SIAM J. SCI. COMPUT.
Vol. 31, No. 5, pp. 3827–3846

A HYBRIDIZABLE DISCONTINUOUS GALERKIN METHOD FOR STEADY-STATE CONVECTION-DIFFUSION-REACTION PROBLEMS*

BERNARDO COCKBURN†, BO DONG‡, JOHNNY GUZMÁN‡, MARCO RESTELLI§, AND RICCARDO SACCO¶

Abstract. In this article, we propose a novel discontinuous Galerkin method for convection-diffusion-reaction problems, characterized by three main properties. The first is that the method is hybridizable; this renders it efficiently implementable and competitive with the main existing methods for these problems. The second is that, when the method uses polynomial approximations of the same degree for both the total flux and the scalar variable, optimal convergence properties are obtained for both variables; this is in sharp contrast with all other discontinuous methods for this problem. The third is that the method exhibits superconvergence properties of the approximation to the scalar variable; this allows us to postprocess the approximation in an element-by-element fashion to obtain another approximation to the scalar variable which converges faster than the original one. In this paper, we focus on the efficient implementation of the method and on the validation of its computational performance. With this aim, extensive numerical tests are devoted to explore the convergence properties of the novel scheme, to compare it with other methods in the diffusion-dominated regime, and to display its stability and accuracy in the convection-dominated case.

Key words. discontinuous Galerkin methods, hybridization, superconvergence, convection-diffusion

AMS subject classifications. 65N30, 65M60, 35L65

DOI. 10.1137/080728810

1. Introduction. This paper is devoted to the study of new discontinuous Galerkin (DG) methods for convection-diffusion-reaction problems. Three novel, interrelated features render these methods attractive. The first is that they are *hybridizable* and hence efficiently implementable; we refer to them as *local discontinuous Galerkin-hybridizable* (LDG-H) methods. The second is that they provide approximations for the flux which are *optimally convergent* when, on each element, both the flux and the scalar variable are approximated by polynomials of the same degree. Finally, the third feature is that the approximations to the scalar variable *superconverge*; this is why we call them superconvergent. As a consequence of this last property, a new approximation for the scalar unknown can be locally obtained which converges faster than the original approximation.

*Received by the editors June 28, 2008; accepted for publication (in revised form) August 6, 2009; published electronically October 14, 2009.
http://www.siam.org/journals/sisc/31-5/72881.html

†School of Mathematics, University of Minnesota, Vincent Hall, Minneapolis, MN 55455 (cockburn@math.umn.edu). This author's work was supported in part by the National Science Foundation (grant DMS-0712955) and by the University of Minnesota Supercomputing Institute.

‡Division of Applied Mathematics, Brown University, Providence, RI 02912 (bdong@brown.edu, guzman033@brown.edu). The second author's work was supported by an NSF Mathematical Science Postdoctoral Research Fellowship (DMS-0503050).

§Ozean im Erdsystem, Max-Planck-Institut für Metereologie, Bundesstraße 53, 20146 Hamburg, Germany (marco.restelli@zmaw.de). This author's work was supported by the MetStröm project of the Deutsche Forschungsgemeinschaft (DFG Schwerpunktprogramm 1276).

¶Dipartimento di Matematica "F. Brioschi," Politecnico di Milano, via Bonardi 9, 20133 Milano, Italy (riccardo.sacco@polimi.it). This author's work was supported by the MIUR grant 2006013187-003, "Approssimazione Numerica di Problemi Multiscala e Multifisica con Tecniche Adattive" (2006–2008).

We carry out the study of LDG-H methods as applied to the following convection-diffusion-reaction model problem:

$$\begin{aligned} -\nabla\cdot(\boldsymbol{a}\nabla u+\boldsymbol{\beta}u)+ru &= f \qquad \text{in } \Omega,\\ u &= g \qquad \text{on } \partial\Omega_D,\\ -(\boldsymbol{a}\nabla u+\boldsymbol{\beta}u)\cdot\boldsymbol{n} &= \mathsf{q_N} \qquad \text{on } \partial\Omega_N, \end{aligned}$$

where $\Omega\subset\mathbb{R}^d$ is a polyhedral domain ($d\geq 2$) with boundary $\partial\Omega=\partial\Omega_D\cup\partial\Omega_N$, $\partial\Omega_D\cap\partial\Omega_N=\emptyset$, $f\in L^2(\Omega)$, $\boldsymbol{a}$ is a symmetric $d\times d$ matrix function that is uniformly positive definite on Ω with components in $L^\infty(\Omega)$.

The work we present here can be considered part of the series of papers [6], [5], and [7]. Indeed, in [6] a unifying framework for the hybridization of mixed, discontinuous, continuous, and nonconforming methods for second order elliptic problems was proposed; however, no convection or reaction terms were considered. Later, in [5] this unifying framework was exploited to devise a new DG method, called the *single-face hybridizable* (SF-H) method, which can be thought of as being *in between* the Raviart–Thomas (RT) [15] and the Brezzi–Douglas–Marini (BDM) [2] mixed methods. The properties of optimal convergence of both the scalar (primal) and the flux (dual) variables, and of superconvergence of projections of the error of the scalar and hybrid variables, shared by the RT and BDM methods, were proven to hold for the SF-H method; again, no convection or reaction terms were considered. Recently, and for the same model problem, it was shown in [7] that these properties hold for many DG methods *not* belonging to the class DG of methods analyzed in [1]. On the other hand, the class of methods considered in [7] does include the LDG-H methods we consider here. In other words, we extend here some of the LDG-H methods considered in [7] to the case in which convection and reaction are added to the model equation.

The inclusion of the convective term in the context of hybridized mixed methods is not straightforward, as it can be done in several ways; see [9]. Here we have chosen to provide an approximation to the *total* flux $\boldsymbol{q}:=-(\boldsymbol{a}\nabla u+\boldsymbol{\beta}u)$, a variable that in many realistic applications has more importance than u itself. This is the case, for example, of reservoir simulation where u is the hydraulic pressure and $\boldsymbol{q}$ is Darcy's velocity [4], or the semiconductor device modeling where u represents the carrier concentration and $\boldsymbol{q}$ is the associated current density [13]. In other words, the weak formulation of our LDG-H methods is based on the following rewriting of the original equations:

$$\boldsymbol{c}\,\boldsymbol{q}+\nabla u+\boldsymbol{b}\,u=0 \qquad \text{in } \Omega, \tag{1.1a}$$

$$\nabla\cdot\boldsymbol{q}+ru=f \qquad \text{in } \Omega, \tag{1.1b}$$

$$u=g \qquad \text{on } \partial\Omega_D, \tag{1.1c}$$

$$\boldsymbol{q}\cdot\boldsymbol{n}=\mathsf{q_N} \qquad \text{on } \partial\Omega_N, \tag{1.1d}$$

where $\boldsymbol{c}:=\boldsymbol{a}^{-1}$ and $\boldsymbol{b}:=\boldsymbol{c}\boldsymbol{\beta}$.

We immediately see that the purely hyperbolic case, namely, the case $\boldsymbol{a}=\boldsymbol{0}$, is precluded by this rewriting. So, we restrict ourselves to dealing with the *diffusion-dominated* case, and we start our validation of the computational performance of the LDG-H method by showing experimentally that the convergence properties theoretically proved in [5] and [7], in the case where there is no convection or reaction, remain the same when these terms are present.

Then, we continue our numerical validation with a thorough comparison of the LDG-H formulation with the standard continuous Galerkin (CG) method and with

approaches that provide a direct approximation of the flux variable, namely, the dual mixed hybridized RT and BDM methods. This comparison shows that the LDG-H method performs remarkably well with respect to the other considered schemes. In particular, it turns out that the L^2-order of convergence of the approximate flux of a LDG-H method using polynomials of degree k (for $k \geq 1$) is the same as that of a CG scheme using polynomials of degree $k+1$, while the convergence rate of a LDG-H method using polynomials of degree k is one order higher than that of a BDM scheme using polynomials of degree k (for the flux). This latter result agrees with the conclusions of [9], where it is shown that suboptimal convergence of the BDM method is to be expected when the *total* advective-diffusive flux is treated as an independent variable instead of the sole diffusive flux.

Finally, we complete the validation process by testing the performance of the LDG-H method *also* in the *convection-dominated* case. Preliminary results demonstrate that an upwinding-like selection of the stabilization parameter τ allows the LDG-H method to accurately compute sharp fronts in the solution u, as with more established methods for problems with dominating hyperbolic terms [3, 11, 16], while computing at the same time a conservative and superconvergent approximation of the flux $\boldsymbol{q}$, as with standard mixed finite element schemes for elliptic problems.

The paper is organized as follows. In section 2, we define the LDG-H approximation, we address the issue of its efficient implementation using the hybridization technique, we discuss the choice of the finite element spaces and the stabilization parameter, and we present the local postprocessings. In section 3, we discuss the numerical experiments, and in section 4 we draw some concluding remarks.

2. The LDG-H methods.

2.1. Definition of the methods. In order to describe the LDG-H methods, we need to introduce some notation. We denote by $\Omega_h = \{K\}$ a triangulation of the domain Ω of shape-regular tetrahedra K, h_K being the diameter of K, and set $\partial\Omega_h := \{\partial K : K \in \Omega_h\}$. We associate with this triangulation the set of interior faces $\mathscr{E}_h^i$ and the set of boundary faces $\mathscr{E}_h^\partial$. We say that $e \in \mathscr{E}_h^i$ if there are two simplexes K^+ and K^- in Ω_h such that $e = \partial K^+ \cap \partial K^-$, and we say that $e \in \mathscr{E}_h^\partial$ if there is a simplex in Ω_h such that $e = \partial K \cap \partial\Omega$. We set $\mathscr{E}_h := \mathscr{E}_h^i \cup \mathscr{E}_h^\partial$.

