

Hybridizable Discontinuous Galerkin Methods

N.C. Nguyen, J. Peraire, and B. Cockburn

Abstract We present an overview of recent developments of HDG methods for numerically solving partial differential equations in fluid mechanics.

1 Background

In recent years, discontinuous Galerkin (DG) finite element methods have emerged as a competitive alternative for solving nonlinear hyperbolic systems of conservation laws. The advantages of the DG methods over classical finite difference and finite volume methods are well-documented in the literature: the DG methods work well on arbitrary meshes, result in stable high-order accurate discretizations of the convective and diffusive operators, allow for a simple and unambiguous imposition of boundary conditions and are very flexible to parallelization and adaptivity. Despite all these advantages, DG methods have not yet made a significant impact for practical applications. This is largely due to the high computational cost associated to them when compared to finite differences or finite volume schemes.

The hybridizable discontinuous Galerkin (HDG) methods were recently introduced to try to address this issue. In this paper, we present an overview of the recent developments of these methods with *implicit time-marching integration* as applied to some basic models in fluid mechanics.

The HDG methods retain the advantages of standard DG methods and result in a significantly reduced degree of freedom count, therefore allowing for a substantial reduction in the computational cost and memory storage. Hybridizable DG methods were initially developed for elliptic problems [4, 5, 9, 10, 12, 13] and have already

N.C. Nguyen and J. Peraire (✉)

Department of Aeronautics, MIT, 77 Massachusetts Avenue, Cambridge, MA 02139, USA
e-mail: cuongng@mit.edu, peraire@mit.edu

B. Cockburn

Department of Mathematics, University of Minnesota, Minneapolis, MN 55455, USA
e-mail: cockburn@math.umn.edu

been developed and demonstrated for linear and nonlinear convection-diffusion problems [6, 20, 21], linear elasticity [27], and incompressible flow [7, 11, 22–24].

The HDG methods we consider have the following main advantages over many existing discontinuous Galerkin methods:

- **Reduced number of globally coupled degrees of freedom** Unlike many other DG methods (analyzed in [1]) which result in a final system involving all the degrees of freedom of the approximate field variables, the HDG methods produce a final system in terms of the degrees of freedom of the *approximate traces* of the field variables. Since the approximate traces are defined on the element borders only, the HDG methods have significantly less the globally coupled unknowns as other DG methods. In fact, a variant of the HDG method – the Embedded DG method (EDG) [12, 17]) – has the same number of globally coupled unknowns than a standard continuous Galerkin method. This large reduction in the degrees of freedom can lead to significant savings for both computational time and memory storage.
- **Superconvergence** For convection-diffusion problems, the HDG methods provide *optimal convergence* for the approximation of the gradient – a special convergence property of the HDG methods for diffusion problems – whereas, for all of the DG methods studied in [1], as well as the standard continuous Galerkin approach the approximate gradient converges *suboptimally*. For incompressible flows, the approximate velocity, pressure, velocity gradient, and vorticity converge with the optimal order. This has to be contrasted with the fact that *all* the other DG methods display the suboptimal order of convergence for the approximate pressure, velocity gradient, and vorticity. Moreover, the HDG methods have superconvergence properties for the numerical traces and the average of the approximate variables.
- **Local postprocessing** Based on the optimal convergence and superconvergence of the HDG methods, local postprocessing can be developed to increase by one the spatial order of convergence of the numerical solution. For incompressible flows, local postprocessing can be employed to obtain a new approximation of the velocity which is exactly divergence-free, $\mathbf{H}(\text{div})$ -conforming, and converges with an additional order. For time-dependent problems, postprocessing only needs to be done at those time levels for which a more accurate result is desired. Moreover, since the postprocessing is performed at the element level, it is less expensive than the solution procedure.
- **Geometric flexibility and mesh adaptation** The HDG methods can be implemented on general unstructured meshes and are well suited to handle h/p adaptivity since grid refinement or coarsening can be achieved without taking into account the continuity restrictions typical of conforming methods, and since different order of approximations can be used on different elements/subdomains. Adaptivity is of particular importance in compressible flow given the complexity of the solution structure and geometries involved.
- **Parallelization** The HDG methods remain highly parallelizable even when implicit time integration is used since the local problems are formulated at the

element or subdomain level, they can be solved independently for each of the subdomain blocks. For the global problem, the iterative techniques with p-multigrid and block ILU preconditioning developed for DG methods can also be applied here [26].

We attempt to give an overview of recent developments of the HDG methods for fluid dynamics. In Sect. 2 we describe the basic ideas of HDG methodology for a convection-diffusion model equation: a mixed formulation of the model equation, a characterization of the numerical solution in terms of the approximate trace, relationship between the HDG method and the standard DG methods, the choice of the stabilization parameter, and the local postprocessing to improve the order of convergence. In Sect. 3 we show how the main ideas can be extended to time-dependent and nonlinear convection-diffusion problems, Stokes flows, and incompressible Navier–Stokes equations. In Sect. 4 we present numerical results for fluid dynamics to demonstrate the performance and accuracy of the HDG method. Finally, in Sect. 5, we end the paper with some concluding remarks on future developments.

2 The HDG Method

2.1 The Convection-Diffusion Model Equation

We will describe the main ideas behind the hybridized discontinuous Galerkin method using the linear convection-diffusion equation as a model problem

$$\nabla \cdot (cu) - \nabla \cdot (\kappa \nabla u) = f, \quad \text{in } \Omega, \quad (1)$$

with boundary conditions

$$\begin{aligned} u &= g_D, \text{ on } \Gamma_D, \\ (-\kappa \nabla u + cu) \cdot \mathbf{n} &= g_N, \text{ on } \Gamma_N. \end{aligned} \quad (2)$$

Here u is the field variable, c and $\kappa > 0$ are constant and f , g_D and g_N are given (see [20] for additional details).

We introduce the auxiliary variable $\mathbf{q} = -\kappa \nabla u$ and rewrite the above equation as a first order system of equations

$$\begin{aligned} \mathbf{q} + \kappa \nabla u &= 0, \text{ in } \Omega, \\ \nabla \cdot (cu + \mathbf{q}) &= f, \text{ in } \Omega, \end{aligned} \quad (3)$$

with boundary conditions

$$\begin{aligned} u &= g_D, \text{ on } \Gamma_D, \\ (\mathbf{q} + cu) \cdot \mathbf{n} &= g_N, \text{ on } \Gamma_N. \end{aligned} \quad (4)$$

Next, we introduce the notation necessary for the description of the HDG method.

2.2 Mesh and Trace Operators

Let \mathcal{T}_h be a collection of disjoint elements that partition Ω . We denote by $\partial\mathcal{T}_h$ the set $\{\partial K : K \in \mathcal{T}_h\}$. 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. Let \mathcal{E}_h^o and \mathcal{E}_h^∂ denote the set of interior and boundary faces, respectively. We denote by \mathcal{E}_h the union of \mathcal{E}_h^o and \mathcal{E}_h^∂ .

Let \mathbf{n}^+ and \mathbf{n}^- be the outward unit normals of ∂K^+ and ∂K^- , respectively, and let (\mathbf{q}^\pm, u^\pm) be the traces of (\mathbf{q}, u) on F from the interior of K^\pm . Then, we define the mean values $\{\cdot\}$ and jumps $[[\cdot]]$ as follows. For $F \in \mathcal{E}_h^o$, we set

$$\begin{aligned} \{\mathbf{q}\} &= (\mathbf{q}^+ + \mathbf{q}^-)/2 & \{u\} &= (u^+ + u^-)/2, \\ [[\mathbf{q} \cdot \mathbf{n}]] &= \mathbf{q}^+ \cdot \mathbf{n}^+ + \mathbf{q}^- \cdot \mathbf{n}^- & [[un]] &= u^+ \mathbf{n}^+ + u^- \mathbf{n}^-. \end{aligned}$$

For $F \in \mathcal{E}_h^\partial$, the set of boundary edges on which \mathbf{q} and u are singled value, we set

$$\begin{aligned} \{\mathbf{q}\} &= \mathbf{q} & \{u\} &= u, \\ [[\mathbf{q} \cdot \mathbf{n}]] &= \mathbf{q} \cdot \mathbf{n} & [[un]] &= un. \end{aligned}$$

Note that the jump in u is a vector, but the jump in \mathbf{q} is a scalar. Furthermore, the jumps will be zero for a continuous function.

2.3 Approximation Spaces

Let $\mathcal{P}_m(D)$ denote the set of polynomials of degree at most m on a domain D . We introduce discontinuous finite element spaces

$$W_h = \{w \in L^2(\Omega) : w|_K \in \mathcal{P}_k(K), \forall K \in \mathcal{T}_h\},$$

and

$$V_h = \{\mathbf{v} \in (L^2(\Omega))^d : \mathbf{v}|_K \in (\mathcal{P}_k(K))^d, \forall K \in \mathcal{T}_h\}.$$

Here $L^2(D)$ is the space of square integrable functions on D . In addition, we introduce a traced finite element space

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

We also set $M_h(g_D) = \{\mu \in M_h : \mu = \mathbf{P}g_D \text{ on } \Gamma_D\}$, where \mathbf{P} denotes the L^2 -projection into the space $\{\mu|_{\partial\Omega} \forall \mu \in M_h\}$. Note that M_h consists of functions

which are continuous inside the faces (or edges) $F \in \mathcal{E}_h$ and discontinuous at their borders.

