72	4.47e-0	0.80	1.89e-0	0.93
	32	1.23e-0	0.68	2.79e-0	0.79	2.43e-0	0.88	9.76e-1	0.95
1	2	4.95e-0	–	8.02e-0	–	8.81e-0	–	5.21e-0	–
	4	1.51e-0	1.72	2.68e-0	1.58	2.93e-0	1.59	2.7e-0	0.95
	8	4.90e-1	1.62	8.39e-1	1.67	8.88e-1	1.72	8.74e-1	1.63
	16	1.35e-1	1.86	2.32e-1	1.85	2.44e-1	1.87	2.44e-1	1.84
	32	3.53e-2	1.93	6.11e-2	1.92	6.42e-2	1.92	6.56e-2	1.90
2	2	1.72e-0	–	2.24e-0	–	2.93e-0	–	3.3e-0	–
	4	2.62e-1	2.72	3.73e-1	2.58	4.56e-1	2.69	4.33e-1	2.93
	8	3.59e-2	2.87	5.33e-2	2.81	6.48e-2	2.82	4.79e-2	3.18
	16	4.76e-3	2.91	7.20e-3	2.89	8.73e-3	2.89	6.13e-3	2.96
	32	6.14e-4	2.96	9.35e-4	2.94	1.14e-3	2.94	7.71e-4	2.99

Table 4 History of convergence of the approximate gradient and the postprocessed velocity for the Stokes problem with smooth solution. Note that $\mathbf{L}_h^{\mathrm{S}}$ and $\mathbf{L}_h^{\mathrm{V}}$ are computed from (11)