The LDG-H methods seek an approximation $(\boldsymbol{q}_h, u_h, \lambda_h)$ to the exact solution $(\boldsymbol{q}|_\Omega, u|_\Omega, u|_{\mathscr{E}_h \setminus \partial\Omega_D})$ of (1.1) in a finite dimensional space $\boldsymbol{V}_h \times W_h \times M_h$ of the form

$$\boldsymbol{V}_h := \{\boldsymbol{v} \in \boldsymbol{L}^2(\Omega) : \boldsymbol{v}|_K \in \boldsymbol{V}(K) \quad \forall K \in \Omega_h\}, \tag{2.1a}$$

$$W_h := \{\omega \in L^2(\Omega) : \omega|_K \in W(K) \quad \forall K \in \Omega_h\}, \tag{2.1b}$$

$$M_h := \{\mathsf{m} \in L^2(\partial\Omega_h) : \mathsf{m}|_e \in M(e) \quad \forall e \in \mathscr{E}_h, \quad \mathsf{m}|_{\partial\Omega_D} = 0\}, \tag{2.1c}$$

and determines it by requiring that

$$(\boldsymbol{c}\,\boldsymbol{q}_h, \boldsymbol{v})_{\Omega_h} + (\boldsymbol{b}u, \boldsymbol{v})_{\Omega_h} - (u_h, \nabla\cdot\boldsymbol{v})_{\Omega_h} + \langle \widehat{u}_h, \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\partial\Omega_h} = 0, \tag{2.2a}$$

$$-(\boldsymbol{q}_h, \nabla\omega)_{\Omega_h} + \langle \widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}, \omega\rangle_{\partial\Omega_h} + (ru_h, \omega)_{\Omega_h} = (f, \omega)_{\Omega_h}, \tag{2.2b}$$

$$\langle \widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}, \mu\rangle_{\partial\Omega_h} = \langle \mathsf{q}_N, \mu\rangle_{\partial\Omega_N} \tag{2.2c}$$

for all $(\boldsymbol{v}, \omega, \mu) \in \boldsymbol{V}_h \times W_h \times M_h$. Here, we have used the notation

$$(\zeta, \omega)_{\Omega_h} := \sum_{K\in\Omega_h} \int_K \zeta(x)\,\omega(x)\,dx \quad \text{and} \quad \langle \zeta, \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\partial\Omega_h} := \sum_{K\in\Omega_h} \int_{\partial K} \zeta(\gamma)\,\boldsymbol{v}(\gamma)\cdot\boldsymbol{n}\,d\gamma,$$

and $(\boldsymbol{\sigma},\boldsymbol{v})_{\Omega_h} := \sum_{j=1}^{d}(\sigma_j, v_j)_{\Omega_h}$ for any functions $\boldsymbol{\sigma},\boldsymbol{v}$ in $\boldsymbol{H}^1(\Omega_h) := [H^1(\Omega_h)]^d$ and ζ,ω in $H^1(\Omega_h)$. The outward normal unit vector to ∂K is denoted by $\boldsymbol{n}$.

To complete the definition of the LDG-H method, the numerical traces $\widehat{\boldsymbol{q}}_h$ and $\widehat{u}_h$ have to be provided. Following [6], we take

$$\widehat{u}_h = \begin{cases} \mathsf{P}_\partial\, g & \text{on } \partial K \cap \partial\Omega_D, \\ \lambda_h & \text{otherwise,} \end{cases} \tag{2.3a}$$

$$\widehat{\boldsymbol{q}}_h = \boldsymbol{q}_h + \tau\,(u_h - \widehat{u}_h)\boldsymbol{n}, \tag{2.3b}$$

where P_∂ denotes an L^2-projection defined as follows. Given any function $\zeta \in L^2(\mathscr{E}_h)$ and an arbitrary face $e \in \mathscr{E}_h$, the restriction of $\mathsf{P}_\partial\zeta$ to e is defined as the element of $M(e)$ that satisfies

$$\langle \mathsf{P}_\partial\zeta - \zeta, \omega\rangle_e = 0 \qquad \forall\, \omega \in M(e). \tag{2.4}$$

The stabilization function τ is defined for each element $K \in \Omega_h$ in such a way that τ is nonnegative over ∂K and is constant over each face of ∂K. Notice that for each $e = \partial K_e^+ \cap \partial K_e^-$ we have, in general, $\tau^{K_e^+}|_e \neq \tau^{K_e^-}|_e$. More details about the choice of τ are discussed in section 2.3.

From definition (2.3a), we see that $\widehat{u}_h$ provides an approximation to u on *all* the faces of $\mathscr{E}_h$. Note also that the last equation of the weak formulation of the LDG-H method, (2.2c), is what was called in [6] the conservativity condition. Indeed, since such equation can be rewritten as

$$\sum_{e\in\mathscr{E}_h^i} \langle\, [\![\widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}]\!], \mu\rangle_e + \langle \widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}, \mu\rangle_{\partial\Omega_N} = \langle \mathsf{q}_\mathsf{N}, \mu\rangle_{\partial\Omega_N} \qquad \forall\, \mu \in M_h,$$

we see that, besides imposing the Neumann boundary conditions, it imposes that the jump of the normal component of the numerical trace $\widehat{\boldsymbol{q}}_h$, namely,

$$[\![\widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}]\!] := \widehat{\boldsymbol{q}}_h^+\cdot\boldsymbol{n}^+ + \widehat{\boldsymbol{q}}_h^-\cdot\boldsymbol{n}^-,$$

be equal to zero (weakly) across interelement boundaries. Such a numerical trace would then be single-valued (provided the space M_h is rich enough) and, hence, would have the important property of being conservative; see [1].

To ensure the existence and uniqueness of the approximate solution, we have to properly choose the local spaces $\boldsymbol{V}(K)$, $W(K)$, and $M(e)$ as well as the stabilization function τ. Before describing the choice we take in this paper, let us assume that the approximate solution is actually well defined, and let us show how the structure of these methods allows for them to be hybridized and, hence, efficiently implemented.

2.2. Implementation by hybridization. To do that, we extend the approach introduced in [6] to our setting. Thus, we begin by introducing the local solvers. The first local solver lifts functions on faces of the simplexes of the triangulation Ω_h to functions on Ω. It associates with each function m in $L^2(\mathscr{E}_h)$ the pair $(\mathfrak{Q}\mathsf{m}, \mathfrak{U}\mathsf{m})$ defined on each element K as the function in $\boldsymbol{V}(K)\times W(K)$ determined by requiring that the equations

$$(\boldsymbol{c}\,\mathfrak{Q}\mathsf{m}, \boldsymbol{v})_K + (\boldsymbol{b}\,\mathfrak{U}\mathsf{m}, \boldsymbol{v})_K - (\,\mathfrak{U}\mathsf{m}, \nabla\cdot\boldsymbol{v})_K = -\langle \mathsf{m}\,,\, \boldsymbol{v}\cdot\boldsymbol{n}\rangle_{\partial K}, \tag{2.5a}$$

$$-(\mathfrak{Q}\mathsf{m}, \nabla w)_K + \langle \widehat{\mathfrak{Q}}\mathsf{m}\cdot\boldsymbol{n}\,,\, w\rangle_{\partial K} + (r\,\mathfrak{U}\mathsf{m}, w)_K = 0, \tag{2.5b}$$

$$\widehat{\mathfrak{Q}}\mathsf{m} = \mathfrak{Q}\mathsf{m} + \tau\,(\,\mathfrak{U}\mathsf{m} - \mathsf{m})\,\boldsymbol{n} \tag{2.5c}$$

hold for all $(\boldsymbol{v}, w) \in \boldsymbol{V}(K) \times W(K)$. The second local solver associates with each $f \in L^2(\Omega)$ the pair $(\mathfrak{Q}f, \mathfrak{U}f)$ defined on each element K as the function in $\boldsymbol{V}(K) \times W(K)$ such that the equations

$$(\boldsymbol{c}\,\mathfrak{Q}f, \boldsymbol{v})_K + (\boldsymbol{b}\,\mathfrak{U}f, \boldsymbol{v})_K - (\,\mathfrak{U}f, \nabla \cdot \boldsymbol{v})_K = 0, \tag{2.6a}$$

$$-(\mathfrak{Q}f, \nabla w)_K + \langle \widehat{\mathfrak{Q}}f \cdot \boldsymbol{n}\,,\, w\rangle_{\partial K} + (r\,\mathfrak{U}f, w)_K = (f, w)_K, \tag{2.6b}$$

$$\widehat{\mathfrak{Q}}f = \mathfrak{Q}f + \tau\,\mathfrak{U}f\,\boldsymbol{n} \tag{2.6c}$$

hold for all $(\boldsymbol{v}, w) \in \boldsymbol{V}(K) \times W(K)$. Note that, to define these local solvers we have used the LDG-H method on each element $K \in \Omega_h$. Since we are assuming the LDG-H method is well defined, so are the local solvers.

Next, we give a characterization of the approximate solution in terms of the local solvers and the function $\lambda_h \in M_h$.

THEOREM 2.1 (characterization of the approximate solution). *Assume that there is a unique solution* $(\boldsymbol{q}_h, u_h) \in \boldsymbol{V}_h \times W_h$ *of* (2.2) *and* (2.3). *Then we have that*

$$(\boldsymbol{q}_h, u_h) = (\mathfrak{Q}\lambda_h + \mathfrak{Q}g + \mathfrak{Q}f,\, \mathfrak{U}\lambda_h + \mathfrak{U}g + \mathfrak{U}f),$$

the function $\lambda_h \in M_h$ *being the unique solution of*

$$a_h(\lambda_h, \mu) = b_h(\mu) \qquad \forall\, \mu \in M_h, \tag{2.7}$$

where $a_h(\eta, \mu) = \langle \mu\,,\, \widehat{\mathfrak{Q}}\eta \cdot \boldsymbol{n}\rangle_{\partial\Omega_h}$ *and* $b_h(\mu) = \langle \mu\,,\, (\widehat{\mathfrak{Q}}f + \widehat{\mathfrak{Q}}g) \cdot \boldsymbol{n}\rangle_{\partial\Omega_h} - \langle \mathsf{q_N}\,,\, \mu\rangle_{\partial\Omega_N}$ *for all* η *and* $\mu \in M_h$.

Proof. The function $(\tilde{\boldsymbol{q}}_h, \tilde{u}_h) := (\mathfrak{Q}\lambda_h + \mathfrak{Q}g + \mathfrak{Q}f,\, \mathfrak{U}\lambda_h + \mathfrak{U}g + \mathfrak{U}f) \in \boldsymbol{V}_h \times W_h$ clearly satisfies (2.2a) and (2.2b), by the very definition of the local solvers (2.5) and (2.6). It also satisfies (2.2c) since this is nothing but a rewriting of (2.7). This implies that $(\tilde{\boldsymbol{q}}_h, \tilde{u}_h)$ must be equal to the unique solution $(\boldsymbol{q}_h, u_h)$. ∎

Theorem 2.1 shows that the only degrees of freedom that are globally coupled are those of λ_h. Once λ_h is computed by solving problem (2.7), then the approximate solution can be easily obtained by solving the postprocessing problems (2.5) and (2.6) in an element-by-element fashion.