For functions \mathbf{w} and \mathbf{v} in $(L^2(D))^d$, we denote $(\mathbf{w}, \mathbf{v})_D = \int_D \mathbf{w} \cdot \mathbf{v}$. For functions u and v in $L^2(D)$, we denote $(u, v)_D = \int_D uv$ if D is a domain in \mathbb{R}^d and $\langle u, v \rangle_D = \int_D uv$ if D is a domain in \mathbb{R}^{d-1} . We finally introduce

$$(w, v)_{\mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} (w, v)_K, \quad \langle \zeta, \rho \rangle_{\partial \mathcal{T}_h} = \sum_{K \in \mathcal{T}_h} \langle w, v \rangle_{\partial K}, \quad \langle \mu, \eta \rangle_{\mathcal{E}_h} = \sum_{F \in \mathcal{E}_h} \langle \mu, \eta \rangle_F,$$

for functions w, v defined on \mathcal{T}_h , ζ, ρ defined on $\partial \mathcal{T}_h$, and μ, η defined on \mathcal{E}_h .

2.4 HDG Formulation

We seek an approximation $(\mathbf{q}_h, u_h) \in \mathbf{V}_h \times W_h$ such that for all $K \in \mathcal{T}_h$,

$$\begin{aligned} (\kappa^{-1} \mathbf{q}_h, \mathbf{v})_K - (u_h, \nabla \cdot \mathbf{v})_K + \langle \widehat{u}_h, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial K} &= 0, & \forall \mathbf{v} \in (\mathcal{P}_k(K))^d, \\ -(\mathbf{c}u_h + \mathbf{q}_h, \nabla w)_K + \langle (\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h) \cdot \mathbf{n}, w \rangle_{\partial K} &= (f, w)_K, & \forall w \in \mathcal{P}_k(K). \end{aligned} \quad (5)$$

Here, the numerical traces $\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h$ and \widehat{u}_h are approximations to $\mathbf{c}u - \kappa \nabla u$ and u over ∂K , respectively. Next, we express (\mathbf{q}_h, u_h) in terms of \widehat{u}_h only. To this end, we consider numerical traces $\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h$ of the form

$$\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h = \mathbf{c}\widehat{u}_h + \mathbf{q}_h + \tau(u_h - \widehat{u}_h)\mathbf{n}, \quad \text{on } \partial K. \quad (6)$$

Here, τ is the so-called *local stabilization parameter*; it has an important effect on both the stability and accuracy of the resulting scheme. The selection of the value of the parameter τ will be described below. Note that both $\widehat{\mathbf{c}}u_h$ and $\mathbf{c}\widehat{u}_h$ are different approximations to the same quantity $\mathbf{c}u$ and that the former is defined in terms of the latter.

We next express \widehat{u}_h in terms of the boundary data g_D and a new variable $\lambda_h \in M_h(0)$ as

$$\widehat{u}_h = \begin{cases} \mathbf{P}g_D, & \text{on } \mathcal{E}_h \cap \Gamma_D, \\ \lambda_h, & \text{on } \mathcal{E}_h \setminus \Gamma_D. \end{cases}$$

By adding the contributions of (5) over all the elements and enforcing the continuity of the normal component of the numerical flux, we arrive at the following problem: find an approximation $(\mathbf{q}_h, u_h, \lambda_h) \in \mathbf{V}_h \times W_h \times M_h(0)$ such that

$$\begin{aligned} (\kappa^{-1} \mathbf{q}_h, \mathbf{v})_{\mathcal{T}_h} - (u_h, \nabla \cdot \mathbf{v})_{\mathcal{T}_h} + \langle \lambda_h, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial \mathcal{T}_h} &= -\langle g_D, \mathbf{v} \cdot \mathbf{n} \rangle_{\Gamma_D}, & \forall \mathbf{v} \in \mathbf{V}_h, \\ -(\mathbf{c}u_h + \mathbf{q}_h, \nabla w)_{\mathcal{T}_h} + \langle (\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h) \cdot \mathbf{n}, w \rangle_{\partial \mathcal{T}_h} &= (f, w)_{\mathcal{T}_h}, & \forall w \in W_h, \\ \langle [(\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h) \cdot \mathbf{n}], \mu \rangle_{\mathcal{E}_h} &= \langle g_N, \mu \rangle_{\Gamma_N}, & \forall \mu \in M_h(0). \end{aligned} \quad (7)$$

Note that the Dirichlet boundary condition has been enforced by requiring that $\widehat{u}_h = \mathbf{P}g_D$ on $\mathcal{E}_h \cap \Gamma_D$, whereas the continuity of the normal component of $\widehat{c}u_h + \widehat{q}_h$ is enforced explicitly by the last equation.

We observe that λ_h is uniquely defined over each edge since λ_h belongs to M_h . Furthermore, if $\llbracket (\widehat{c}u_h + \widehat{q}_h) \cdot \mathbf{n} \rrbracket$ belongs to M_h , then the last equation (7) simply states that $\llbracket (\widehat{c}u_h + \widehat{q}_h) \cdot \mathbf{n} \rrbracket = 0$ pointwise over $\mathcal{E}_h \setminus \Gamma_N$ and that $(\widehat{c}u_h + \widehat{q}_h) \cdot \mathbf{n} = \mathbf{P}g_N$ on Γ_N ; in other words, the normal component of the numerical trace $\widehat{c}u_h + \widehat{q}_h$ is single-valued. Hence, both λ_h and $\widehat{c}u_h + \widehat{q}_h$ are conservative fluxes according to the definition in [1]. Note that our numerical traces remain conservative even when the diffusion coefficient κ is discontinuous at the interior element interface.

We note that, due to the discontinuous nature of both V_h and W_h , the first two equations in (6) can be used to eliminate both \mathbf{q}_h and u_h to obtain a weak formulation in terms of λ_h only and thus a global system of equations involving the degrees of freedom of λ_h , as described below.

2.5 Characterization of the Numerical Trace

We first introduce the so-called local solver which associate to each function $(\mathbf{m}, f) \in M_h \times L^2(\Omega)$, the pair $(\mathbf{q}_h^{\mathbf{m},f}, u_h^{\mathbf{m},f})$ on Ω whose restriction to each element K is in $(\mathcal{P}_k(K))^d \times \mathcal{P}_k(K)$ and satisfies

$$(\kappa^{-1} \mathbf{q}_h^{\mathbf{m},f}, \mathbf{v})_K - (u_h^{\mathbf{m},f}, \nabla \cdot \mathbf{v})_K = - \langle \mathbf{m}, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial K}, \quad (8a)$$

$$-(c u_h^{\mathbf{m},f} + \mathbf{q}_h^{\mathbf{m},f}, \nabla w)_K + \left\langle (\widehat{c}u_h^{\mathbf{m},f} + \widehat{q}_h^{\mathbf{m},f}) \cdot \mathbf{n}, w \right\rangle_{\partial K} = (f, w)_K, \quad (8b)$$

for all $(\mathbf{v}, w) \in (\mathcal{P}_k(K))^d \times \mathcal{P}_k(K)$, where

$$\widehat{c}u_h^{\mathbf{m},f} + \widehat{q}_h^{\mathbf{m},f} = c\mathbf{m} + \mathbf{q}_h^{\mathbf{m},f} + \tau(u_h^{\mathbf{m},f} - \mathbf{m})\mathbf{n}. \quad (8c)$$

It is now clear, see (7), that the approximate solution $(\mathbf{q}_h, u_h) \in V_h \times W_h$ satisfies

$$\mathbf{1}\mathbf{q}_h = \mathbf{q}_h^{\lambda_h, f}, \quad u_h = u_h^{\lambda_h, f}, \quad (9a)$$

where $\lambda_h \in M_h(0)$ is such that

$$\left\langle \llbracket (\widehat{c}u_h^{\lambda_h, f} + \widehat{q}_h^{\lambda_h, f}) \cdot \mathbf{n} \rrbracket, \mu \right\rangle_{\mathcal{E}_h} = \langle g_N, \mu \rangle_{\Gamma_N}, \quad \forall \mu \in M_h(0). \quad (9b)$$

We next show that we can eliminate \mathbf{q}_h and u_h from the above equations to obtain a weak formulation in terms of λ_h only.

Let $(\mathbf{q}_h^{\mathbf{m},0}, u_h^{\mathbf{m},0})$ (respectively, $(\mathbf{q}_h^{0,f}, u_h^{0,f})$) solve (8) when we set $f = 0$ (respectively, $\mathbf{m} = 0$). If, for all η and $\mu \in M_h$, we set

$$a_h(\eta, \mu) = - \left\langle \llbracket (\widehat{\mathbf{c}}\widehat{u}_h^{\eta,0} + \widehat{\mathbf{q}}_h^{\eta,0}) \cdot \mathbf{n} \rrbracket, \mu \right\rangle_{\mathcal{E}_h}, \quad (10a)$$

$$b_h(\mu) = \left\langle \llbracket (\widehat{\mathbf{c}}\widehat{u}_h^{0,f} + \widehat{\mathbf{q}}_h^{0,f}) \cdot \mathbf{n} \rrbracket, \mu \right\rangle_{\mathcal{E}_h}, \quad (10b)$$

we have from (9b) and linearity of the problem (8) that the function $\lambda_h \in M_h(0)$ is the solution of the variational formulation

$$a_h(\lambda_h, \mu) = b_h(\mu) - \langle g_N, \mu \rangle_{\Gamma_N}, \quad \forall \mu \in M_h(0). \quad (11)$$

The existence and uniqueness of the numerical trace λ_h is presented in [20].