Degree	Mesh	$\|\mathbf{L}-\mathbf{L}_h^{\mathrm{G}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{G},\star}\|_{\mathcal{T}_h}$		$\|\mathbf{L}-\mathbf{L}_h^{\mathrm{S}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{S},\star}\|_{\mathcal{T}_h}$		$\|\mathbf{L}-\mathbf{L}_h^{\mathrm{V}}\|_{\mathcal{T}_h}$		$\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{V},\star}\|_{\mathcal{T}_h}$	
k	h^{-1}	error	order	error	order	error	order	error	order	error	order	error	order
0	2	1.47e-9	–	2.25e-0	–	1.47e-9	–	2.26e-0	–	1.66e-9	–	2.49e-0	–
	4	1.05e-9	0.48	1.23e-0	0.87	9.31e-0	0.66	1.14e-0	0.98	1.72e-9	−0.05	1.73e-0	0.52
	8	6.75e-0	0.64	4.61e-1	1.42	6.32e-0	0.56	4.21e-1	1.44	1.81e-9	−0.07	9.14e-1	0.92
	16	4.14e-0	0.71	2.00e-1	1.20	4.78e-0	0.40	1.98e-1	1.09	1.85e-9	−0.03	4.74e-1	0.95
	32	2.45e-0	0.76	9.38e-2	1.09	4.10e-0	0.22	9.84e-2	1.01	1.86e-9	−0.01	2.40e-1	0.98
	64	1.45e-0	0.75	4.59e-2	1.03	3.81e-0	0.11	4.95e-2	0.99	1.86e-9	0.00	1.21e-1	0.99
1	2	6.97e-0	–	5.25e-1	–	7.10e-0	–	5.46e-1	–	1.16e-9	–	8.07e-1	–
	4	2.34e-0	1.57	1.01e-1	2.38	2.58e-0	1.46	1.10e-1	2.31	6.81e-0	0.77	4.43e-1	0.87
	8	7.48e-1	1.65	1.68e-2	2.59	8.95e-1	1.53	1.98e-2	2.48	3.33e-0	1.03	1.19e-1	1.89
	16	2.08e-1	1.85	2.39e-3	2.81	3.60e-1	1.31	3.91e-3	2.34	1.72e-0	0.95	3.08e-2	1.95
	32	5.51e-2	1.92	3.21e-4	2.89	1.71e-1	1.07	9.04e-4	2.11	8.98e-1	0.94	7.84e-3	1.98
	64	1.42e-2	1.96	4.18e-5	2.94	8.66e-2	0.98	2.24e-4	2.01	4.62e-1	0.96	1.98e-3	1.99
2	2	2.12e-0	–	1.03e-1	–	2.41e-0	–	1.15e-1	–	4.94e-0	–	4.38e-1	–
	4	3.50e-1	2.60	1.19e-2	3.11	3.87e-1	2.64	1.23e-2	3.22	1.23e-0	2.01	4.24e-2	3.37
	8	4.89e-2	2.84	8.20e-4	3.86	6.26e-2	2.63	9.44e-4	3.71	3.00e-1	2.03	4.91e-3	3.11
	16	6.56e-3	2.90	5.56e-5	3.88	8.95e-3	2.80	6.69e-5	3.82	7.23e-2	2.05	5.89e-4	3.06
	32	8.49e-4	2.95	3.62e-6	3.94	1.22e-3	2.87	4.51e-6	3.89	1.80e-2	2.00	7.37e-5	3.00
	64	1.08e-4	2.97	2.31e-7	3.97	1.67e-4	2.87	3.03e-7	3.90	4.56e-3	1.98	9.35e-6	2.98
3	1	4.05e-0	–	3.16e-1	–	4.54e-0	–	3.43e-1	–	7.48e-0	–	9.78e-1	–
	2	4.32e-1	3.23	1.78e-2	4.15	4.72e-1	3.27	1.89e-2	4.18	9.23e-1	3.02	4.46e-2	4.45
	4	3.37e-2	3.68	8.39e-4	4.41	4.17e-2	3.50	9.74e-4	4.28	1.41e-1	2.71	3.92e-3	3.51
	8	2.48e-3	3.77	3.24e-5	4.69	3.54e-3	3.56	4.35e-5	4.48	1.99e-2	2.83	2.78e-4	3.82
	16	1.64e-4	3.91	1.09e-6	4.89	3.32e-4	3.42	2.19e-6	4.31	2.69e-3	2.89	1.82e-5	3.93
	32	1.06e-5	3.96	3.53e-8	4.95	3.74e-5	3.15	1.29e-7	4.09	3.51e-4	2.94	1.16e-6	3.97
4	1	1.38e-0	–	1.09e-1	–	1.51e-0	–	1.14e-1	–	2.29e-0	–	2.31e-1	–
	2	7.23e-2	4.26	2.53e-3	5.43	8.45e-2	4.16	2.89e-3	5.31	2.02e-1	3.50	1.15e-2	4.33
	4	2.93e-3	4.63	6.45e-5	5.30	3.71e-3	4.51	7.25e-5	5.32	1.36e-2	3.89	3.28e-4	5.13
	8	9.98e-5	4.87	1.10e-6	5.87	1.38e-4	4.75	1.33e-6	5.77	8.81e-4	3.95	1.00e-5	5.03
	16	3.25e-6	4.94	1.81e-8	5.93	4.79e-6	4.85	2.30e-8	5.85	5.57e-5	3.98	3.17e-7	4.99
	32	1.04e-7	4.97	2.9e-10	5.97	1.59e-7	4.91	3.82e-10	5.91	3.54e-6	3.98	1.01e-8	4.97

condition number ratio and in Table 6 the number of iterations for convergence as a function of h, k, and Δt for the gradient-based and vorticity-based HDG methods. We observe that the matrix of the vorticity-based HDG method has a slightly smaller condition number than that of the gradient-based HDG method. We also see that the condition number ratio is close to 2 for most cases. As a consequence, we have that the condition numbers are close to

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

Moreover, both the HDG methods the number of iterations for convergence is quite similar, relatively small, and independent of the mesh size h and polynomial degree k. We note that the results obtained for the stress-based HDG method are similar to those obtained for the gradient-based HDG method.