Theorem 2.1 also contains information about the sparsity structure of the matrix for the degrees of freedom of λ_h. Indeed, the variational formulation (2.7) defining $\lambda_h \in M_h$ can be rewritten in matrix form as

$$A\,[\lambda_h] = b, \tag{2.8}$$

where $[\mu]^t\, A\,[\lambda_h] = a_h(\lambda_h, \mu)$ and $[\mu]^t\, b = b_h(\mu)$. Here $[\mu]$ is the vector of coefficients of the representation of μ in a given basis of M_h. By Theorem 2.1,

$$A = \sum_{K \in \mathcal{T}_h} A_K \quad \text{and} \quad b = \sum_{K \in \mathcal{T}_h} b_K,$$

where the matrix A_K is defined by $[\mu]^t\, A_K\,[\eta] = -\langle \mu\,,\, \widehat{\mathfrak{Q}}\eta \cdot \boldsymbol{n}\rangle_{\partial K}$ and the matrix b_K by $[\mu]^t\, b_K = \langle \mu\,,\, (\widehat{\mathfrak{Q}}f + \widehat{\mathfrak{Q}}g) \cdot \boldsymbol{n}\rangle_{\partial K} - \langle \mathsf{q_N}\,,\, \mu\rangle_{\partial K \cap \partial\Omega_N}$. As discussed in [6], the only nonzero entries of the matrices A_K and b_K are the ones associated with the degrees of freedom of μ and η on the border of the element K. This implies, in particular, that the nonzero entries of the matrix A are blocks of square matrices of order $\dim M(e)$. In each block-row, there are at most five nonzero matrices in two-space dimensions if

we are using meshes made of triangles, and at most seven nonzero matrices in three-space dimensions if we are using meshes made of tetrahedra. The nonzero entries of A_K and b_K can be interpreted as the *local* stiffness matrix and load vector of a standard displacement-based finite element formulation. Their actual computation requires the solution of algebraic linear systems corresponding to the local problems (2.5) and (2.6). The method can thus be efficiently implemented; see section 3.2, where the method is compared with other finite element methods.

2.3. Choice of the local spaces and stabilization parameter. In this paper, we take the following local spaces:

$$\boldsymbol{V}(K) := \boldsymbol{\mathcal{P}}_k(K), \quad W(K) := \mathcal{P}_k(K), \quad \text{and} \quad M(e) := \mathcal{P}_k(e),$$

where k is a nonnegative integer and $\boldsymbol{\mathcal{P}}_k(K) := [\mathcal{P}_k(K)]^d$, $\mathcal{P}_k(D)$ being the space of polynomials of total degree at most k defined on the domain D.

The precise choice of the *local stabilization* function τ has a strong influence on the accuracy of the method, as shown in [5] and in [7] in the case in which there is no convection and no reaction. In order to borrow from these references the definition of τ and extend to the case where convection and reaction are presently the corresponding properties of the LDG-H formulation, we assume that problem (1.1) is in the diffusion-dominated regime, i.e., that there exists a positive constant $\gamma < 1$ such that the following coercivity condition is satisfied (cf. [10]):

$$\boldsymbol{\beta}^t \boldsymbol{c} \boldsymbol{\beta} \leq (1-\gamma)\, 4\, r. \tag{2.9}$$

Then, we introduce the following quantities:

$$\tau_K := \tau|_{e_K^\tau} \quad \text{and} \quad \overline{\tau}_K := \max \tau|_{\partial K \setminus e_K^\tau}, \tag{2.10}$$

e_K^τ being any face of K where $\tau|_{\partial K}$ is maximum. We notice that in [5] the function τ was chosen in such a way that $\overline{\tau}_K = 0$ for every $K \in \Omega_h$, whereas in [7], $\overline{\tau}_K$ was allowed to be different from zero. Proceeding as in [5, 7], we can prove the following result.

THEOREM 2.2 (well-posedness of the LDG-H method). *Assume that* (2.9) *is satisfied and that* $\tau_K > 0$ *for each* $K \in \Omega_h$. *Then, there is a unique solution* $(\boldsymbol{q}_h, u_h) \in \boldsymbol{V}_h \times W_h$ *of* (2.2) *and* (2.3), *or, equivalently, the linear algebraic system* (2.8) *is uniquely solvable.*

The assumptions in Theorem 2.2 are verified by the LDG-H method used in section 3.1, where we investigate how its convergence properties depend on the actual choice of τ_K and $\overline{\tau}_K$. An alternative to the coercivity assumption (2.9) consists of assuming that the mesh size h is small enough, as will be shown in detail in a forthcoming paper devoted to the analysis of the LDG-H method. To conclude this discussion, one may legitimately wonder about the behavior of the LDG-H formulation proposed in the present article in the important case where problem (1.1) is convection-dominated. With this purpose, the numerical experiments of section 3.3 show that if the definition of the stabilization function τ enforces the classical *upwinding stabilization* of the classical DG method for hyperbolic problems, then the LDG-H method turns out to be well defined also in the convection-dominated regime.

2.4. Postprocessing of the approximate solution.

Postprocessing of the total flux. Let us show how to postprocess $\boldsymbol{q}_h$ and $\widehat{\boldsymbol{q}}_h$ with an element-by-element procedure to obtain an optimally convergent approximation of $\boldsymbol{q}$, denoted $\boldsymbol{q}_h^\star$, belonging to H(div, Ω); see [5, 7] and the references therein. On each simplex $K \in \Omega_h$, we define the function $\boldsymbol{q}_h^\star$ as the only element of $\boldsymbol{\mathcal{P}}^k(K) + \boldsymbol{x}\,\mathcal{P}^k(K)$ satisfying, for $k \geq 0$,

$$\langle (\boldsymbol{q}_h^\star - \widehat{\boldsymbol{q}}_h)\cdot \boldsymbol{n}, \mu\rangle_e = 0 \quad \forall \mu \in \mathcal{P}^k(e)\ \forall \text{ faces } e \text{ of } K, \tag{2.11a}$$

$$(\boldsymbol{q}_h^\star - \boldsymbol{q}_h, \boldsymbol{v})_K = 0 \quad \forall \boldsymbol{v} \in \boldsymbol{\mathcal{P}}^{k-1}(K). \tag{2.11b}$$

Clearly, if $k = 0$, the second set of equations is empty. Note that the function $\boldsymbol{q}_h^\star$ belongs to H(div, Ω).

Postprocessing of the scalar variable. Next, we show how to postprocess u_h and $\widehat{\boldsymbol{q}}_h$ with an element-by-element procedure to obtain a better approximation to u, denoted $u_h^\star$. With this aim, we assume that the advective flow is potential driven, i.e., $\boldsymbol{\beta} := \boldsymbol{a}\nabla\xi$, for a given potential function $\xi \in W^{1,\infty}(\Omega)$, as typically happens in the modeling of electrochemical ionic transport [17] and in semiconductor device simulation [13]. Let us introduce the change of dependent variable

$$u := \nu\, e^{-\xi},$$

which is a special instance of the classical Cole–Hopf transformation employed in the study of parabolic equations with quadratic gradient nonlinearities [8, 12]. Using the above change of variable, the advective-diffusive flux $\boldsymbol{q} = -(\boldsymbol{a}\nabla u + \boldsymbol{\beta}u)$ can be written in the following equivalent diffusive form: $\boldsymbol{q} = -\boldsymbol{a}e^{-\xi}\nabla\nu$. On each element $K \in \Omega_h$, we define ν_h^*, an approximation to ν, as follows. First, assume that $r|_K = 0$. Then we write $\nu_h = \tilde{\nu}_h + \overline{\nu}_h$, where $\overline{\nu}_h := \frac{1}{|K|}\int_K u_h\, e^{\xi}\, dx$, and $\tilde{\nu}_h$ is the element of $\mathcal{P}_0^{k+1}(K)$ which satisfies

$$(\boldsymbol{a}e^{-\xi}\nabla\tilde{\nu}_h, \nabla w)_K = (f, w)_K - \langle \widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}, w\rangle_{\partial K} \quad \forall w \in \mathcal{P}_0^{k+1}(K).$$

If $r|_K$ is not zero, we take ν_h to be the element of $\mathcal{P}^{k+1}(K)$ such that

$$(\boldsymbol{a}e^{-\xi}\nabla\nu_h, \nabla w)_K + (re^{-\xi}\nu_h, w)_K = (f, w)_K - \langle \widehat{\boldsymbol{q}}_h\cdot\boldsymbol{n}, w\rangle_{\partial K} \quad \forall w \in \mathcal{P}^{k+1}(K).$$

Finally, the approximation u_h^* of u, called *exponential-fitting*, is defined by

$$u_h^* := \nu_h^* e^{-\xi},$$

where $\nu_h^* := \tilde{\nu}_h + \overline{\nu}_h$. Notice that u_h^*, unlike ν_h^*, is *not* a polynomial.

Let us end by emphasizing that, when $\boldsymbol{\beta}$ is not potential driven, it is still possible to construct u_h^* by using a straightforward extension to the present case of the corresponding quantity defined in the purely elliptic case in [5, 7].

3. Numerical results. This section is devoted to an extensive validation of the computational performance of the LDG-H method, in terms of convergence, accuracy, stability, and efficiency, as applied to problem (1.1) in the case where Ω is a two-dimensional polygon ($d = 2$).

Section 3.1 deals with the experimental study of the theoretical convergence properties of the method in the *diffusion-dominated* regime. Special emphasis will be put

on investigating the dependence of the orders of convergence of the method upon different choices of the stabilization parameters τ_K and $\overline{\tau}_K$ defined by (2.10).

Section 3.2 deals with the comparison of the LDG-H formulation with the standard conforming Galerkin (CG) finite element method and with approaches that provide a direct approximation of the flux variable, namely, the dual mixed hybridized RT and BDM methods. Exploiting the similarities among the LDG-H, BDM, and RT formulations, these three methods are implemented in a single FORTRAN90 code to allow the user a flexible tool for immediate cross-validation of the various methods. In the case of the BDM and RT mixed schemes, τ_K and $\overline{\tau}_K$ are set equal to zero.

The concluding section (section 3.3) discusses a preliminary assessment of the performance of the LDG-H method in the *convection-dominated* regime.

Throughout the section, we use uniform grids of triangles. The grid associated to the integer "l" is obtained by meshing the domain with squares of size $h = 2^{-l}$, which are then divided into two triangles. We also use the weighted norm $\|\cdot\|_{L^2(\Omega_h;\boldsymbol{c})} = (\boldsymbol{c}\,\cdot,\cdot)^{1/2}_{\Omega_h}$, and the projection P^ℓ defined as follows. Given a function $\zeta \in H^1(\Omega_h)$ and an arbitrary simplex $K \in \Omega_h$, the restriction of $\mathsf{P}^\ell\zeta$ to K is (for $\ell \geq 0$) the L^2-projection of ζ over $\mathcal{P}^\ell(K)$. If $\ell < 0$, we let P^ℓ be the zero operator. Moreover, we use the following norm for a function $\eta \in L^2(\mathscr{E}_h)$:

$$\|\,\eta\,\|_{L^2(\mathscr{E}_h;h)} = \left(\sum_{K\in\Omega_h} h_K \|\,\eta\,\|^2_{L^2(\partial K)}\right)^{1/2}.$$

3.1. Validation in the diffusion-dominated regime. The test case considered in this section is a modification of that proposed in [14]. The problem data are as follows: $\Omega = [0,1]\times[0,1]$, $\boldsymbol{a} = \varepsilon\boldsymbol{I}$, where $\boldsymbol{I}$ denotes the 2×2 identity matrix, $\boldsymbol{\beta} = [-x^2, -y^4]^T$, $r = x + y^3$, and $f = xy\left(4x\eta_5^\varepsilon(y) + 6y^3\eta_3^\varepsilon(x)\right)$, having defined for $s \geq 1$ the function $\eta_s^\varepsilon(t) = 1 - \exp\left((t^s-1)/(s\varepsilon)\right)$. The exact solution of the problem is $u(x,y) = x\,y\,\eta_3^\varepsilon(x)\,\eta_5^\varepsilon(y)$. Here we take $\varepsilon = 0.5$, so that the coercivity condition (2.9) is satisfied for all $x, y \in \overline{\Omega}$, and problem (1.1) models a *diffusion-dominated* regime.