The above weak formulation gives rise to a matrix system of the form

$$\mathbb{K} \Lambda = \mathbb{F}, \quad (12)$$

where Λ is the vector of degrees of freedom of λ_h , \mathbb{K} is the matrix associated with the bilinear form $a_h(\cdot, \cdot)$, and \mathbb{F} the vector associated with the linear form $b_h(\cdot) - \langle g_N, \cdot \rangle_{\Gamma_N}$. Note that since

$$a_h(\eta, \mu) = - \left\langle (\widehat{\mathbf{c}}\widehat{u}_h^{\eta,0} + \widehat{\mathbf{q}}_h^{\eta,0}) \cdot \mathbf{n}, \mu \right\rangle_{\partial \mathcal{T}_h},$$

we can easily deduce that if the support of η is the interior face $F = \partial K^+ \cap \partial K^-$, or the boundary face $F = \partial K \cap \partial \Omega$, then $a_h(\eta, \mu) = 0$ when the support of μ does not intersect $\partial K^+ \cup \partial K^-$, or ∂K , respectively. Thus, the matrix \mathbb{K} has a block-structure of blocks of square matrices of order $\dim \mathcal{P}_k$. In each block-row or block-column, there are at most five non-zero blocks when the elements are triangles, and at most seven non-zero blocks in three space dimension.

The construction of the matrix system (12) can be carried out in two steps. In the first step, we solve the local problem (8) for every element $K \in \mathcal{T}_h$. In the second step, we evaluate the face integrals (10) by using the standard finite element quadrature rule and assembly. This procedure can be implemented for arbitrary polynomial degrees. The detailed implementation discussed in [20] is omitted here to save space.

2.6 Relation to Other DG Methods

In order to derive an explicit expression for the numerical traces in terms of (u_h, \mathbf{q}_h) , we proceed as follows. Since the conservativity condition implies $\llbracket (\widehat{\mathbf{c}}\widehat{u}_h + \widehat{\mathbf{q}}_h) \cdot \mathbf{n} \rrbracket = 0$ pointwise, we have, using expression (6), that

$$\llbracket \mathbf{q}_h \cdot \mathbf{n} \rrbracket + \tau^+ u_h^+ + \tau^- u_h^- - (\tau^+ + \tau^-) \lambda_h = 0, \quad \text{on } \mathcal{E}_h^o.$$

Solving for λ_h and inserting the result into the expression for $\widehat{\mathbf{c}}\widehat{u}_h + \widehat{\mathbf{q}}_h$ (6), we obtain on \mathcal{E}_h^o

$$\begin{aligned}\lambda_h &= \frac{\tau^+}{\tau^+ + \tau^-} u_h^+ + \frac{\tau^-}{\tau^+ + \tau^-} u_h^- + \left(\frac{1}{\tau^+ + \tau^-} \right) \llbracket \mathbf{q}_h \cdot \mathbf{n} \rrbracket, \\ \widehat{\mathbf{c}} u_h + \widehat{\mathbf{q}}_h &= \mathbf{c} \lambda_h + \frac{\tau^-}{\tau^+ + \tau^-} \mathbf{q}_h^+ + \frac{\tau^+}{\tau^+ + \tau^-} \mathbf{q}_h^- + \left(\frac{\tau^+ \tau^-}{\tau^+ + \tau^-} \right) \llbracket u_h \mathbf{n} \rrbracket.\end{aligned}\quad (13)$$

These expressions for the numerical traces highlight the relationship between the HDG method and the more standard DG methods, as discussed below.

In the convective limit we have $\kappa = 0$ and consequently $\mathbf{q}_h = 0$. In this case, the expressions (13) become

$$\begin{aligned}\lambda_h &= \frac{\tau^+}{\tau^+ + \tau^-} u_h^+ + \frac{\tau^-}{\tau^+ + \tau^-} u_h^-, \\ \widehat{\mathbf{c}} u_h \cdot \mathbf{n}^+ &= \frac{\tau^+}{\tau^+ + \tau^-} (\mathbf{c} \cdot \mathbf{n}^+ + \tau^-) u_h^+ + \frac{\tau^-}{\tau^+ + \tau^-} (\mathbf{c} \cdot \mathbf{n}^+ - \tau^+) u_h^-.\end{aligned}\quad (14)$$

In the diffusive limit $\mathbf{c} = \mathbf{0}$, expressions (13) become

$$\begin{aligned}\lambda_h &= \frac{\tau^+}{\tau^+ + \tau^-} u_h^+ + \frac{\tau^-}{\tau^+ + \tau^-} u_h^- + \left(\frac{1}{\tau^+ + \tau^-} \right) \llbracket \mathbf{q}_h \cdot \mathbf{n} \rrbracket, \\ \widehat{\mathbf{q}}_h &= \frac{\tau^-}{\tau^+ + \tau^-} \mathbf{q}_h^+ + \frac{\tau^+}{\tau^+ + \tau^-} \mathbf{q}_h^- + \left(\frac{\tau^+ \tau^-}{\tau^+ + \tau^-} \right) \llbracket u_h \mathbf{n} \rrbracket.\end{aligned}\quad (15)$$

This case has been originally studied in [3]; see also [4, 9, 13].

By rearranging terms these expressions can be transformed into the more standard form considered in [3],

$$\begin{aligned}\widehat{\mathbf{q}}_h &= \{\mathbf{q}_h\} + C_{11} \llbracket u_h \mathbf{n} \rrbracket + C_{12} \llbracket \mathbf{q}_h \cdot \mathbf{n} \rrbracket, \\ \lambda_h = \widehat{u}_h &= \{u_h\} - C_{12} \cdot \llbracket u_h \mathbf{n} \rrbracket + C_{22} \llbracket \mathbf{q}_h \cdot \mathbf{n} \rrbracket.\end{aligned}\quad (16)$$

where,

$$C_{11} = \left(\frac{\tau^+ \tau^-}{\tau^+ + \tau^-} \right), \quad C_{12} = \frac{1}{2} \left(\frac{\llbracket \tau \mathbf{n} \rrbracket}{\tau^+ + \tau^-} \right), \quad C_{22} = \left(\frac{1}{\tau^+ + \tau^-} \right).$$

It is interesting to note that for the simple choice of τ^\pm of order unity everywhere HDG methods yield optimal convergence rate of $k + 1$ for both the scalar variable and the flux, and that they display superconvergence properties of the scalar variable [4, 6, 13].

We point out that in the Local DG method [15], the trace λ_h is chosen to be independent of \mathbf{q}_h , that is $C_{22} = 0$. This has the advantage of allowing the degrees of freedom associated with the \mathbf{q}_h to be locally eliminated and a global system involving only the degrees of freedom associated to u_h is thus solved. However, using $C_{22} = 0$ yields suboptimal convergence for the approximate gradient. It is shown in [13] that the superconvergent schemes require that C_{22} be non-zero.

While this presents a serious inconvenience for LDG methods, for HDG methods this represents no difficulty.

2.7 The Local Stabilization Parameter τ

To account for the diffusion and convection effects our local stabilization parameter τ will take the following form

$$\tau = \tau_d + \tau_c$$

where τ_d and τ_c are the local stabilization parameters related to the diffusion and convection, respectively. This allows us to write each component of the numerical trace $\widehat{\mathbf{q}}_h + \widehat{\mathbf{c}}u_h$ as

$$\begin{aligned}\widehat{\mathbf{q}}_h &= \mathbf{q}_h + \tau_d(u_h - \lambda_h)\mathbf{n}, \\ \widehat{\mathbf{c}}u_h &= \mathbf{c}\lambda_h + \tau_c(u_h - \lambda_h)\mathbf{n}.\end{aligned}$$

A suitable expression for τ_c and τ_d is to take on each edge $\tau_c^+ = \tau_c^- = \eta_c$ and $\tau_d^+ = \tau_d^- = \eta_d$, where

$$\eta_c = |\mathbf{c} \cdot \mathbf{n}|, \quad \eta_d = \frac{\kappa}{\ell}, \quad (17)$$

where ℓ denotes a representative diffusive length scale which is typically of unity order and independent of the mesh size h . In this case, the expressions for the numerical traces becomes

$$\begin{aligned}\lambda_h &= \{u_h\} + \frac{1}{2\tau} \llbracket \mathbf{q}_h \cdot \mathbf{n} \rrbracket, \\ \widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h &= \mathbf{c}\lambda_h + \{\mathbf{q}_h\} + \frac{\tau}{2} \llbracket u_h \mathbf{n} \rrbracket.\end{aligned}$$

It can be shown that the HDG method is well-defined with the above choice of the stabilization parameter. Alternative forms for τ_c and τ_d can be found in [20].

Numerical experiment and theory (see [6, 20]) confirm that the above choice of the stabilization parameter is optimal in the sense that both the approximate scalar variable and gradient converge with the optimal order $k + 1$. We point out that our stabilization parameter is independent of the polynomial degree and the mesh size. This is different from some DG methods such as the interior penalty DG method which typically select stabilization parameter to depend on the mesh size.

2.8 Local Postprocessing

We first show that we can postprocess the total approximate flux $\mathbf{q}_h^T = \mathbf{q}_h + \mathbf{c}u_h$ and its numerical trace $\widehat{\mathbf{q}}_h^T = \widehat{\mathbf{q}}_h + \widehat{\mathbf{c}}u_h$ with an element-by-element procedure to

obtain an approximation of $\mathbf{q} + \mathbf{c}u$, denoted \mathbf{q}_h^{T*} that belongs to $H(\text{div}, \Omega)$ and also converges in an optimal fashion [2, 6, 14]. On each simplex $K \in \mathcal{T}_h$, we define the new total flux \mathbf{q}_h^{T*} as the only element of $(\mathcal{P}_k(K))^d + \mathbf{x} \mathcal{P}_k(K)$ satisfying, for $k \geq 0$,

$$\begin{aligned} \langle (\mathbf{q}_h^{T*} - \widehat{\mathbf{q}}_h^T) \cdot \mathbf{n}, \mu \rangle_F &= 0, \quad \forall \mu \in \mathcal{P}_k(F), \forall F \in \partial K, \\ (\mathbf{q}_h^{T*} - \mathbf{q}_h^T, \mathbf{v})_K &= 0, \quad \forall \mathbf{v} \in (\mathcal{P}_{k-1}(K))^d \quad \text{if } k \geq 1. \end{aligned} \quad (18)$$

It is clear that the function \mathbf{q}_h^{T*} belongs to $H(\text{div}, \Omega)$, thanks to the singlevaluedness of the normal component of the numerical trace $\widehat{\mathbf{q}}_h + \widehat{\mathbf{c}}u_h$.