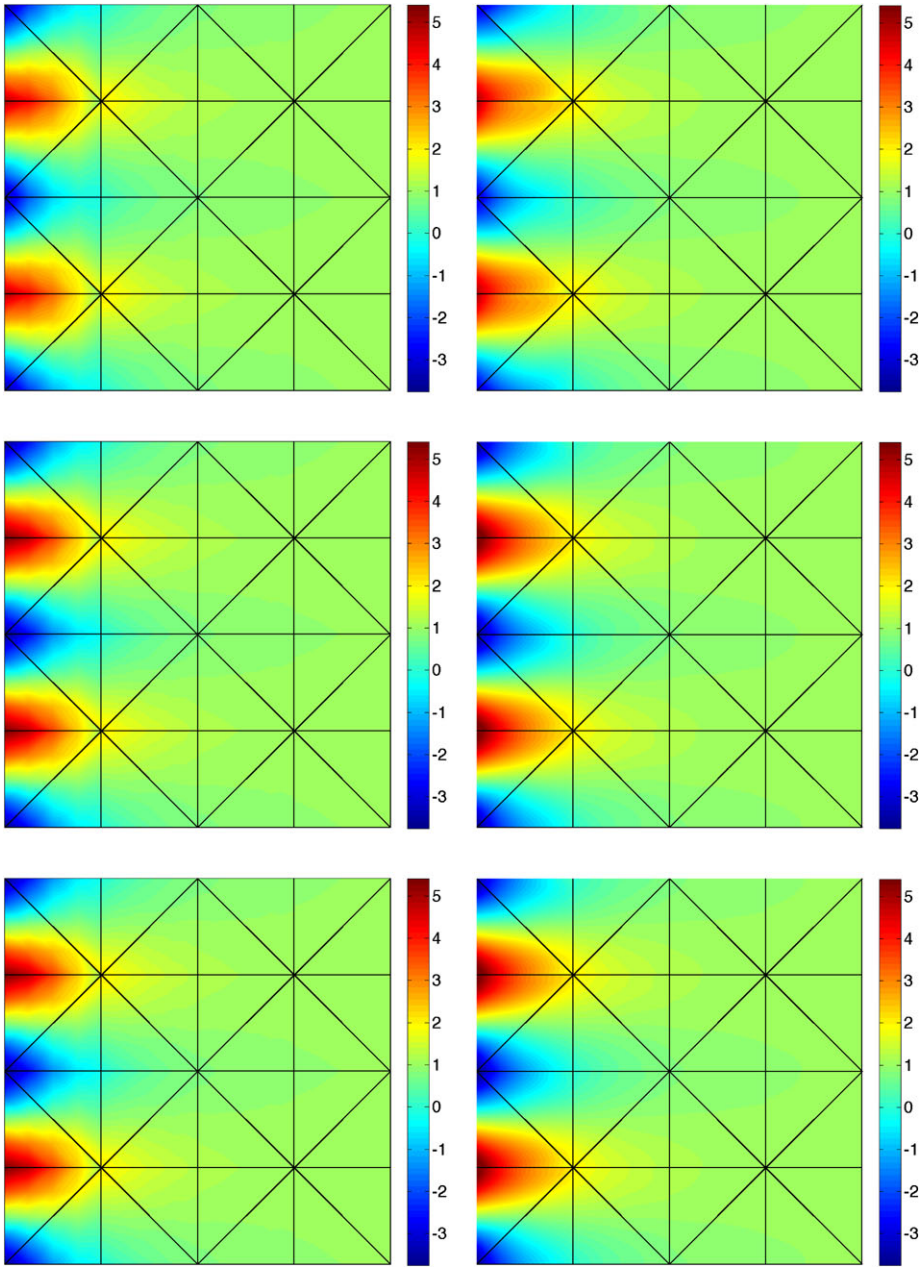

Fig. 1 The horizontal component of the approximate velocity (*left*) and postprocessed velocity (*right*) obtained using $k = 2$ on the same mesh: $\boldsymbol{u}_h^{\mathrm{V}}$ (*top left*) and $\boldsymbol{u}_h^{\mathrm{V},*}$ (*top right*), $\boldsymbol{u}_h^{\mathrm{S}}$ (*middle left*) and $\boldsymbol{u}_h^{\mathrm{S},*}$ (*middle right*), and $\boldsymbol{u}_h^{\mathrm{G}}$ (*bottom left*) and $\boldsymbol{u}_h^{\mathrm{G},*}$ (*bottom right*)