The aim of the numerical experiments is to explore the convergence properties of the method for different choices of the stabilization parameters τ_K and $\overline{\tau}_K$. Tables 3.1, 3.2, and 3.3 indicate that the orders of convergence for $\|\,e_{\boldsymbol{q}}\,\|_{L^2(\Omega_h;\boldsymbol{c})}$, $\|\,e_u\,\|_{L^2(\Omega_h)}$, $\|\,e_{\boldsymbol{q}^\star}\,\|_{L^2(\Omega_h;\boldsymbol{c})}$, and $\|\,e_{u^\star}\,\|_{L^2(\Omega_h)}$ are *optimal* as long as $\overline{\tau}_K \leq C \leq \tau_K$ for all $K \in \Omega_h$, C being a positive constant. This means that the optimal combinations of $\overline{\tau}_K$ and τ_K correspond to the intersection of the second and third columns with the first and second rows of the above mentioned tables. These results agree with the theoretical estimates proved in [7] in the purely elliptic case. Moreover, if $\overline{\tau}_K$ is set equal to zero for all $K \in \Omega_h$, it turns out that optimal convergence orders can be obtained only if τ_K is not too small, in full agreement with the results in [5] for the purely diffusive case.

Next, to evaluate the effect of the polynomial degree k on the efficiency of the LDG-H method, we plot in Figure 3.1 the errors as a function of the computational complexity (see also the numerical values in Table 3.4), which we *define* as the number of nonzero entries of the stiffness matrix after the static condensation of the local degrees of freedom. We refer to section 3.2 for some considerations concerning the computational cost of the local problems arising from the lifting operators.

For these tests, the stabilization parameters are $\tau_K = \overline{\tau}_K = 1$; similar results are obtained for different choices of τ_K and $\overline{\tau}_K$. For the problems under consideration, the number of nonzero entries of the above-mentioned stiffness matrix for the LDG-H

TABLE 3.1
Experimental orders of convergence of $e_{\boldsymbol{q}} = \| \boldsymbol{q} - \boldsymbol{q}_h \|_{L^2(\Omega_h;\mathbb{C})}$ *and* $e_u = \| u - u_h \|_{L^2(\Omega_h)}$ *for* $k \geq 1$.

	$\overline{\tau}_K = \mathcal{O}(\frac{1}{h})$		$\overline{\tau}_K = \mathcal{O}(1)$		$\overline{\tau}_K = \mathcal{O}(h), 0$	
	$e_{\boldsymbol{q}}$	e_u	$e_{\boldsymbol{q}}$	e_u	$e_{\boldsymbol{q}}$	e_u
$\tau_K = \mathcal{O}(\frac{1}{h})$	k	$k+1$	$k+1$	$k+1$	$k+1$	$k+1$
$\tau_K = \mathcal{O}(1)$	$-$	$-$	$k+1$	$k+1$	$k+1$	$k+1$
$\tau_K = \mathcal{O}(h)$	$-$	$-$	$-$	$-$	k	k

TABLE 3.2
Experimental orders of convergence of $e_{\boldsymbol{q}^\star} = \| \boldsymbol{q} - \boldsymbol{q}_h^\star \|_{L^2(\Omega_h;\mathbb{C})}$ *and* $e_{u^\star} = \| u - u_h^\star \|_{L^2(\Omega_h)}$, *or* $e_{u^\star} = \| \mathsf{P}^{k-1}(u - u_h) \|_{L^2(\Omega_h)}$, *or* $e_{u^\star} = \| \mathsf{P}_\partial u - \widehat{u}_h \|_{L^2(\mathscr{E}_h;h)}$ *for* $k \geq 1$.

	$\overline{\tau}_K = \mathcal{O}(\frac{1}{h})$		$\overline{\tau}_K = \mathcal{O}(1)$		$\overline{\tau}_K = \mathcal{O}(h), 0$	
	$e_{\boldsymbol{q}^\star}$	$e_{u^\star}$	$e_{\boldsymbol{q}^\star}$	$e_{u^\star}$	$e_{\boldsymbol{q}^\star}$	$e_{u^\star}$
$\tau_K = \mathcal{O}(\frac{1}{h})$	k	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$
$\tau_K = \mathcal{O}(1)$	$-$	$-$	$k+1$	$k+2$	$k+1$	$k+2$
$\tau_K = \mathcal{O}(h)$	$-$	$-$	$-$	$-$	k	$k+1$

TABLE 3.3
Experimental orders of convergence of $e_{\boldsymbol{q}} = \| \boldsymbol{q} - \boldsymbol{q}_h \|_{L^2(\Omega_h;\mathbb{C})}$ *and* $e_u = \| u - u_h \|_{L^2(\Omega_h)}$ *for* $k = 0$.

	$\overline{\tau}_K = \mathcal{O}(\frac{1}{h})$		$\overline{\tau}_K = \mathcal{O}(1)$		$\overline{\tau}_K = \mathcal{O}(h), 0$	
	$e_{\boldsymbol{q}}$	e_u	$e_{\boldsymbol{q}}$	e_u	$e_{\boldsymbol{q}}$	e_u
$\tau_K = \mathcal{O}(\frac{1}{h})$	0	0	1	1	1	1
$\tau_K = \mathcal{O}(1)$	$-$	$-$	1	1	1	1
$\tau_K = \mathcal{O}(h)$	$-$	$-$	$-$	$-$	0	0

method using polynomials of degree k on a uniform mesh of size $h = 1/n$ is, for $n \geq 3$,

$$(k+1)^2\,(15\,n^2 - 18\,n + 4).$$

In view of the forthcoming sections, we also note that for the BDM and RT methods the number of nonzero entries of the stiffness matrix is the same as for the LDG-H method, while for the CG method it is

$$(15\,k^2 - 6\,k - 2)\,n^2 + (-18\,k^2 - 20\,k^2 + 16)\,n + (4\,k^2 + 24\,k - 11).$$

Results in Figure 3.1 show that the efficiency of the LDG-H method increases with the polynomial degree k.

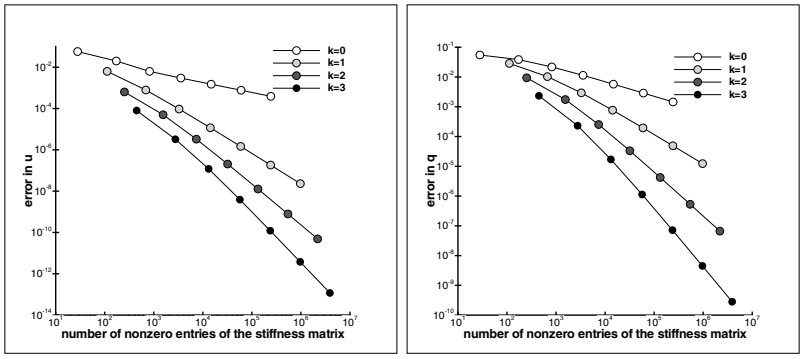

FIG. 3.1. *The effect of the polynomial degree k on the convergence properties of the error in u, $\| u - u_h^* \|_{L^2(\Omega)}$ (left), and on those of the error in $\boldsymbol{q}$, $\| \boldsymbol{q} - \boldsymbol{q}_h^* \|_{\boldsymbol{L}^2(\Omega)}$ (right).*

TABLE 3.4
Comparison between the LDG-H and CG methods.

		$\|u - u_h^{\mathrm{CG}}\|_{L^2(\Omega_h)}$		$\|u - u_h^{\mathrm{LDG\text{-}H}}\|_{L^2(\Omega_h)}$		$\|\boldsymbol{q} - \boldsymbol{q}_h^{\mathrm{CG}}\|_{L^2(\Omega_h;\mathbb{C})}$		$\|\boldsymbol{q} - \boldsymbol{q}_h^{\mathrm{LDG\text{-}H}}\|_{L^2(\Omega_h;\mathbb{C})}$	
k	l	error	order	error	order	error	order	error	order
0	1			2.64e-02	-			6.07e-02	-
	2			1.49e-02	0.82			4.22e-02	0.52
	3			7.60e-03	0.97			2.48e-02	0.77
	4			3.77e-03	1.01			1.33e-02	0.90
	5			1.87e-03	1.01			6.86e-03	0.96
	6			9.29e-04	1.01			3.47e-03	0.98
	7			4.63e-04	1.01			1.75e-03	0.99
1	1	1.43e-02	-	8.10e-03	-	8.03e-02	-	2.55e-02	-
	2	5.21e-03	1.45	2.54e-03	1.67	5.14e-02	0.64	1.05e-02	1.28
	3	1.49e-03	1.81	7.11e-04	1.84	2.82e-02	0.87	3.15e-03	1.74
	4	3.86e-04	1.95	1.85e-04	1.94	1.45e-02	0.96	8.37e-04	1.91
	5	9.74e-05	1.99	4.71e-05	1.98	7.29e-03	0.99	2.14e-04	1.97
	6	2.44e-05	2.00	1.18e-05	1.99	3.65e-03	1.00	5.39e-05	1.99
	7	6.10e-06	2.00	2.97e-06	2.00	1.83e-03	1.00	1.35e-05	1.99
2	1	3.45e-03	-	1.94e-03	-	3.57e-02	-	9.63e-03	-
	2	6.57e-04	2.39	4.13e-04	2.23	1.35e-02	1.40	2.04e-03	2.24
	3	9.25e-05	2.83	6.36e-05	2.70	3.97e-03	1.77	3.06e-04	2.74
	4	1.18e-05	2.97	8.52e-06	2.90	1.04e-03	1.93	4.05e-05	2.92
	5	1.49e-06	2.99	1.09e-06	2.97	2.64e-04	1.98	5.16e-06	2.97
	6	1.86e-07	3.00	1.37e-07	2.99	6.63e-05	1.99	6.49e-07	2.99
	7	2.33e-08	3.00	1.72e-08	2.99	1.65e-05	2.00	8.13e-08	3.00
3	1	8.86e-04	-	5.64e-04	-	1.32e-02	-	2.87e-03	-
	2	9.95e-05	3.16	6.83e-05	3.05	2.84e-03	2.22	3.05e-04	3.23
	3	7.38e-06	3.75	5.39e-06	3.66	4.27e-04	2.74	2.29e-05	3.73
	4	4.71e-07	3.97	3.63e-07	3.89	5.57e-05	2.94	1.52e-06	3.91
	5	2.89e-08	4.03	2.32e-08	3.97	6.99e-06	2.99	9.68e-08	3.97
	6	1.77e-09	4.03	1.46e-09	3.99	8.71e-07	3.01	6.09e-09	3.99
	7	1.09e-10	4.02	9.17e-11	4.00	1.08e-07	3.01	3.81e-10	4.00