Next, we consider postprocessing u_h , \mathbf{q}_h , and $\widehat{\mathbf{q}}_h$ to obtain the new approximate scalar variable u_h^* of u . Towards this end, we find $(u_h^*, \mathbf{q}_h^*, \lambda_h^*) \in \mathcal{P}_{k^*}(K) \times (\mathcal{P}_{k^*}(K))^d \times (\mathcal{P}_{k^*}(F))^{d+1}$ for $k^* = k + 1$ on the simplex $K \in \mathcal{T}_h$ such that

$$\begin{aligned} (\kappa^{-1} \nabla \mathbf{q}_h^*, \mathbf{v})_K - (u_h^*, \nabla \cdot \mathbf{v})_K + \langle \lambda_h^*, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial K} &= 0 \\ -(\mathbf{q}_h^* + \mathbf{c}u_h^*, \nabla w)_K + \langle (\widehat{\mathbf{q}}_h^* + \widehat{\mathbf{c}}u_h^*) \cdot \mathbf{n}, w \rangle_{\partial K} &= \langle \nabla \cdot \mathbf{q}_h^{T*}, w \rangle_K, \\ \langle (\widehat{\mathbf{q}}_h^* + \widehat{\mathbf{c}}u_h^*) \cdot \mathbf{n}, \mu \rangle_{\partial K} &= \langle \mathbf{q}_h^{T*} \cdot \mathbf{n}, \mu \rangle_{\partial K}, \\ (u_h^*, 1)_K &= (u_h, 1)_K, \end{aligned} \quad (19)$$

for all $(\mathbf{v}, w, \mu) \in (\mathcal{P}_{k^*}(K))^d \times \mathcal{P}_{k^*}(K) \times (\mathcal{P}_{k^*}(F))^{d+1}$, where

$$\widehat{\mathbf{q}}_h^* + \widehat{\mathbf{c}}u_h^* = \mathbf{q}_h^* + \mathbf{c}\lambda_h^* + \tau(u_h^* - \lambda_h^*)\mathbf{n}.$$

We note that this local postprocessing is nothing but the HDG discretization *at the element level* of the following convection-diffusion Neumann problem

$$\begin{aligned} \nabla \cdot (-\kappa \nabla u + \mathbf{c}u) &= \nabla \cdot \mathbf{q}_h^{T*}, \quad \text{in } K, \\ (-\kappa \nabla u + \mathbf{c}u) \cdot \mathbf{n} &= \mathbf{q}_h^{T*} \cdot \mathbf{n}, \quad \text{on } \partial K, \\ (u, 1)_K &= (u_h, 1)_K. \end{aligned} \quad (20)$$

Therefore, the new approximation u_h^* is even much less expensive to compute than the original approximation u_h . This is because the local problems (19) have very few degrees of freedom and also because they can be solved independently of each other.

Our postprocessing procedure relies on the optimal convergence of \mathbf{q}_h^{T*} and its divergence $\nabla \cdot \mathbf{q}_h^{T*}$, and on the superconvergence of the average of the approximate scalar variable u_h . In fact, these properties for the HDG method have been theoretically analyzed and confirmed by numerical experiments for the steady symmetric diffusion case in [4, 13]: both \mathbf{q}_h^{T*} and $\nabla \cdot \mathbf{q}_h^{T*}$ converge with order $k + 1$, while $(u_h, 1)_K$ superconverges with order $k + 2$. We may thus expect that the scalar variable u_h^* converges with order $k + 2$.

3 Extensions of the Basic Algorithm

In this section, we present several extensions of the basic algorithm described in the previous section.

3.1 Time-Dependent Convection-Diffusion Problems

We consider the time-dependent convection-diffusion model written as a system of first-order equations

$$\begin{aligned}
\mathbf{q} + \kappa \nabla u &= 0, & \text{in } \Omega \times (0, T], \\
\frac{\partial u}{\partial t} + \nabla \cdot (\mathbf{c}u + \mathbf{q}) &= f, & \text{in } \Omega \times (0, T], \\
u &= g_D, & \text{on } \Gamma_D \times (0, T], \\
(\mathbf{q} + \mathbf{c}u) \cdot \mathbf{n} &= g_N, & \text{on } \Gamma_N \times (0, T], \\
u &= u_0, & \text{in } \Omega \text{ for } t = 0.
\end{aligned} \tag{21}$$

The HDG method of lines for the above problem seeks an approximation $(\mathbf{q}_h, u_h) \in V_h \times W_h$ such that for all $K \in \mathcal{T}_h$,

$$\begin{aligned}
(\kappa^{-1} \mathbf{q}_h, \mathbf{v})_K - (u_h, \nabla \cdot \mathbf{v})_K + \langle \widehat{u}_h, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial K} &= 0, \\
\left(\frac{\partial u_h}{\partial t}, w \right)_K - (\mathbf{c}u_h + \mathbf{q}_h, \nabla w)_K + \langle (\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h) \cdot \mathbf{n}, w \rangle_{\partial K} &= (f, w)_K,
\end{aligned} \tag{22}$$

for all $(\mathbf{v}, w) \in (\mathcal{P}_k(K))^d \times \mathcal{P}_k(K)$ and for all $t \in (0, T]$. Here, the numerical traces $\widehat{\mathbf{c}}u_h + \widehat{\mathbf{q}}_h$ and \widehat{u}_h are approximations to $\mathbf{c}u - \kappa \nabla u$ and u over ∂K , respectively.

The above HDG formulation (22) can then be discretized in time using an appropriate time-stepping scheme. Here we consider backward difference formulae (BDF) for the discretization of the time derivative. For instance, using the Backward-Euler scheme at time-level t^n with timestep Δt^n the HDG method then seeks an approximation $(\mathbf{q}_h^n, u_h^n, \lambda_h^n) \in V_h \times W_h \times M_h(0)$ such that

$$\begin{aligned}
(\kappa^{-1} \mathbf{q}_h^n, \mathbf{v})_{\mathcal{T}_h} - (u_h^n, \nabla \cdot \mathbf{v})_{\mathcal{T}_h} + \langle \lambda_h^n, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial \mathcal{T}_h} &= -\langle g_D, \mathbf{v} \cdot \mathbf{n} \rangle_{\Gamma_D}, \\
\frac{1}{\Delta t^n} (u_h^n, w)_{\mathcal{T}_h} - (\mathbf{c}u_h^n + \mathbf{q}_h^n, \nabla w)_{\mathcal{T}_h} &+ \langle (\widehat{\mathbf{c}}u_h^n + \widehat{\mathbf{q}}_h^n) \cdot \mathbf{n}, w \rangle_{\partial \mathcal{T}_h} = (f, w)_{\mathcal{T}_h} + \frac{1}{\Delta t^n} (u_h^{k-1}, w)_{\mathcal{T}_h}, \\
\langle [(\widehat{\mathbf{c}}u_h^n + \widehat{\mathbf{q}}_h^n) \cdot \mathbf{n}], \mu \rangle_{\mathcal{E}_h} &= \langle g_N, \mu \rangle_{\Gamma_N},
\end{aligned} \tag{23}$$

for all $(\mathbf{v}, w, \mu) \in V_h \times W_h \times M_h(0)$, where, as we did for the steady-state case, we choose $\widehat{\mathbf{c}}u_h^n + \widehat{\mathbf{q}}_h^n$ of the form

$$\widehat{\mathbf{c}}u_h^n + \widehat{\mathbf{q}}_h^n = \mathbf{c} \widehat{u}_h^n + \mathbf{q}_h^n + \tau(u_h^n - \widehat{u}_h^n) \mathbf{n}, \quad \text{on } \partial K.$$

This discrete system has a similar form as the system (7) for the steady-state case. Hence, we can apply exactly the same solution procedure described earlier for the steady-state case to the time-dependent case at every time step.

Of course, a similar procedure can be applied to treat any higher-order BDF method such as the widely used second-order and third-order BDF schemes. The HDG method can also work with other implicit time-stepping methods such as the fully implicit Runge–Kutta methods and DG methods in time.

The post-processing method described for the steady state convection diffusion problem can also be applied in the time-dependent case with identical results. That is, both \mathbf{q}_h^T and $\nabla \cdot \mathbf{q}_h^T$ converge spatially with order $p + 1$, while $(u_h, 1)_K$ superconverges in space with order $p + 2$. This means that it is then possible to reconstruct, at any desired time level, a new scalar variable, u_h^* , which superconverges with order $p + 2$ (see [20] for additional details).

3.2 Nonlinear Convection-Diffusion Problems

Here, we describe the HDG method for steady-state nonlinear convection-diffusion equations presented in [21]. Consider a nonlinear convection-diffusion equation of the form

$$\begin{aligned} -\nabla \cdot (\kappa \nabla u) + \nabla \cdot \mathbf{F}(u) &= f, \quad \text{in } \Omega, \\ u &= g_D, \quad \text{on } \partial\Omega. \end{aligned} \quad (24)$$

We rewrite the above equation as a first order system of equations

$$\begin{aligned} \mathbf{q} + \kappa \nabla u &= 0, \quad \text{in } \Omega, \\ \nabla \cdot (\mathbf{q} + \mathbf{F}(u)) &= f, \quad \text{in } \Omega, \\ u &= g_D, \quad \text{on } \partial\Omega. \end{aligned} \quad (25)$$

Here, $\mathbf{F} \in (L^\infty(\Omega))^d$ are vector-valued nonlinear functions of the scalar variable u .