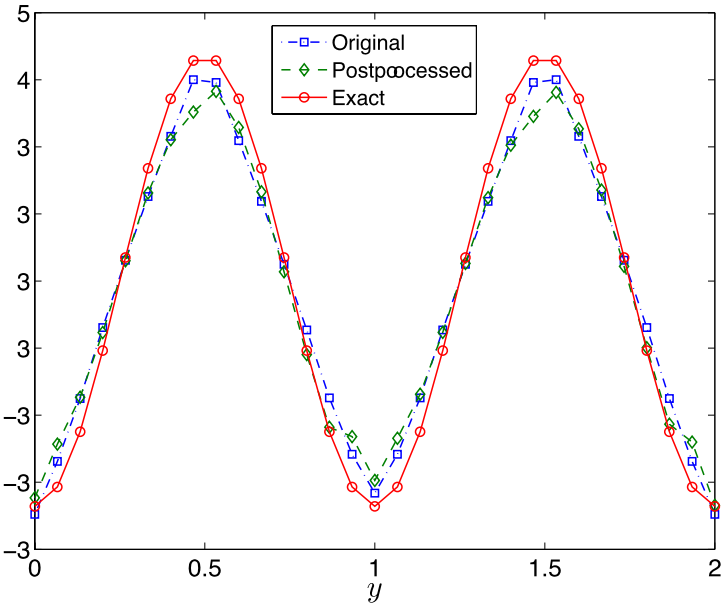

Fig. 2 Comparison of the original horizontal velocity and postprocessed horizontal velocity along the line $x = -0.4$ for the vorticity-based HDG method

Table 5 The condition number ratio R, see (18), as a function of h, k, and Δt

Degree	Mesh	Gradient-based HDG method					Vorticity-based HDG method				
k	h	$\Delta t = 1$	$\Delta t = 2$	$\Delta t = 4$	$\Delta t = 8$	$\Delta t = 16$	$\Delta t = 1$	$\Delta t = 2$	$\Delta t = 4$	$\Delta t = 8$	$\Delta t = 16$
1	2	2.16	2.08	2.00	2.00	1.96	1.30	1.29	1.29	1.29	1.29
	4	2.04	2.00	2.00	2.00	2.00	1.48	1.53	1.55	1.56	1.57
	8	1.92	1.88	1.88	1.88	1.88	1.54	1.60	1.63	1.64	1.65
	16	1.88	1.84	1.84	1.84	1.84	1.60	1.66	1.70	1.72	1.73
	32	1.84	1.84	1.84	1.84	1.84	1.64	1.71	1.74	1.76	1.77
2	2	2.28	2.16	2.12	2.08	2.08	1.36	1.36	1.36	1.36	1.35
	4	2.08	2.00	1.96	1.96	1.92	1.49	1.51	1.52	1.53	1.53
	8	1.96	1.92	1.88	1.88	1.88	1.59	1.62	1.64	1.65	1.65
	16	1.96	1.88	1.88	1.84	1.84	1.66	1.70	1.72	1.73	1.73
	32	1.92	1.88	1.88	1.84	1.84	1.70	1.74	1.76	1.77	1.78

4.2 Example with Singular Solution

To study the limitations imposed by singularities of the geometry, we consider the L-shaped domain $\Omega = \Omega_0 \backslash \Omega_1$, where $\Omega_0 \equiv (-1, 1) \times (-1, 1)$ and $\Omega_1 \equiv (0, 1) \times (-1, 0)$ are the square domains. Since Ω has a reentrant corner at the point $(0, 0)$, the exact solution is singular at

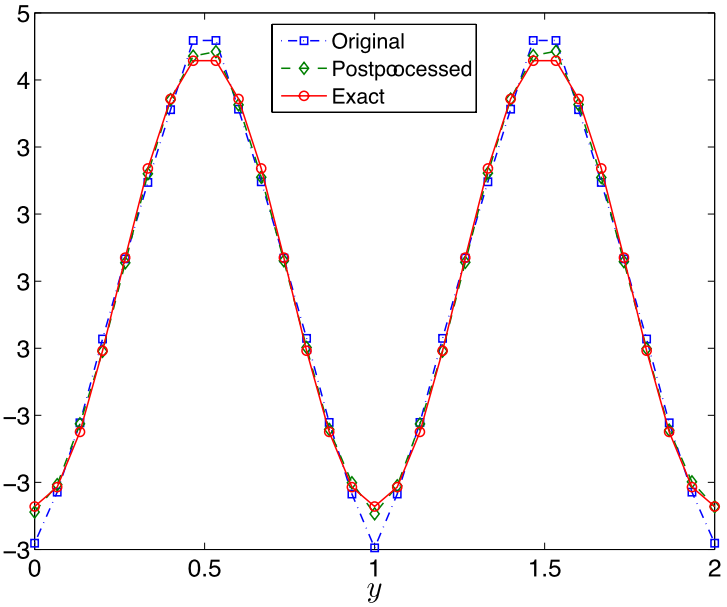

Fig. 3 Comparison of the original horizontal velocity and postprocessed horizontal velocity along the line $x = -0.4$ for the stress-based HDG method