3.2. Comparison with other finite element approximations. This section deals with the comparison of the LDG-H formulation with the CG finite element method and with the hybridized versions of the RT and BDM mixed methods. Defining a metric for such a comparison is not obvious, since the considered methods present not only quantitative differences, such as number of degrees of freedom and error norms, but also qualitative ones, such as local conservation properties, possible upwinding strategies, and ease of *hp* grid adaptation. Here we restrict ourselves to comparing different error norms as functions of the computational complexity. In general terms, we have that the computational complexity of a finite element method, and hence its computational cost, is the sum of two components: the construction of the local stiffness matrices (in our case, the matrices A_K introduced in section 2.2), and the construction and resolution of the resulting global linear system (in our case, problem (2.8)). The first component essentially requires a loop over the elements and, for methods with hybridization, also includes the local solvers $\mathbf{Q}$ and $\mathcal{U}$, while the second component requires a linear solver. In general, it is not possible to identify *a*

priori which of these two components will be responsible for the largest computational cost, since they scale differently when changing either the size of the problem or the implementation. The first component scales linearly with the number of elements, and when considering parallel implementations it does not pose any difficulty. The second component, on the contrary, can be expected to increase more than linearly with the number of elements, and it also represents a limitation to an efficient parallel implementation because of the implied global communications. For this reason, the computational costs of these two components should be kept separate. Here, we mainly consider the complexity associated with the linear system, as it is defined in section 3.1, since its definition is independent from the particular implementation. To complete this information, we also include some results concerning the CPU times for the sole computation of the local matrices.

We consider the same test case as in the previous section. We start with the comparison between the LDG-H method, with stabilization parameters $\tau_K = \overline{\tau}_K = 1$, and the CG method. Table 3.4 illustrates the obtained results for the errors $\|u-u_h\|_{L^2(\Omega_h)}$ and $\|\boldsymbol{q} - \boldsymbol{q}_h\|_{L^2(\Omega_h;\mathbf{c})}$, respectively. Therein, we have denoted the solutions provided by the LDG-H and CG schemes by the superscripts "LDG-H" and "CG," respectively.

We can see that the error $\|u - u_h\|_{L^2(\Omega_h)}$ converges with the optimal order of $k+1$ for both methods. The result is expected for the CG method, while for the LDG-H method it agrees with conclusions drawn in the previous section. As far as the amplitude of the error, it turns out that the LDG-H solution is more accurate than that of the CG method. This should be ascribed to the fact that the total number of degrees of freedom over Ω_h in the case of the LDG-H method is far superior to the corresponding quantity for a CG method of the same polynomial degree. We can also see that the order of convergence of the error $\|\boldsymbol{q} - \boldsymbol{q}_h\|_{L^2(\Omega_h;\boldsymbol{c})}$ for the LDG-H scheme of degree k is the same as that of a CG scheme of degree $k+1$. This is to be expected, because $\|\boldsymbol{q}-\boldsymbol{q}_h^{CG}\|_{L^2(\Omega_h;\boldsymbol{c})}$ is, roughly speaking, the H^1-error on u for the CG method. The CPU time for the computation of the local matrices in the two methods is given in Figure 3.2, for two serial implementations with comparable optimization. The same computational grid with $n = 32$ is used for different polynomial orders k, and problem coding is organized assuming that all coefficients in (1.1) are spatially varying functions, so that the local matrices cannot be precomputed. On the x-axis in Figure 3.2, we indicate the order $k+1$ for the LDG-H method and k for the CG method, while on the y-axis we indicate the quantity

$$R = \frac{t_A/\mathtt{nnz}(\mathtt{A})}{t_{A_{CG_1}}/\mathtt{nnz}(\mathtt{A}_{\mathtt{CG}_1})},$$

t_A and $\mathtt{nnz}(\mathtt{A})$ denoting (for each method) the CPU time required to assembly the local stiffness matrices A_K, and the number of nonzero entries of A, respectively. Moreover, for the LDG-H method, the time spent in the postprocessing of the scalar variable is also indicated, while we do not include $\boldsymbol{q}_h^\star$ since the projector (2.11) does not depend on the problem coefficients and the associated cost is negligible. Results clearly show that the computational effort required by the LDG-H method to construct the local stiffness matrices is comparable with that required by a CG method of the same degree, the difference of the costs being mainly due to the computation of the lifting operator $\mathfrak{Q}$, especially for large values of k. A key point of the efficient implementation of LDG-H is the use of the Cholesky factorization for inverting the mass matrix obtained from $(\boldsymbol{c}\,\mathfrak{Q}\mathsf{m}, \boldsymbol{v})_K$ in (2.5a). Further improvement is possible in the case where the diffusion coefficient is constant, by choosing an orthogonal basis

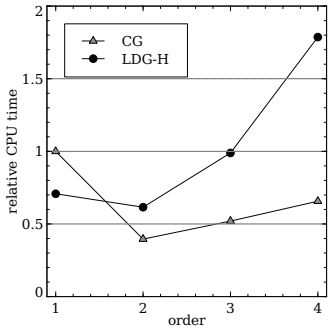

FIG. 3.2. *Normalized CPU time for the CG and LDG-H methods, in correspondence with different polynomial orders k and fixed spatial resolution $l = 5$. On the x axis, the order is k for the CG method and $k + 1$ for the LDG-H method.*

TABLE 3.5
The local spaces LDG-H$_k(K)$, BDM$_k(K)$, and RT$_k(K)$.

method	$\boldsymbol{V}(K) \times W(K)$	Dimension	Degree
BDM$_k$	$\boldsymbol{\mathcal{P}}^k(K) \times \mathcal{P}^{k-1}(K)$	$(k+1)(3k/2+2)$	$k \geq 1$
LDG-H$_k$	$\boldsymbol{\mathcal{P}}^k(K) \times \mathcal{P}^k(K)$	$(k+1)(3k/2+3)$	$k \geq 0$
RT$_k$	$(\boldsymbol{\mathcal{P}}^k(K) + \boldsymbol{x}\,\mathcal{P}^k(K)) \times \mathcal{P}^k(K)$	$(k+1)(3k/2+4)$	$k \geq 0$

for $\boldsymbol{V}_h$. Finally, we mention that an experimental analysis of the conditioning of the global stiffness matrix A reveals that it is of the same order for the two methods, irrespectively from the choice of τ_K and $\overline{\tau}_K$, and grows like $\mathcal{O}(h^{-2})$ as in standard displacement-based formulations.

Let us now compare the LDG-H method and the RT and BDM methods. We denote the solutions given by the RT and BDM methods by the superscripts "BDM" and "RT," respectively. The stabilization parameters for the LDG-H scheme are the same as in the previous comparison.

Note that for these three methods, the space M_h is *identical* since they all take $M(e) = \mathcal{P}_k(e)$. However, the spaces used to compute the local solvers $\boldsymbol{V}(K) \times W(K)$ on each element K are different for each of these methods. In Table 3.5 we display those spaces. Note that for each $k \geq 1$, we have

$$RT_{k-1}(K) \subset BDM_k(K) \subset \text{LDG-H}_k(K) \subset RT_k(K),$$

with

$$\dim(RT_k(K)) - \dim(\text{LDG-H}_k(K)) = \dim(\text{LDG-H}_k(K)) - \dim(BDM_k(K)) = k + 1.$$

In Table 3.6 we display the history of convergence of the approximations given by the LDG-H, BDM, and RT methods. For the LDG-H method, we consider the quantities $\|u - u_h^*\|_{L^2(\Omega_h)}$, $\|\boldsymbol{q} - \boldsymbol{q}_h^\star\|_{L^2(\Omega_h;\mathbf{c})}$, and $\|\nabla \cdot (\boldsymbol{q} - \boldsymbol{q}_h^\star)\|_{L^2(\Omega_h)}$, whereas for the BDM and RT methods, the postprocessed flux $\boldsymbol{q}_h^\star$ is replaced by the approximate flux $\boldsymbol{q}_h$ directly computed by the BDM and RT formulations.

We can see that the error $\|u - u_h^*\|_{L^2(\Omega_h)}$ converges with the optimal order of $k+2$ for the LDG-H and RT methods, whereas it converges with the suboptimal order of $k + 1$ for the BDM method. This degradation of accuracy is to be ascribed to the presence of convection in the differential model, and is consistent with the theoretical

TABLE 3.6
Convergence history for the LDG-H, BDM, and RT methods.