Multiplying the first two equations of (25) by test functions, integrating by parts, and enforcing the continuity of the normal component of the total numerical flux, we obtain the following problem: find an approximation $(\mathbf{q}_h, u_h, \widehat{u}_h) \in \mathbf{V}_h \times W_h \times M_h(g_D)$ such that

$$\begin{aligned} (\kappa^{-1} \mathbf{q}_h, \mathbf{v})_{\mathcal{T}_h} - (u_h, \nabla \cdot \mathbf{v})_{\mathcal{T}_h} + \langle \widehat{u}_h, \mathbf{v} \cdot \mathbf{n} \rangle_{\partial\mathcal{T}_h} &= 0, \\ -(\mathbf{q}_h + \mathbf{F}(u_h), \nabla w)_{\mathcal{T}_h} + \left\langle (\widehat{\mathbf{q}}_h + \widehat{\mathbf{F}}_h) \cdot \mathbf{n}, w \right\rangle_{\partial\mathcal{T}_h} &= (f, w)_{\mathcal{T}_h}, \\ \left\langle (\widehat{\mathbf{q}}_h + \widehat{\mathbf{F}}_h) \cdot \mathbf{n}, \mu \right\rangle_{\partial\mathcal{T}_h} &= 0, \end{aligned} \quad (26)$$

for all $(\mathbf{v}, w, \mu) \in \mathbf{V}_h \times W_h \times M_h(0)$, where

$$\widehat{\mathbf{q}}_h + \widehat{\mathbf{F}}_h = \mathbf{q}_h + \mathbf{F}(\widehat{u}_h) + \tau(u_h, \widehat{u}_h)(u_h - \widehat{u}_h)\mathbf{n}, \quad \text{on } \mathcal{E}_h. \quad (27)$$

This completes the definition of the general form of the HDG method. This nonlinear system of equations is solved by the Newton–Raphson method as described in [21]. Here we observe that, at each Newton iteration, we recover the HDG structure of the linear problem (7), and thus solve for the degrees of freedom of \widehat{u}_h only.

The choice of the numerical flux $\widehat{q}_h + \widehat{F}_h$ is an extension of the expression for the numerical flux used for the linear case. The main difference is that, due to the nonlinearity of the convection, the *stabilization function* $\tau(\cdot, \cdot) : \partial\mathcal{T}_h \rightarrow \mathbb{R}$ can now be a nonlinear function of u_h and \widehat{u}_h . This implies that the last equation (26) *cannot* force the normal component of the total flux $\widehat{q}_h + \widehat{F}_h$ to be single valued on all interior faces $e \in \mathcal{E}_h^o$; it only forces its L^2 -projection into $M_h(0)$ to be single valued. This is enough to guarantee the local conservativity of the method, as we can see from the second term of the left-hand side of the second equation (26).

Suitable expressions for the stabilization function and the associated entropy inequality as well as the extension to nonlinear time dependent problems and the postprocessing procedure are described in [21].

3.3 Stokes Flows

We describe here a hybridizable discontinuous Galerkin (HDG) method for the Stokes system [22]

$$\begin{aligned} -\nu\Delta\mathbf{u} + \nabla p &= \mathbf{f}, & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} &= 0, & \text{in } \Omega, \\ \mathbf{u} &= \mathbf{g}, & \text{on } \partial\Omega. \end{aligned} \quad (28)$$

We rewrite the above equation as the following first order system of equations

$$\begin{aligned} \mathbf{L} - \nabla\mathbf{u} &= 0, & \text{in } \Omega \\ -\nu\nabla \cdot \mathbf{L} + \nabla p &= \mathbf{f}, & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} &= 0, & \text{in } \Omega, \\ \mathbf{u} &= \mathbf{g} & \text{on } \partial\Omega. \end{aligned} \quad (29)$$

As usual we assume that \mathbf{g} satisfies the compatibility condition $\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} = 0$.

We first introduce discontinuous finite element approximation spaces for the gradient, velocity, and pressure as

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

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

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

We also set

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

where \mathbf{P} denotes the L^2 -projection into the space $\{\boldsymbol{\mu}|_{\partial\Omega} \mid \boldsymbol{\mu} \in \mathbf{M}_h\}$. 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. For a function q in $L^2(\partial\mathcal{T}_h)$, the mean of q on the element boundary ∂K of an element K is defined as

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

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

We next define various inner products for our finite element spaces as

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

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

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

for $r, q \in L^2(\mathcal{E}_h)$, $\mathbf{w}, \mathbf{v} \in (L^2(\mathcal{E}_h))^d$, and $\mathbf{H}, \mathbf{G} \in (L^2(\mathcal{E}_h))^{d \times d}$. Recall the standard notation $(\mathbf{H}, \mathbf{G})_D = \int_D \text{tr}(\mathbf{H}^T \mathbf{G})$, where tr is the trace operator.

The HDG method then seeks an approximation $(\mathbf{L}_h, \mathbf{u}_h, p_h, \widehat{\mathbf{u}}_h, \bar{p}_h) \in \mathbf{G}_h \times \mathbf{V}_h \times P_h \times \mathbf{M}_h(\mathbf{g}) \times \overline{\Psi}_h$ such that

$$\begin{aligned} (\mathbf{L}_h, \mathbf{G})_{\mathcal{T}_h} + (\mathbf{u}_h, \nabla \cdot \mathbf{G})_{\mathcal{T}_h} - \langle \widehat{\mathbf{u}}_h, \mathbf{G} \cdot \mathbf{n} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (\nu \mathbf{L}_h - p_h \mathbf{I}, \nabla \mathbf{v})_{\mathcal{T}_h} + \left\langle (-\nu \widehat{\mathbf{L}}_h + \widehat{p}_h \mathbf{I}) \cdot \mathbf{n}, \mathbf{v} \right\rangle_{\partial\mathcal{T}_h} &= (\mathbf{f}, \mathbf{v})_{\mathcal{T}_h}, \\ -(\mathbf{u}_h, \nabla q)_{\mathcal{T}_h} + \langle \widehat{\mathbf{u}}_h \cdot \mathbf{n}, q - \bar{q} \rangle_{\partial\mathcal{T}_h} &= 0, \\ \bar{p}_h - \bar{\rho}_h &= 0, \\ \left\langle (-\nu \widehat{\mathbf{L}}_h + \widehat{p}_h \mathbf{I}) \cdot \mathbf{n}, \boldsymbol{\mu} \right\rangle_{\partial\mathcal{T}_h} &= 0, \\ \langle \widehat{\mathbf{u}}_h \cdot \mathbf{n}, \bar{\psi} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (p_h, 1)_{\mathcal{T}_h} &= 0, \end{aligned} \tag{30}$$

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

$$-\nu \widehat{\mathbf{L}}_h + \widehat{p}_h \mathbf{I} = -\nu \mathbf{L}_h + p_h \mathbf{I} + \mathbf{S}(\mathbf{u}_h - \widehat{\mathbf{u}}_h) \otimes \mathbf{n}. \tag{31}$$

Here \mathbf{S} is the second-order tensor consisting of stabilization parameters and \mathbf{I} is the second-order identity tensor. Note also that the Dirichlet boundary condition has been enforced by requiring that $\widehat{\mathbf{u}}_h \in \mathbf{M}_h(\mathbf{g})$.

The first four equations of (30) define the local solver which can be used to eliminate all the variables L_h , \mathbf{u}_h , and p_h by inserting them into the last three equations of (30), thereby obtaining a linear system in terms of $(\widehat{\mathbf{u}}_h, \bar{\rho}_h)$ only. Since $\widehat{\mathbf{u}}_h$ is defined on the element faces and $\bar{\rho}_h$ has one degree of freedom per element, the HDG method reduces significantly the number of the globally coupled unknowns. In practice, we implement the HDG method by using the augmented Lagrangian approach [16]; see [22] for a detailed discussion.

Finally, we use the element-by-element postprocessing proposed in [11] obtain a new approximate velocity which is exactly divergence-free, $\mathbf{H}(\text{div})$ -conforming, and converges with the order $k + 2$. In the three dimensional case, we define the postprocessed approximate velocity \mathbf{u}_h^* on the tetrahedron $K \in \mathcal{T}_h$ as the element of $(\mathcal{P}_{k+1}(K))^d$ such that

$$\langle (\mathbf{u}_h^* - \widehat{\mathbf{u}}_h) \cdot \mathbf{n}, \mu \rangle_F = 0 \quad \forall \mu \in \mathcal{P}_k(F), \quad (32a)$$

$$\langle (\mathbf{n} \times \nabla)(\mathbf{u}_h^* \cdot \mathbf{n}) - \mathbf{n} \times (\{\mathbf{L}_h^t\} \mathbf{n}), (\mathbf{n} \times \nabla)\mu \rangle_F = 0 \quad \forall \mu \in \mathcal{P}_{k+1}(F)^\perp, \quad (32b)$$

for all faces F of K , and such that

$$(\mathbf{u}_h^* - \mathbf{u}_h, \nabla w)_K = 0 \quad \forall w \in \mathcal{P}_k(K), \quad (32c)$$

$$(\nabla \times \mathbf{u}_h^* - \mathbf{w}_h, (\nabla \times \mathbf{v}) \mathbf{B}_K)_K = 0 \quad \forall \mathbf{v} \in \mathcal{S}_k(K). \quad (32d)$$

Here

$$\mathcal{P}_{k+1}(F)^\perp := \{\mu \in \mathcal{P}_{k+1}(F) : \langle \mu, \widetilde{\mu} \rangle_F = 0, \quad \forall \widetilde{\mu} \in \mathcal{P}_k(F)\},$$

and

$$\mathbf{w}_h := (\mathbf{L}_{32}^h - \mathbf{L}_{23}^h, \mathbf{L}_{13}^h - \mathbf{L}_{31}^h, \mathbf{L}_{21}^h - \mathbf{L}_{12}^h)$$

is the approximation to the vorticity. Furthermore, \mathbf{B}_K is the so-called *symmetric bubble matrix* introduced in [8], 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 . Finally, $\mathcal{S}_k(K) := \{\mathbf{p} \in \mathbf{N}_k : (\mathbf{p}, \nabla \phi)_K = 0 \text{ for all } \phi \in \mathcal{P}_{k+1}(K)\}$, where $\mathbf{N}_k = \mathcal{P}_{k-1}(K) \oplus \mathcal{S}_k$ and \mathcal{S}_ℓ is the space of vector-valued homogeneous polynomials \mathbf{v} of degree ℓ such that $\mathbf{v} \cdot \mathbf{x} = 0$; see [18, 19].