Table 6 The number of iterations required for convergence as a function of h, k, and Δt

Degree	Mesh	Gradient-based HDG method					Vorticity-based HDG method				
k	h	$\Delta t = 1$	$\Delta t = 2$	$\Delta t = 4$	$\Delta t = 8$	$\Delta t = 16$	$\Delta t = 1$	$\Delta t = 2$	$\Delta t = 4$	$\Delta t = 8$	$\Delta t = 16$
	2	16	12	9	7	6	14	11	8	7	6
	4	16	12	9	7	6	14	10	8	7	6
1	8	16	12	9	7	6	14	10	8	7	6
	16	16	12	9	8	6	14	11	8	7	6
	32	17	12	9	8	6	14	11	8	7	6
	2	16	12	9	7	6	14	10	8	7	6
	4	16	12	9	7	6	14	10	8	7	6
2	8	16	12	9	8	6	14	11	8	7	6
	16	17	12	9	8	6	15	11	9	7	6
	32	17	12	9	8	6	15	11	9	7	6

the origin. This example proposed in [18] has the exact solution

$$u_1 = r^{\lambda}\left((1+\lambda)\sin(\theta)\Phi(\theta) + \cos(\theta)\Phi'(\theta)\right),$$
$$u_2 = r^{\lambda}\left(-(1+\lambda)\cos(\theta)\Phi(\theta) + \sin(\theta)\Phi'(\theta)\right),$$
$$p = -r^{\lambda-1}\left((1+\lambda)^2\Phi'(\theta) + \Phi'''(\theta)\right)/(1-\lambda),$$

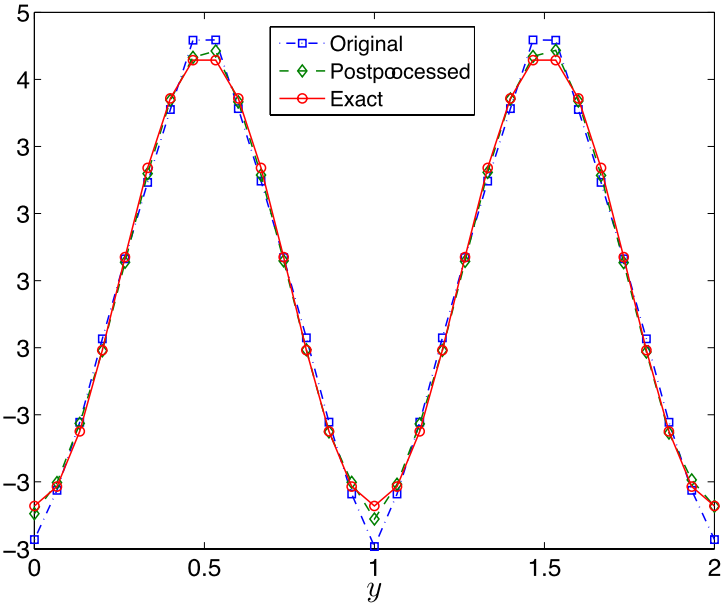

Fig. 4 Comparison of the original horizontal velocity and postprocessed horizontal velocity along the line $x = -0.4$ for the gradient-based HDG method

where

$$\Phi(\theta) = \sin((1+\lambda)\theta)\cos(\omega\lambda)/(1+\lambda) - \cos((1+\lambda)\theta) - \sin((1-\lambda)\theta)\cos(\omega\lambda)/(1-\lambda) + \cos((1-\lambda)\theta),$$

with $\omega = 3\pi/2$ and $\lambda = 0.54448373678246$. Here (r, θ) are the polar coordinates. Note that the velocity gradient $\nabla \boldsymbol{u}$ and the pressure p blow up at the re-entrant corner.