		$\|u - u_h^{*,\mathrm{LDG\text{-}H}}\|_{L^2(\Omega_h)}$		$\|\boldsymbol{q} - \boldsymbol{q}_h^{\star,\mathrm{LDG\text{-}H}}\|_{L^2(\Omega_h;\mathbf{c})}$		$\|\nabla\cdot(\boldsymbol{q} - \boldsymbol{q}_h^{\star,\mathrm{LDG\text{-}H}})\|_{L^2(\Omega_h)}$	
k	l	error	order	error	order	error	order
0	1	5.72e-02	-	4.78e-02	-	2.00e-01	-
	2	1.99e-02	1.52	3.72e-02	0.36	1.13e-01	0.82
	3	6.82e-03	1.55	2.16e-02	0.78	6.10e-02	0.89
	4	2.98e-03	1.19	1.13e-02	0.93	3.13e-02	0.96
	5	1.50e-03	0.99	5.75e-03	0.98	1.58e-02	0.99
	6	7.66e-04	0.97	2.89e-03	0.99	7.90e-03	1.00
	7	3.89e-04	0.98	1.45e-03	1.00	3.95e-03	1.00
1	1	6.29e-03	-	2.68e-02	-	7.20e-02	-
	2	7.87e-04	3.00	1.01e-02	1.41	2.91e-02	1.31
	3	9.44e-05	3.06	2.93e-03	1.79	8.41e-03	1.79
	4	1.17e-05	3.02	7.68e-04	1.93	2.19e-03	1.94
	5	1.46e-06	3.00	1.95e-04	1.98	5.53e-04	1.99
	6	1.83e-07	2.99	4.91e-05	1.99	1.38e-04	2.00
	7	2.31e-08	2.99	1.23e-05	2.00	3.46e-05	2.00
2	1	6.34e-04	-	8.90e-03	-	3.17e-02	-
	2	4.99e-05	3.67	1.72e-03	2.37	6.33e-03	2.32
	3	3.28e-06	3.93	2.52e-04	2.77	9.14e-04	2.79
	4	2.05e-07	4.00	3.31e-05	2.93	1.19e-04	2.94
	5	1.27e-08	4.02	4.21e-06	2.98	1.50e-05	2.99
	6	7.86e-10	4.01	5.29e-07	2.99	1.88e-06	3.00
	7	4.89e-11	4.01	6.62e-08	3.00	2.53e-07	3.00
3	1	7.98e-05	-	2.25e-03	-	9.01e-03	-
	2	3.25e-06	4.62	2.29e-04	3.29	8.85e-04	3.34
	3	1.20e-07	4.76	1.71e-05	3.74	6.42e-05	3.78
	4	3.89e-09	4.95	1.13e-06	3.92	4.19e-06	3.94
	5	1.22e-10	5.00	7.16e-08	3.98	2.64e-07	3.98
	6	3.78e-12	5.01	4.49e-09	3.99	1.66e-08	4.00
	7	1.18e-13	5.00	2.81e-10	4.00	1.04e-09	4.00
		$\|u - u_h^{*,\mathrm{BDM}}\|_{L^2(\Omega_h)}$		$\|\boldsymbol{q} - \boldsymbol{q}_h^{\mathrm{BDM}}\|_{L^2(\Omega_h;\mathbf{c})}$		$\|\nabla\cdot(\boldsymbol{q} - \boldsymbol{q}_h^{\mathrm{BDM}})\|_{L^2(\Omega_h)}$	
k	l	error	order	error	order	error	order
1	1	7.84e-03	-	2.96e-02	-	1.98e-01	-
	2	1.31e-03	2.58	1.16e-02	1.35	1.15e-01	0.79
	3	2.89e-04	2.18	4.17e-03	1.48	6.11e-02	0.91
	4	7.25e-05	2.00	1.64e-03	1.35	3.11e-02	0.97
	5	1.86e-05	1.96	7.41e-04	1.15	1.56e-02	0.99
	6	4.78e-06	1.96	3.59e-04	1.04	7.82e-03	1.00
	7	1.22e-06	1.96	1.78e-04	1.01	3.91e-03	1.00
2	1	1.02e-03	-	1.00e-02	-	7.52e-02	-
	2	1.17e-04	3.12	2.36e-03	2.09	2.93e-02	1.36
	3	1.10e-05	3.41	4.75e-04	2.32	8.35e-03	1.81
	4	1.11e-06	3.31	1.03e-04	2.20	2.16e-03	1.95
	5	1.26e-07	3.14	2.43e-05	2.09	5.45e-04	1.99
	6	1.52e-08	3.05	5.93e-06	2.04	1.37e-04	2.00
	7	1.88e-09	3.01	1.47e-06	2.02	3.42e-05	2.00
3	1	1.18e-04	-	3.08e-03	-	3.30e-02	-
	2	6.74e-06	4.14	3.68e-04	3.07	6.25e-03	2.40
	3	4.03e-07	4.06	4.10e-05	3.17	8.87e-04	2.82
	4	2.43e-08	4.05	4.84e-06	3.08	1.15e-04	2.95
	5	1.44e-09	4.08	5.88e-07	3.04	1.45e-05	2.99
	6	8.56e-11	4.07	7.23e-08	3.02	1.81e-06	3.00
	7	5.18e-12	4.05	8.95e-09	3.01	2.27e-07	3.00
		$\|u - u_h^{*,\mathrm{RT}}\|_{L^2(\Omega_h)}$		$\|\boldsymbol{q} - \boldsymbol{q}_h^{\mathrm{RT}}\|_{L^2(\Omega_h;\mathbf{c})}$		$\|\nabla\cdot(\boldsymbol{q} - \boldsymbol{q}_h^{\mathrm{RT}})\|_{L^2(\Omega_h)}$	
k	l	error	order	error	order	error	order
0	1	1.18e-02	-	5.26e-02	-	1.98e-01	-
	2	4.39e-03	1.42	3.69e-02	0.51	1.15e-01	0.79
	3	1.29e-03	1.77	2.10e-02	0.81	6.11e-02	0.91
	4	3.37e-04	1.93	1.09e-02	0.95	3.11e-02	0.97
	5	8.55e-05	1.98	5.50e-03	0.99	1.56e-02	0.99
	6	2.15e-05	1.99	2.75e-03	1.00	7.82e-03	1.00
	7	5.38e-06	2.00	1.38e-03	1.00	3.91e-03	1.00
1	1	2.71e-03	-	2.62e-02	-	7.52e-02	-
	2	4.62e-04	2.55	9.52e-03	1.46	2.93e-02	1.36
	3	6.32e-05	2.87	2.73e-03	1.80	8.35e-03	1.81
	4	8.05e-06	2.97	7.15e-04	1.93	2.16e-03	1.95
	5	1.01e-06	3.00	1.82e-04	1.97	5.45e-04	1.99
	6	1.25e-07	3.00	4.58e-05	1.99	1.37e-04	2.00
	7	1.56e-08	3.00	1.15e-05	2.00	3.42e-05	2.00
2	1	6.07e-04	-	8.51e-03	-	3.30e-02	-
	2	4.72e-05	3.68	1.58e-03	2.43	6.25e-03	2.40
	3	3.04e-06	3.95	2.29e-04	2.78	8.87e-04	2.82
	4	1.88e-07	4.02	3.02e-05	2.92	1.15e-04	2.95
	5	1.15e-08	4.02	3.85e-06	2.97	1.45e-05	2.99
	6	7.13e-10	4.02	4.85e-07	2.99	1.81e-06	3.00
	7	4.43e-11	4.01	6.08e-08	3.00	2.27e-07	3.00
3	1	8.07e-05	-	2.08e-03	-	9.47e-03	-
	2	3.30e-06	4.61	2.01e-04	3.37	8.89e-04	3.41
	3	1.18e-07	4.81	1.49e-05	3.75	6.38e-05	3.80
	4	3.74e-09	4.98	9.87e-07	3.92	4.15e-06	3.94
	5	1.15e-10	5.02	6.29e-08	3.97	2.62e-07	3.99
	6	3.55e-12	5.02	3.96e-09	3.99	1.64e-08	4.00
	7	1.10e-13	5.01	2.48e-10	4.00	1.03e-09	4.00

analysis of [9] where it is shown that suboptimal convergence (for the flux variable) is to be expected when the *total* advective-diffusive flux is treated as an independent variable instead of the sole diffusive flux. As a matter of fact, numerical tests carried out on reaction-diffusion problems reveal that the optimal convergence rate $O(h^{k+2-\delta_{k,1}})$ is recovered also for the BDM formulation.

We also see that, for $k \geq 0$, the error $\|\boldsymbol{q} - \boldsymbol{q}_h^*\|_{L^2(\Omega_h;\mathbf{c})}$ converges with the optimal order of $k+1$ for the LDG-H scheme of degree k and the RT scheme of the same degree, the accuracy of the RT method being slightly better most probably because of the higher dimension of the local finite element space. Again, a degradation of the order of convergence to k is visible for the BDM method of degree k, $k \geq 1$.

Concerning the L^2 error of the divergence, we see that the LDG-H and the RT discretization have orders of convergence higher by one order than those of the BDM method.

To have an idea of the relative efficiency of the method under consideration, we plot in Figures 3.3 and 3.4 the errors as a function of the computational complexity of the method as defined in the previous subsection. In the plots of Figure 3.3, we put together the approximations given by the different methods using the same polynomial degree for the unknown λ_h. We see that, as expected, the CG and BDM methods are outperformed by the RT and LDG-H methods which give similar answers, the RT method being slightly better.

Next, we compare the approximations given by the different methods that converge with the same order. Thus, in each of the plots of Figure 3.4, we compare the approximations given by the CG and BDM methods of degree $k+1$ with those of the LDG-H and RT methods of degree k. We see that all of these methods behave essentially in the same manner, with the BDM method outperforming all of the other methods.

Two important conclusions can be drawn from the several tests and comparisons discussed in this section. The first conclusion is that hybridization is the key for a highly efficient implementation of the LDG formulation. The second conclusion is that the LDG-H method has a computational cost completely comparable with that of a standard displacement-based finite element scheme of the same order. These two conclusions provide a strongly encouraging motivation towards a systematic use of the novel formulation in a wide variety of problems arising in computational fluid-mechanics, where flux conservation and self-equilibrium, as well as displacement superconvergence, are desirable, and often even mandatory, requirements.

3.3. Performance in the convection-dominated regime. In this section we present a preliminary assessment of the performance of the LDG-H method in the *convection-dominated* regime. To start, we assume that the problem coefficients ε, $\boldsymbol{\beta}$, and r are constant over Ω. Then, we define the velocity $\boldsymbol{v} := -\boldsymbol{\beta}$ and, as customary with the approximation of hyperbolic problems using the discontinuous Galerkin (DG) method, we define for each simplex $K \in \Omega_h$ the *inflow* and *outflow* boundaries of K as $\partial K^{in} := \{\boldsymbol{x} \in \partial K,\, \boldsymbol{v} \cdot \boldsymbol{n} < 0\}$ and $\partial K^{out} := \{\boldsymbol{x} \in \partial K,\, \boldsymbol{v} \cdot \boldsymbol{n} \geq 0\}$, respectively.