In the two dimensional case, the postprocessing is defined by the above equations if (32d) is replaced by

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

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

We refer the reader to [11] for a proof of the fact that \mathbf{u}_h^* is a divergence-free velocity in $\mathbf{H}(\operatorname{div}, \Omega)$ and converges with the order $k + 2$ in the L^2 -norm.

3.4 Incompressible Navier–Stokes Equations

Let us extend the HDG method described above to the steady incompressible Navier–Stokes equations written in conservative form

$$\begin{aligned} \mathbf{L} - \nabla \mathbf{u} &= 0, & \text{in } \Omega \\ -\nu \nabla \cdot \mathbf{L} + \nabla p + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) &= \mathbf{f}, & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} &= 0, & \text{in } \Omega, \\ \mathbf{u} &= \mathbf{g}, & \text{on } \partial\Omega. \end{aligned} \tag{33}$$

The Navier–Stokes system differs from the Stokes one due to the presence of the nonlinear convective term $\nabla \cdot (\mathbf{u} \otimes \mathbf{u})$.

The HDG method for the above system seeks an approximation $(\mathbf{L}_h, \mathbf{u}_h, p_h, \widehat{\mathbf{u}}_h, \bar{\rho}_h) \in \mathbf{G}_h \times \mathbf{V}_h \times P_h \times \mathbf{M}_h(\mathbf{g}) \times \bar{\Psi}_h$ such that

$$\begin{aligned} (\mathbf{L}_h, \mathbf{G})_{\mathcal{T}_h} + (\mathbf{u}_h, \nabla \cdot \mathbf{G})_{\mathcal{T}_h} - \langle \widehat{\mathbf{u}}_h, \mathbf{G}\mathbf{n} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (v\mathbf{L}_h - p_h\mathbf{I} - \mathbf{u}_h \otimes \mathbf{u}_h, \nabla \mathbf{v})_{\mathcal{T}_h} \\ + \left\langle (-v\widehat{\mathbf{L}}_h + \widehat{p}_h\mathbf{I} + \widehat{\mathbf{u}}_h \otimes \widehat{\mathbf{u}}_h)\mathbf{n}, \mathbf{v} \right\rangle_{\partial\mathcal{T}_h} &= (\mathbf{f}, \mathbf{v})_{\mathcal{T}_h}, \\ -(\mathbf{u}_h, \nabla q)_{\mathcal{T}_h} + \langle \widehat{\mathbf{u}}_h \cdot \mathbf{n}, q - \bar{q} \rangle_{\partial\mathcal{T}_h} &= 0, \\ \bar{p}_h - \bar{\rho}_h &= 0, \\ \left\langle (-v\widehat{\mathbf{L}}_h + \widehat{p}_h\mathbf{I} + \widehat{\mathbf{u}}_h \otimes \widehat{\mathbf{u}}_h)\mathbf{n}, \boldsymbol{\mu} \right\rangle_{\partial\mathcal{T}_h} &= 0, \\ \langle \widehat{\mathbf{u}}_h \cdot \mathbf{n}, \bar{\psi} \rangle_{\partial\mathcal{T}_h} &= 0, \\ (p_h, 1)_{\mathcal{T}_h} &= 0, \end{aligned} \tag{34}$$

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

$$\left(-v\widehat{\mathbf{L}}_h + \widehat{p}_h\mathbf{I} \right) \mathbf{n} = (-v\mathbf{L}_h + p_h\mathbf{I}) \mathbf{n} + s_h(\mathbf{u}_h, \widehat{\mathbf{u}}_h). \tag{35}$$

Here $s_h(\mathbf{u}_h, \widehat{\mathbf{u}}_h)$ is the *stabilization vector-valued function* the choice of which is crucial since it does have an important effect on both the stability and accuracy of the method. We consider an extension of the expression for $s_h(\mathbf{u}_h, \widehat{\mathbf{u}}_h)$ proposed in [11, 22] for the Stokes system as follows

$$s_h(\mathbf{u}_h, \widehat{\mathbf{u}}_h) = \mathbf{S}(\mathbf{u}_h, \widehat{\mathbf{u}}_h)(\mathbf{u}_h - \widehat{\mathbf{u}}_h), \tag{36}$$

where $\mathbf{S}(\mathbf{u}_h, \widehat{\mathbf{u}}_h)$ is the *stabilization tensor* which may depend on \mathbf{u}_h and $\widehat{\mathbf{u}}_h$.

Substituting (35) into (34) we obtain that $(\mathbf{L}_h, \mathbf{u}_h, p_h, \widehat{\mathbf{u}}_h, \bar{\rho}_h) \in \mathbf{G}_h \times \mathbf{V}_h \times P_h \times \mathbf{M}_h(\mathbf{g}) \times \bar{\Psi}_h$ is the solution of

$$\begin{aligned}
(\mathbf{L}_h, \mathbf{G})_{\mathcal{T}_h} + (\mathbf{u}_h, \nabla \cdot \mathbf{G})_{\mathcal{T}_h} - \langle \widehat{\mathbf{u}}_h, \mathbf{G}\mathbf{n} \rangle_{\partial\mathcal{T}_h} &= 0, \\
(\nabla \cdot (-\nu\mathbf{L}_h + p_h\mathbf{I}), \mathbf{v})_{\mathcal{T}_h} - (\mathbf{u}_h \otimes \mathbf{u}_h, \nabla\mathbf{v})_{\mathcal{T}_h} \\
+ \langle (\widehat{\mathbf{u}}_h \otimes \widehat{\mathbf{u}}_h)\mathbf{n} + s_h(\mathbf{u}_h, \widehat{\mathbf{u}}_h), \mathbf{v} \rangle_{\partial\mathcal{T}_h} &= (\mathbf{f}, \mathbf{v})_{\mathcal{T}_h}, \\
-(\mathbf{u}_h, \nabla q)_{\mathcal{T}_h} + \langle \widehat{\mathbf{u}}_h \cdot \mathbf{n}, q - \bar{q} \rangle_{\partial\mathcal{T}_h} &= 0, \\
\bar{p}_h - \bar{\rho}_h &= 0, \\
\langle (-\nu\mathbf{L}_h + p_h\mathbf{I} + \widehat{\mathbf{u}}_h \otimes \widehat{\mathbf{u}}_h)\mathbf{n} + s_h(\mathbf{u}_h, \widehat{\mathbf{u}}_h), \boldsymbol{\mu} \rangle_{\partial\mathcal{T}_h} &= 0, \\
\langle \widehat{\mathbf{u}}_h \cdot \mathbf{n}, \bar{\psi} \rangle_{\partial\mathcal{T}_h} &= 0, \\
(p_h, 1)_{\mathcal{T}_h} &= 0,
\end{aligned} \tag{37}$$

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

The above nonlinear system of equations is solved by the Newton–Raphson method: Given the m th current iterate $(\mathbf{L}_h^m, \mathbf{u}_h^m, p_h^m, \widehat{\mathbf{u}}_h^m, \bar{\rho}_h^m)$, we find an increment $(\delta\mathbf{L}_h^m, \delta\mathbf{u}_h^m, \delta p_h^m, \delta\widehat{\mathbf{u}}_h^m, \delta\bar{\rho}_h^m) \in \mathbf{G}_h \times \mathbf{V}_h \times P_h \times \mathbf{M}_h(\mathbf{0}) \times \bar{\Psi}_h$ such that

$$\begin{aligned}
(\delta\mathbf{L}_h^m, \mathbf{G})_{\mathcal{T}_h} + (\delta\mathbf{u}_h^m, \nabla \cdot \mathbf{G})_{\mathcal{T}_h} - \langle \delta\widehat{\mathbf{u}}_h^m, \mathbf{G}\mathbf{n} \rangle_{\partial\mathcal{T}_h} &= r_1(\mathbf{G}), \\
(\nabla \cdot (-\nu\delta\mathbf{L}_h^m + \delta p_h^m\mathbf{I}), \mathbf{v})_{\mathcal{T}_h} - (\delta\mathbf{u}_h^m \otimes \mathbf{u}_h^m + \mathbf{u}_h^m \otimes \delta\mathbf{u}_h^m, \nabla\mathbf{v})_{\mathcal{T}_h} \\
+ \langle (\delta\widehat{\mathbf{u}}_h^m \otimes \widehat{\mathbf{u}}_h^m + \widehat{\mathbf{u}}_h^m \otimes \delta\widehat{\mathbf{u}}_h^m)\mathbf{n} + \partial_1 s_h(\mathbf{u}_h^m, \widehat{\mathbf{u}}_h^m)\delta\mathbf{u}_h^m + \partial_2 s_h(\mathbf{u}_h^m, \widehat{\mathbf{u}}_h^m)\delta\widehat{\mathbf{u}}_h^m, \mathbf{v} \rangle_{\partial\mathcal{T}_h} &= r_2(\mathbf{v}), \\
-(\delta\mathbf{u}_h^m, \nabla q)_{\mathcal{T}_h} + \langle \delta\widehat{\mathbf{u}}_h^m \cdot \mathbf{n}, q - \bar{q} \rangle_{\partial\mathcal{T}_h} &= r_3(q), \\
\delta\bar{p}_h^m - \delta\bar{\rho}_h^m &= r_4, \\
\langle (-\nu\delta\mathbf{L}_h^m + \delta p_h^m\mathbf{I} + \delta\widehat{\mathbf{u}}_h^m \otimes \widehat{\mathbf{u}}_h^m + \widehat{\mathbf{u}}_h^m \otimes \delta\widehat{\mathbf{u}}_h^m)\mathbf{n} \rangle_{\partial\mathcal{T}_h} \\
+ \langle \partial_1 s_h(\mathbf{u}_h^m, \widehat{\mathbf{u}}_h^m)\delta\mathbf{u}_h^m + \partial_2 s_h(\mathbf{u}_h^m, \widehat{\mathbf{u}}_h^m)\delta\widehat{\mathbf{u}}_h^m, \boldsymbol{\mu} \rangle_{\partial\mathcal{T}_h} &= r_5(\boldsymbol{\mu}), \\
\langle \delta\widehat{\mathbf{u}}_h^m \cdot \mathbf{n}, \bar{\psi} \rangle_{\partial\mathcal{T}_h} &= r_6(\bar{\psi}), \\
(\delta p_h^m, 1)_{\mathcal{T}_h} &= r_7,
\end{aligned} \tag{38}$$