We present in Table 7 the history of convergence of the original and postprocessed velocities for the three HDG methods. We observe that both the original and postprocessed velocities converge with order at most 1 due to the singularity at the re-entrant corner. Comparing the results of the three methods we see that the gradient-based method yields the smallest errors, while the vorticity-based method has the largest errors. These results confirm the superiority of the gradient-based HDG method for numerically solving the Stokes system.

5 Conclusions

We have compared three different HDG methods for numerically solving the velocity-pressure-gradient, velocity-pressure-stress, and velocity-pressure-vorticity formulations of Stokes flow. All the present methods yield optimal convergence for the approximate variables. Moreover, they have the same global degrees of freedom and thus the same computational cost and memory storage. In terms of accuracy, our numerical experiments show that the gradient-based HDG method is the best method since it provides the most accurate

Table 7 History of convergence of the original and the postprocessed velocity for the Stokes problem with singular solution

Degree	Mesh	$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{G}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{G},\star}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{S}}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{S},\star}\|_{\mathcal{T}_h}$		$\|\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{V}}\|_{\mathcal{T}_h}$		$\boldsymbol{u}-\boldsymbol{u}_h^{\mathrm{V},\star}\|_{\mathcal{T}_h}$	
k	h^{-1}	error	order	error	order	error	order	error	order	error	order	error	order
0	2	5.47e-1	–	5.95e-1	–	4.96e-1	–	5.56e-1	–	6.66e-1	–	7.51e-1	–
	4	3.29e-1	0.74	3.35e-1	0.83	3.24e-1	0.61	3.50e-1	0.67	5.14e-1	0.38	5.55e-1	0.44
	8	1.79e-1	0.88	1.79e-1	0.90	2.15e-1	0.59	2.38e-1	0.56	3.97e-1	0.37	4.28e-1	0.38
	16	9.27e-2	0.95	9.23e-2	0.96	1.40e-1	0.62	1.58e-1	0.59	3.00e-1	0.41	3.22e-1	0.41
	32	4.72e-2	0.98	4.69e-2	0.98	9.02e-2	0.64	1.03e-1	0.63	2.19e-1	0.45	2.34e-1	0.46
1	2	1.01e-1	–	1.05e-1	–	8.40e-2	–	9.03e-2	–	3.19e-1	–	4.15e-1	–
	4	4.47e-2	1.17	4.59e-2	1.19	3.99e-2	1.08	4.65e-2	0.96	2.08e-1	0.62	2.64e-1	0.65
	8	1.98e-2	1.18	2.01e-2	1.19	1.91e-2	1.06	2.24e-2	1.05	1.22e-1	0.77	1.53e-1	0.79
	16	8.89e-3	1.15	8.99e-3	1.16	8.94e-3	1.09	1.03e-2	1.11	6.87e-2	0.82	8.52e-2	0.84
	32	4.07e-3	1.13	4.09e-3	1.14	4.16e-3	1.10	4.71e-3	1.13	3.84e-2	0.84	4.69e-2	0.86
2	2	4.37e-2	–	5.02e-2	–	4.25e-2	–	5.85e-2	–	2.02e-1	–	2.75e-1	–
	4	1.95e-2	1.17	2.12e-2	1.24	2.07e-2	1.04	2.9e-2	1.02	1.12e-1	0.86	1.46e-1	0.92
	8	8.83e-3	1.14	9.29e-3	1.19	9.65e-3	1.10	1.34e-2	1.11	6.02e-2	0.89	7.51e-2	0.95
	16	4.06e-3	1.12	4.18e-3	1.15	4.45e-3	1.12	6.03e-3	1.15	3.22e-2	0.90	3.90e-2	0.95
	32	1.89e-3	1.11	1.91e-3	1.13	2.04e-3	1.13	2.67e-3	1.18	1.72e-2	0.91	2.05e-2	0.93

results and a postprocessed velocity which converges with order $k+2$ for $k \geq 1$. The stress-based HDG method performs better than the vorticity-based HDG method and it yields a postprocessed velocity converging with order $k+1$ for $k = 0, 1$, but tends to converge with order $k+2$ for $k = 2, 4$. For the vorticity-based HDG method, the postprocessed velocity converges with order $k+1$ only.