The stabilization parameter that we propose to use to deal with the case where the problem (1.1) is in the convection-dominated regime is defined as

$$\tau = \tau_{ell} + \tau_{hyp}, \tag{3.1}$$

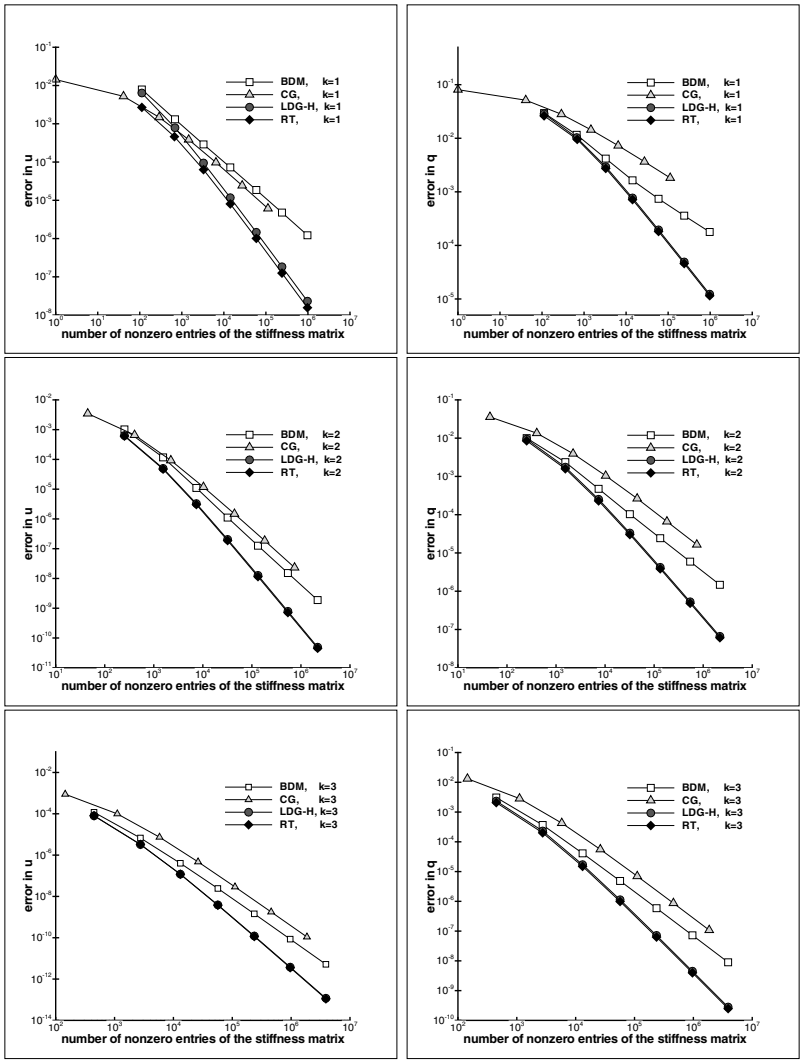

FIG. 3.3. *Comparison of the convergence of the L^2-error in u (left) and in $\boldsymbol{q}$ (right) for different methods. The postprocessed solution u_h^* was taken as the approximation for the BDM, LDG-H, and RT methods. The postprocessed solution $\boldsymbol{q}_h^\star$ was taken as the approximation for the LDG-H method.*

where

(3.2)
$$\tau_{ell} = \begin{cases} \dfrac{\varepsilon}{L} & \text{on } e_\tau^K, \\ 0 & \text{otherwise,} \end{cases} \qquad \tau_{hyp} = \begin{cases} 0 & \text{on } \partial K^{out}, \\ |\boldsymbol{\beta}\cdot\boldsymbol{n}| & \text{otherwise,} \end{cases}$$

where L is a characteristic length of the problem (for example, the diameter of Ω or the edge length $|e_\tau^K|$).

Notice that, in terms of the parameters τ_K and $\overline{\tau}_K$ introduced in (2.10), relation (3.1) leads to various possibilities depending on the choice of e_τ^K, with possible forms of τ^K and $\overline{\tau}_K$ being ε/L, $|\boldsymbol{\beta}\cdot\boldsymbol{n}|$, $\varepsilon/L(1+|\boldsymbol{\beta}\cdot\boldsymbol{n}|L/\varepsilon)$, and, for the sole $\overline{\tau}_K$, 0. As a consequence, $\overline{\tau}_K$ is constant with respect to h, while τ_K is either proportional

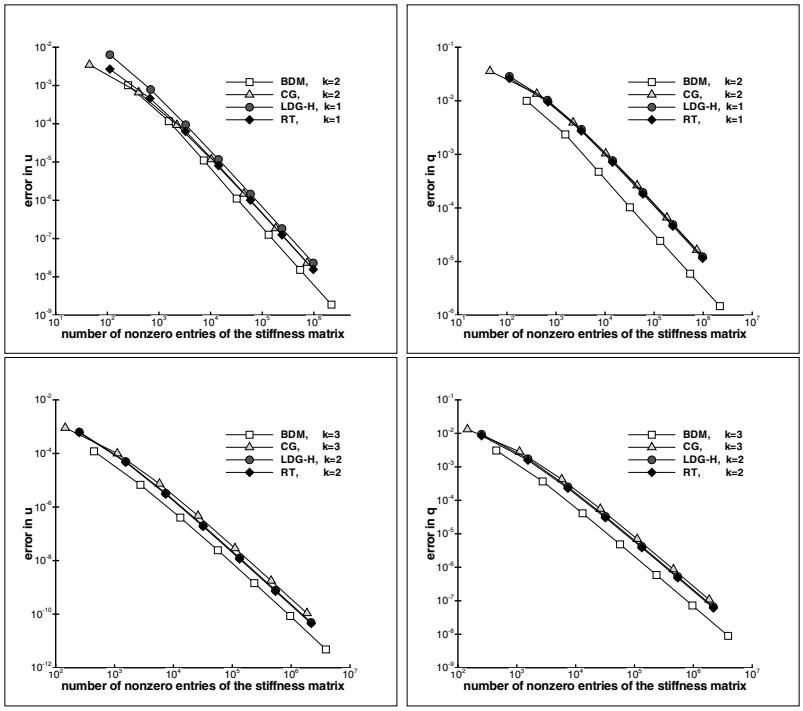

FIG. 3.4. *Comparison of the convergence of the L^2-error in u (left) and in $\boldsymbol{q}$ (right) for different methods. Only approximations converging at the same order are compared. The postprocessed solution u_h^* was taken as the approximation for the BDM, LDG-H, and RT methods. The postprocessed solution $\boldsymbol{q}_h^\star$ was taken as the approximation for the LDG-H method.*

to h^{-1}, for $L = |e_\tau^K|$, or constant, for other choices of L. In the vanishing viscosity limit $\varepsilon \to 0^+$, the LDG-H formulation using the upwind stabilization parameters (3.1) tends to reproduce exactly the DG approximation, which provides the desired upwinding behavior to the method. To prove this, let us start to observe that (2.2a) gives $\boldsymbol{q}_h = -\boldsymbol{\beta} u_h$ for each $K \in \Omega_h$, so that, replacing this quantity into (2.2b), with $\tau_{ell} \to 0^+$, we get for all $\omega \in W(K)$

$$\int_{\partial K} \widehat{\boldsymbol{q}}_h \cdot \boldsymbol{n}\, \omega \, d\gamma = \begin{cases} -\displaystyle\int_{\partial K^{out}} \boldsymbol{\beta} u_h^K \cdot \boldsymbol{n} \omega \, d\gamma & \text{on } \partial K^{out}, \\ -\displaystyle\int_{\partial K^{in}} \boldsymbol{\beta} \lambda_h \cdot \boldsymbol{n} \omega \, d\gamma & \text{on } \partial K^{in}. \end{cases} \tag{3.3}$$

Enforcing the conservativity condition (2.2c) on each face $e \in \mathscr{E}_h$, and denoting by K_e^{upstr} and K_e^{down} the two neighboring mesh simplexes located *upstream* and *downstream* with respect to the velocity $\boldsymbol{v}$, we get for all $\mu \in M(e)$

$$0 = \int_e [\![\widehat{\boldsymbol{q}}_h \cdot \boldsymbol{n}]\!] \mu \, d\gamma = \int_e (-\boldsymbol{\beta} \lambda_h \cdot \boldsymbol{n}^{down} - \boldsymbol{\beta} u_h^{K_e^{upstr}} \cdot \boldsymbol{n}^{upstr}) \mu \, d\gamma$$

from which it follows that

$$\lambda_h|_e = u_h^{K_e^{upstr}}|_e, \qquad e \in \mathscr{E}_h,$$

TABLE 3.7
Convergence history for the LDG-H methods in the convection-dominated regime. The errors are computed in $\widetilde{\Omega} = (0, 0.9)^2 \subset \Omega$.

		$\|u - u_h\|_{L^2(\widetilde{\Omega}_h)}$		$\|\boldsymbol{q} - \boldsymbol{q}_h^\star\|_{L^2(\widetilde{\Omega}_h;\mathbf{c})}$		$\|\nabla \cdot (\boldsymbol{q} - \boldsymbol{q}_h^\star)\|_{L^2(\widetilde{\Omega}_h)}$	
k	l	error	order	error	order	error	order
0	1	5.78e-02	-	8.95e-00	-	9.76e-02	-
	2	3.33e-02	0.80	6.48e-00	0.46	5.36e-02	0.86
	3	1.99e-02	0.74	3.53e-00	0.88	2.72e-02	0.98
	4	9.77e-03	1.03	1.83e-00	0.95	1.32e-02	1.04
	5	4.93e-03	0.99	9.65e-01	0.92	6.59e-03	1.01
	6	2.53e-03	0.96	4.99e-01	0.95	3.32e-03	0.99
	7	1.28e-03	0.98	2.51e-01	0.99	1.66e-03	1.00
	8	6.33e-04	1.02	1.25e-01	1.00	8.29e-04	1.00
1	1	1.22e-02	-	2.57e-00	-		
	2	3.54e-03	1.79	7.13e-01	1.85		
	3	9.09e-04	1.96	2.01e-01	1.83		
	4	2.14e-04	2.09	4.92e-02	2.03		
	5	5.41e-05	1.98	1.23e-02	2.00		
	6	1.35e-05	2.00	3.07e-03	2.00		
	7	3.34e-06	2.02	7.62e-04	2.01		
	8	8.13e-07	2.04	1.86e-04	2.03		

i.e., the value of the hybrid variable λ_h on each face of the triangulation equals the *upstream* trace of the discrete solution u_h in the interior of each simplex. Replacing the obtained expression for $\lambda_h|_e$ into $(3.3)_2$, we immediately see that the boundary term $\int_{\partial K} \widehat{\boldsymbol{q}}_h \cdot \boldsymbol{n}\, \omega\, d\gamma$ computed by the LDG-H formulation with the upwinding-like choice (3.1) of the stabilization parameter τ coincides with the corresponding term in the standard DG approximation of a linear hyperbolic problem.

The above result establishes a precise link between mixed-hybridized formulations and DG methods, and considerably widens the range of applicability of the LDG-H method to the treatment of parabolic-hyperbolic problems in any fluid-dynamical regime.