for all $(\mathbf{G}, \mathbf{v}, q, \boldsymbol{\mu}, \bar{\psi}) \in \mathbf{G}_h \times \mathbf{V}_h \times P_h \times \mathbf{M}_h(\mathbf{0}) \times \bar{\Psi}_h$. Note here that the right-hand side residuals are evaluated from (37) at the current iterate.

We observe that the above system (38) has a similar structure as the HDG system (30) for the Stokes flow except that there are some additional terms due to the convective nonlinearity. Therefore, it can be solved in a similar manner by means of the hybridization technique. This leads to a linear system of algebraic equations involving the degrees of freedom of $(\delta\widehat{\mathbf{u}}_h^m, \delta\bar{\rho}_h^m)$ only. Alternatively, we may apply the augmented Lagrangian approach to the nonlinear system (37) and then use the hybridization technique to obtain a system in terms of $\delta\widehat{\mathbf{u}}_h^m$ only [23].

Although our discussion has focused primarily on the steady-state case, the same HDG method can be applied to the time-dependent problem with using an implicit time-stepping method; see Sect. 3.1 for further details. Finally, we emphasize that the postprocessing procedure described for Stokes flow can be used for both the steady and unsteady Navier–Stokes problems.

4 Numerical Results

In this section, we present numerical results for a benchmark problem in fluid dynamics. We would like to refer the readers to the previous work [4, 6, 7, 11, 20–23, 27] for many other examples which demonstrate the performance and accuracy of the HDG methods described in this paper.

The Taylor vortex problem is a well-known example of the unsteady incompressible Navier–Stokes equations. The problem has an exact solution of the form

$$\begin{aligned} u_x &= -\cos(\pi x) \sin(\pi y) \exp\left(\frac{-2\pi^2 t}{Re}\right), \\ u_y &= \sin(\pi x) \cos(\pi y) \exp\left(\frac{-2\pi^2 t}{Re}\right), \\ p &= -\frac{1}{4}(\cos(2\pi x) + \cos(2\pi y)) \exp\left(\frac{-4\pi^2 t}{Re}\right), \end{aligned}$$

where $Re = 1/\nu$ is the Reynolds number. We consider the above problem on $\Omega = (0, 1)^2$ with Reynolds number $Re = 20$ and final time $T = 1$. We take the Dirichlet boundary condition for the velocity as the restriction of the exact solution to the domain boundary and the initial condition as an instantiation of the exact solution at $t = 0$.

In our experiments, we consider triangular meshes that are obtained by splitting a regular $n \times n$ Cartesian grid into a total of $2n^2$ triangles, giving uniform element sizes of $h = 1/n$. 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. We use the third-order backward difference formula (BDF3) for the temporal discretization. The stabilization tensor S is chosen as

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

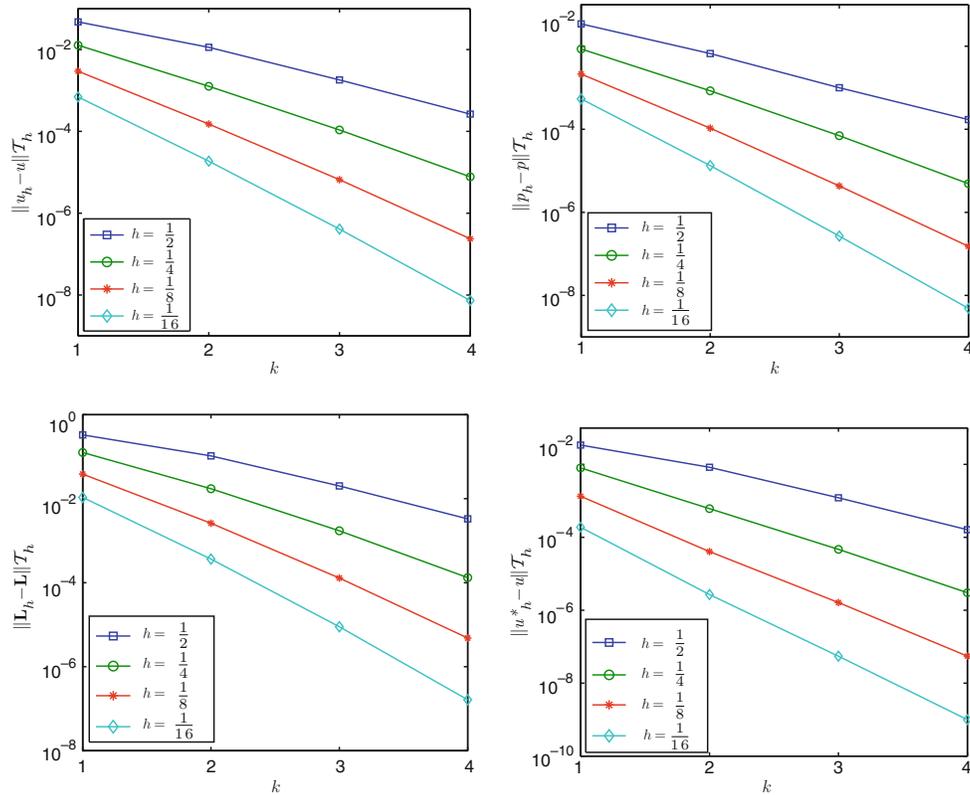
where τ is equal to 1 on \mathcal{E}_h .

We first look at the convergence and accuracy in terms of both k and h refinements. For this purpose, we select a small timestep of $\Delta t = 0.005$, so that the spatial error is dominant and the temporal error is negligible. We present in Table 1 the history of convergence of the HDG method at the final time $t = 1$. We observe that the approximate velocity, pressure, and velocity gradient converge with the optimal order $k + 1$ for $k = 1, 2, 3$. The fact that the HDG method yields optimal convergence for both the approximate pressure and velocity gradient is a very important advantage since many other DG methods provide suboptimal convergence of order k for the approximate pressure and velocity gradient. Moreover, we observe that all the approximate variables converge *exponentially* with the polynomial degree k as depicted in Fig. 1. We emphasize that these results are obtained with τ being set to 1 and thus independent of both k and h .

Equally important is the fact that the postprocessed velocity \mathbf{u}_h^* converges with the order $k + 2$, which is one order higher than the original approximate velocity

Table 1 History of convergence of the HDG method for the Taylor vortex problem with $Re = 20$

Degree k	Mesh 1/h	$\ \mathbf{u} - \mathbf{u}_h\ _{\mathcal{T}_h}$		$\ p - p_h\ _{\mathcal{T}_h}$		$\ \mathbf{L} - \mathbf{L}_h\ _{\mathcal{T}_h}$		$\ \mathbf{u} - \mathbf{u}_h^*\ _{\mathcal{T}_h}$	
		Error	Order	Error	Order	Error	Order	Error	Order
1	2	4.73e-2	–	3.44e-2	–	3.29e-1	–	3.40e-2	–
	4	1.27e-2	1.89	8.59e-3	2.00	1.26e-1	1.39	8.04e-3	2.08
	8	2.94e-3	2.11	2.14e-3	2.01	3.85e-2	1.71	1.34e-3	2.59
	16	6.95e-4	2.08	5.38e-4	1.99	1.07e-2	1.84	1.89e-4	2.82
	32	1.70e-4	2.03	1.36e-4	1.99	2.85e-3	1.91	2.50e-5	2.92
2	2	1.14e-2	–	6.67e-3	–	1.04e-1	–	8.35e-3	–
	4	1.26e-3	3.17	8.43e-4	2.98	1.72e-2	2.60	6.12e-4	3.77
	8	1.51e-4	3.06	1.07e-4	2.98	2.60e-3	2.73	4.07e-5	3.91
	16	1.87e-5	3.01	1.33e-5	3.00	3.64e-4	2.84	2.70e-6	3.91
	32	2.33e-6	3.00	1.67e-6	3.00	4.85e-5	2.91	1.76e-7	3.94
3	2	1.81e-3	–	1.00e-3	–	2.01e-2	–	1.22e-3	–
	4	1.08e-4	4.06	7.00e-5	3.84	1.72e-3	3.54	4.67e-5	4.70
	8	6.59e-6	4.04	4.33e-6	4.01	1.29e-4	3.74	1.63e-6	4.84
	16	4.08e-7	4.01	2.68e-7	4.01	8.92e-6	3.85	5.48e-8	4.89
	32	2.55e-8	4.00	1.67e-8	4.00	5.88e-7	3.92	1.82e-9	4.91