The extension of this work to the numerical solution of the incompressible Navier-Stokes equations will be presented in a forthcoming paper.

Acknowledgements J. Peraire and N.C. Nguyen would like to acknowledge the Singapore-MIT Alliance for partially supporting this work. B. Cockburn would like to acknowledge the National Science Foundation for partially supporting this work through Grant DMS-0712955.

References

1. Baker, G.A., Jureidini, W.N., Karakashian, O.A.: Piecewise solenoidal vector fields and the Stokes problem. SIAM J. Numer. Anal. **27**, 1466–1485 (1990)
2. Brezzi, F., Fortin, M.: Mixed and Hybrid Finite Element Methods. Springer, Berlin (1991)
3. Carrero, J., Cockburn, B., Schötzau, D.: Hybridized globally divergence-free LDG methods, I: the Stokes problem. Math. Comput. **75**, 533–563 (2006)
4. Cockburn, B., Gopalakrishnan, J.: Incompressible finite elements via hybridization, part I: the stokes system in two space dimensions. SIAM J. Numer. Anal. **43**(4), 1627–1650 (2005)
5. Cockburn, B., Gopalakrishnan, J.: Incompressible finite elements via hybridization, part II: the Stokes system in three space dimensions. SIAM J. Numer. Anal. **43**(4), 1651–1672 (2005)
6. Cockburn, B., Gopalakrishnan, J.: The derivation of hybridizable discontinuous Galerkin methods for Stokes flow. SIAM J. Numer. Anal. **47**, 1092–1125 (2009)
7. Cockburn, B., Gopalakrishnan, J., Guzmán, J.: A new elasticity element made for enforcing weak stress symmetry. Math. Comput. (to appear)
8. Cockburn, B., Gopalakrishnan, J., Nguyen, N.C., Peraire, J., Sayas, F.-J.: Analysis of an HDG method for Stokes flow. Math. Comput. (to appear)
9. Cockburn, B., Kanschat, G., Schötzau, D.: A locally conservative LDG method for the incompressible Navier-Stokes equations. Math. Comput. **74**, 1067–1095 (2005)
10. Cockburn, B., Kanschat, G., Schötzau, D.: A note on discontinuous Galerkin divergence-free solutions of the Navier-Stokes equations. J. Sci. Comput. **31**, 61–73 (2007)
11. Cockburn, B., Kanschat, G., Schötzau, D., Schwab, C.: Local discontinuous Galerkin methods for the Stokes system. SIAM J. Numer. Anal. **40**(1), 319–343 (2002)
12. Fortin, M., Glowinski, R.: Augmented Lagrangian Methods. Studies in Mathematics and its Applications, vol. 15. North-Holland, Amsterdam (1983). Applications to the numerical solution of boundary value problems, Translated from the French by B. Hunt and D.C. Spicer
13. Kovasznay, L.I.G.: Laminar flow behind two-dimensional grid. Proc. Camb. Philos. Soc. **44**, 58–62 (1948)
14. Montlaur, A., Fernández-Méndez, S., Huerta, A.: Discontinuous Galerkin methods for the Stokes equations using divergence-free approximations. Int. J. Numer. Methods Fluids **57**, 1071–1092 (2008)
15. Nédélec, J.-C.: A new family of mixed finite elements in $\boldsymbol{R}^3$. Numer. Math. **50**, 57–81 (1986)
16. Nguyen, N.C., Peraire, J., Cockburn, B.: A hybridizable discontinuous Galerkin method for Stokes flow. Comput. Methods Appl. Mech. Eng. **199**, 582–597 (2010)
17. Soon, S.-C., Cockburn, B., Stolarski, H.: A hybridizable discontinuous Galerkin method for linear elasticity. Int. J. Numer. Methods Eng. **80**(8), 1058–1092 (2009)
18. Verfürth, R.: A posteriori error estimators for the Stokes equations. Numer. Math. **55**, 309–325 (1989)
19. Wang, J., Ye, X.: New Finite element methods in computational fluid dynamics by $H(\mathrm{div})$ elements. SIAM J. Numer. Anal. **45**, 1269–1286 (2007)