**Fig. 1** The L^2 error in log scale as a function of h and k for \mathbf{u}_h (top left), p_h (top right), \mathbf{L}_h (bottom left), and \mathbf{u}_h^* (bottom right)

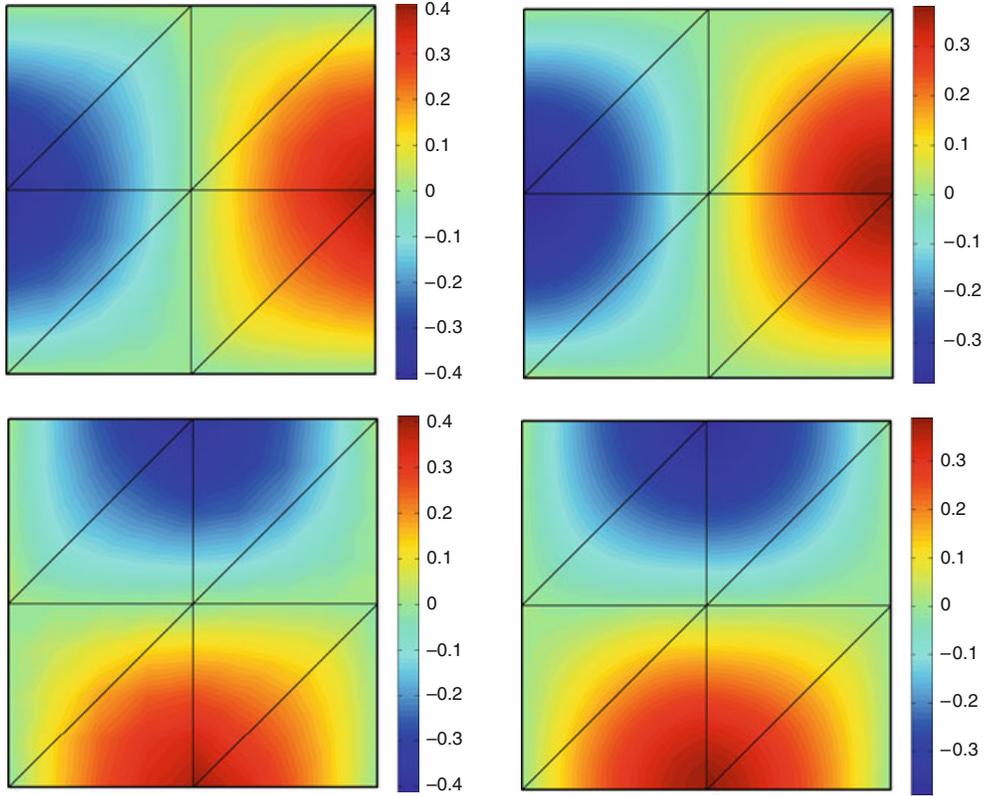


Fig. 2 The approximate velocity \mathbf{u}_h (left) and the postprocessed velocity \mathbf{u}_h^* (right) for $k = 2$ on the grid $h = 1/2$, with horizontal velocity at the *top* and vertical velocity at the *bottom*

\mathbf{u}_h . Furthermore, we emphasize that \mathbf{u}_h^* is an exactly divergence-free and $\mathbf{H}(\text{div})$ -conforming velocity field. To visualize the effect of the local postprocessing, we show in Fig. 2 the plots of the approximate velocity and the postprocessed velocity for $k = 2$ on the grid $h = 1/2$. We observe that the local postprocessing does provide a significant improvement in the approximation of the velocity field, since \mathbf{u}_h^* is clearly superior to \mathbf{u}_h .

Moreover, since the local postprocessing is performed at the element level and only at the timestep where higher accuracy is desired, it adds very little to the overall computational cost. As a result, with the HDG method, the $(k + 2)$ -convergent velocity, $(k + 1)$ -convergent pressure, and $(k + 1)$ -convergent velocity gradient can be computed at the cost of a DG approximation using polynomials of degree k .

5 Conclusions

We present an overview of recent developments of HDG methods for numerically solving partial differential equations in fluid mechanics. The main philosophy of the HDG methodology includes the following steps:

- Identify the globally coupled unknowns as the numerical traces of the field variables associated with the essential boundary condition.
- Enforce explicitly the continuity of the normal component of the numerical fluxes associated with the Neumann boundary condition. This is called the conservativity condition.
- Define the local solver by applying the HDG method to the governing equations at the element level.
- Substitute all the volumetric unknowns from the local solver into the conservativity condition to obtain a final system in terms of the numerical traces only.
- Apply the local postprocessing to obtain an improved approximation of the field variables.

The above guidelines are very general and applicable beyond problems considered in this paper. Indeed, based on this general framework we have successfully developed HDG methods for the compressible Euler and Navier–Stokes equations [25]. Inspired by the simplicity and generality of this new DG methodology, our current research effort focuses on devising HDG methods for multi-physics applications.

References

1. D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39(5):1749–1779, 2001
2. P. Bastian and B. Rivière. Superconvergence and $H(\text{div})$ projection for discontinuous Galerkin methods. *Int. J. Numer. Methods Fluids*, 42:1043–1057, 2003
3. P. Castillo, B. Cockburn, I. Perugia, and D. Schötzau. An a priori error analysis of the local discontinuous Galerkin method for elliptic problems. *SIAM J. Numer. Anal.*, 38(5):1676–1706, 2001
4. B. Cockburn, B. Dong, and J. Guzmán. A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems. *Math. Comp.*, 77:1887–1916, 2008
5. B. Cockburn and J. Gopalakrishnan. A characterization of hybridized mixed methods for second order elliptic problems. *SIAM J. Numer. Anal.*, 42(1):283–301, 2004
6. B. Cockburn, B. Dong, J. Guzmán, M. Restelli, and R. Sacco. An hybridizable discontinuous Galerkin method for steady-state convection-diffusion-reaction problems. *SIAM J. Sci. Comput.*, 31(5):3827–3846, 2009
7. B. Cockburn and J. Gopalakrishnan. The derivation of hybridizable discontinuous Galerkin methods for Stokes flow. *SIAM J. Numer. Anal.*, 47:1092–1125, 2009
8. B. Cockburn, J. Gopalakrishnan, and J. Guzmán. A new elasticity element made for enforcing weak stress symmetry. *Math. Comp.*, 79:1331–1349, 2009
9. B. Cockburn, J. Gopalakrishnan, and R. Lazarov. Unified hybridization of discontinuous Galerkin, mixed and continuous Galerkin methods for second-order elliptic problems. *SIAM J. Numer. Anal.*, 47:1319–1365, 2009
10. B. Cockburn, J. Gopalakrishnan, and F.-J. Sayas. A projection-based error analysis of HDG methods. *Math. Comp.*, 79:1351–1367, 2010
11. B. Cockburn, J. Gopalakrishnan, N. C. Nguyen, J. Peraire, and F.-J. Sayas. Analysis of an HDG method for Stokes flow. *Math. Comp.* DOI: 10.1090/S0025-5718-2010-02410-X

12. B. Cockburn, J. Guzmán, S.-C. Soon, and H. K. Stolarski. An analysis of the embedded discontinuous Galerkin method for second-order elliptic problems. *SIAM J. Numer. Anal.*, 47(4):2686–2707, 2009
13. B. Cockburn, J. Guzmán, and H. Wang. Superconvergent discontinuous Galerkin methods for second-order elliptic problems. *Math. Comp.*, 78:1–24, 2009
14. B. Cockburn, G. Kanschat, and D. Schötzau. A locally conservative LDG method for the incompressible Navier–Stokes equations. *Math. Comp.*, 74:1067–1095, 2005
15. B. Cockburn and C. W. Shu. The local discontinuous Galerkin method for convection-diffusion systems. *SIAM J. Numer. Anal.*, 35:2440–2463, 1998
16. M. Fortin, and R. Glowinski. *Augmented Lagrangian methods*, volume 15 of *Studies in Mathematics and its Applications*. 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
17. S. Güzey, B. Cockburn, and H. Stolarski. The embedded discontinuous Galerkin methods: Application to linear shells problems, *Int. J. Numer. Methods Eng.*, 70:757–790, 2007
18. J.-C. Nédélec. Mixed finite elements in \mathbf{R}^3 . *Numer. Math.*, 35:315–341, 1980
19. J.-C. Nédélec. A new family of mixed finite elements in \mathbf{R}^3 . *Numer. Math.*, 50:57–81, 1986
20. N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for linear convection-diffusion equations. *J. Comput. Phys.*, 228:3232–3254, 2009
21. N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations. *J. Comput. Phys.*, 228:8841–8855, 2009
22. N. C. Nguyen, J. Peraire, and B. Cockburn. A hybridizable discontinuous Galerkin method for Stokes flow. *Comput. Methods Appl. Mech. Engrg.*, 199:582–597, 2010
23. N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations. Submitted to Journal of Computational Physics, 2009.
24. N. C. Nguyen, J. Peraire, and B. Cockburn. A hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations. In Proceedings of the 48th AIAA Aerospace Sciences Meeting and Exhibit, Orlando, Florida, January 2010. AIAA-2010-362
25. J. Peraire, N. C. Nguyen, and B. Cockburn. A hybridizable discontinuous Galerkin method for the compressible Euler and Navier–Stokes Equations. In Proceedings of the 48th AIAA Aerospace Sciences Meeting and Exhibit, Orlando, Florida, January 2010. AIAA-2010-363
26. P. O. Persson and J. Peraire. Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier–Stokes equations *SIAM J. Sci. Comput.*, 30(6):2709–2733, 2008
27. S.-C. Soon, B. Cockburn, and H. K. Stolarski. A hybridizable discontinuous Galerkin method for linear elasticity. *Int. J. Numer. Methods Eng.*, 80(8):1058–1092, 2009