3.3.1. A convection-dominated problem with homogeneous boundary conditions. In the first example, the problem data are $\Omega = [0, 1]^2$, $\varepsilon = 10^{-4}$, $\boldsymbol{\beta} = [-1, -1]^T$, $r = 0$, and $f = 2(x\eta_1(x) + y\eta_1(y)) - (x + y)\eta_1(x)\eta_1(y)$, with the boundary condition $u = 0$ on $\partial\Omega$. Table 3.7 summarizes the convergence history of the upwind-stabilized formulation for the cases $k = 0$ and $k = 1$, the errors being computed on the reduced domain $\widetilde{\Omega} = (0, 0.9)^2 \subset \Omega$ to exclude the unresolved boundary layer (see [16]). This convergence analysis does not consider $k \geq 2$ since in this case the solution $u_{ex} = xy$ of the reduced problem (corresponding to taking $\varepsilon = 0$) belongs to the finite element space and the errors are negligible even for a very coarse choice of the grid size. For the same reason, the error in the divergence of $\boldsymbol{q}^\star$ for $k = 1$ is also omitted. It can be seen that the theoretically expected convergence rates are attained. For these tests we set $L = |e_\tau^K|$ in (3.2). In Figure 3.5, we address the issue of the robustness of the method when increasingly small diffusivity is considered. Here, we set the grid size equal to $h = 1/64$, progressively reduce ε from 0.5 to 10^{-9}, and compare the upwind stabilized LDG-H method with $L = |e_\tau^K|$ and the plain version of the method with $\tau = 1/h$, $\overline{\tau} = 0$. The L^2 errors computed on the domain $\widetilde{\widetilde{\Omega}} = (0, 0.99)^2 \subset \Omega$ are shown the scalar variable u_h; analogous results are obtained for $\boldsymbol{q}_h$ and $\boldsymbol{q}_k^\star$ which we do not report here. For large values of ε the boundary layer can be captured by the computational grid, and the upwind-stabilized method yields a solution similar to that of the plain method. For smaller values of ε, such that the boundary layer cannot be resolved on the chosen grid, the solution of the plain version of the method is affected by spurious oscillations, resulting in large errors, while the upwind-stabilized method converges to the solution of the reduced problem. Figure 3.6 shows the quantities u_h and λ_h in the case $k = 3$, computed by the LDG-

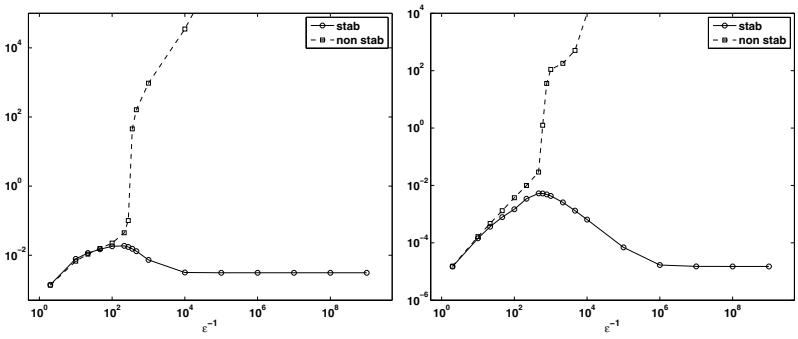

FIG. 3.5. *Error* $\|u - u_h\|_{L^2(\widetilde{\Omega}_h)}$ *for* $k = 0$ *(left) and* $k = 1$ *(right) for* ε *varying from* 0.5 *to* 10^{-9} *for the upwind-stabilized and the plain version of the LDG-H method.*

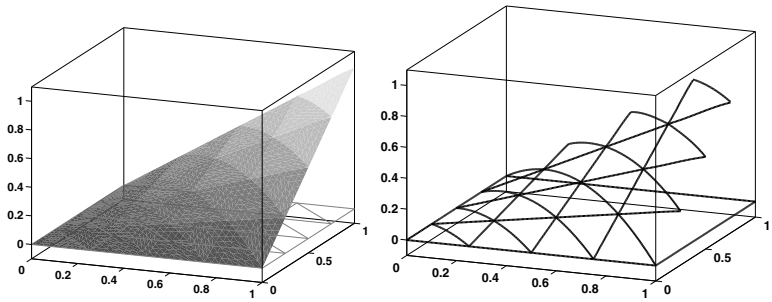

FIG. 3.6. *Approximate solution in the case* $k = 3$. *Left:* u_h, *right:* λ_h.

H method with τ as in (3.1) on a rather coarse triangulation with average grid size $h = 1/4$. The complete absence of spurious oscillations for u_h at the outflow boundary layer clearly indicates the ability of the upwinded LDG-H method in dealing with the almost hyperbolic nature of the problem. Similar results can be obtained for other values of the polynomial degree k, while the use of the plain (nonupwinded) LDG-H formulation (i.e., with $\tau = \tau_{ell}$) would provide a completely meaningless solution. We investigate in Figure 3.7 the dependence of the condition number `cond(A)` of the stiffness matrix A on the value of the diffusion coefficient ε. Figure 3.7 (left) shows `cond(A)` for the stabilized LDG-H method as a function of increasing values of ε^{-1} for $k = 0, \ldots 3$, and for a given mesh size $h = 1/32$. The obtained results indicate that, for each considered value of k, the condition number becomes *independent* of ε^{-1} as the problem enters the convective-dominated regime, with a mild increase of `cond(A)` as k increases. Figure 3.7 (right) compares the stabilized version of the LDG-H method with the CG method for $k = 1$ and the plain versions of the LDG-H and RT methods for $k = 0$. For the CG method, a finer grid with $h = 1/94$ is adopted to have a similar number of nonzero entries in the matrix. Results clearly demonstrate the effectiveness of the upwinding choice of τ, as `cond(A)` immediately deteriorates in highly convective-dominated regimes for the nonstabilized formulations.

3.3.2. The Smith and Hutton benchmark test case. In the second example, we study the well-known Smith and Hutton benchmark test case [18]. The diffusion coefficient is $\varepsilon = 10^{-4}$, $r = f = 0$, while the (divergence-free) advective field is $\boldsymbol{\beta} = (2y(1 - x^2), -2x(1 - y^2))^T$. The computational domain is $\Omega = [-1, 1] \times [0, 1]$,

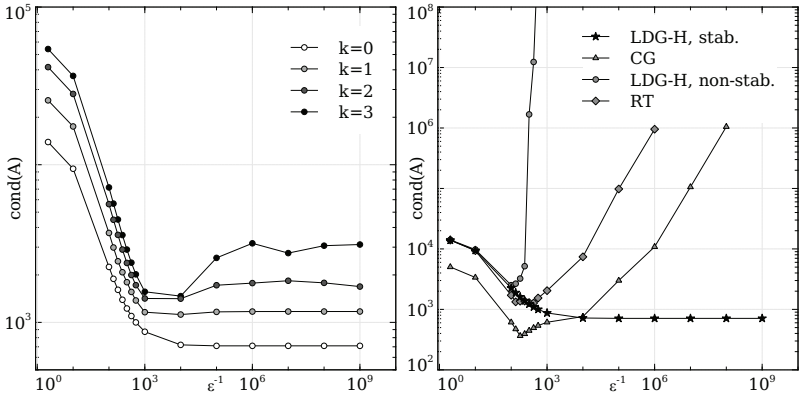

FIG. 3.7. *Condition number for the LDG-H method as a function of* ε^{-1} *in the cases* $k = 0, 1, 2, 3$ *(left); comparison with the nonstabilized LDG-H and RT methods of degree zero and the nonstabilized CG method of degree* 1 *(right).*

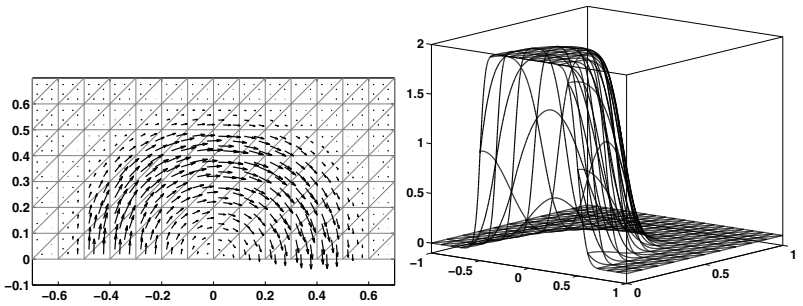

FIG. 3.8. *Approximate solution* $\boldsymbol{q}_h$ *(left) and* λ_h *(right) in the case* $k = 3$.

and a uniform triangulation with 40 subdivisions along the x axis and 20 subdivisions along the y axis is used. The boundary conditions at the inflow boundary $y = 0$, $x \in [-1, 0]$ are such that the solution u almost exhibits a discontinuity at $x = -0.5$. This discontinuity is then transported by the advective field in a clockwise rotation and exits out of the domain at the outflow boundary located at $y = 0$, $x \in [0, 1]$.

Figure 3.8 shows the quantities $\boldsymbol{q}_h$ and λ_h computed by the upwind-stabilized LDG-H method in the case $k = 3$. Results show that a very accurate and numerically stable solution is obtained, whereas severe spurious oscillations would arise if the plain LDG-H formulation were adopted.

4. Concluding remarks. In this article, we have introduced a novel class of discontinuous Galerkin methods, called LDG-H methods, for convection-diffusion-reaction problems. Extensive numerical experiments demonstrate that the novel proposed formulation is competitive with more standard displacement-based approaches, as far as the balance between computational effort and accuracy is concerned, while computing at the same time a conservative and superconvergent approximation of the flux variable and a superconvergent approximation of the scalar variable, as with standard mixed finite element schemes for elliptic problems. In the convection-dominated regime, a proper choice of the stabilization parameter τ allows one to recover a numerical scheme which is very close to classical DG approximations of hyperbolic problems.

This turns out to be particularly effective in the highly accurate and stable treatment of the sharp boundary and internal layers arising in the exact solution, and considerably widens the range of applicability of the novel method to the treatment of parabolic-hyperbolic problems in any fluid-dynamical regime.

A complete theoretical analysis of the well-posedness and a priori error estimates of the proposed LDG-H method will be presented in a forthcoming paper.

Acknowledgments. The authors wish to express their heartfelt gratitude to the late Prof. Fausto Saleri, who suggested we carry out an extensive comparison between the LDG-H and the standard CG formulations, after a talk given in May 2007 by the first author during his visit at Dipartimento di Matematica "F. Brioschi," Politecnico di Milano.

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 (2002), pp. 1749–1779.

[2] F. Brezzi, J. Douglas, Jr., and L. D. Marini, *Two families of mixed finite elements for second order elliptic problems*, Numer. Math., 47 (1985), pp. 217–235.

[3] A. N. Brooks and T. J. R. Hughes, *Streamline upwind-Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations*, Comput. Methods Appl. Mech. Engrg., 32 (1982), pp. 199–259.

[4] G. Chavent and J. Jaffré, *Mathematical Models and Finite Elements for Reservoir Simulation. Single Phase, Multiphase, and Multicomponent Flows through Porous Media*, Stud. Math. Appl. 17, North–Holland, Amsterdam, 1986.

[5] B. Cockburn, B. Dong, and J. Guzmán, *A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems*, Math. Comp., 77 (2008), pp. 1887–1916.

[6] 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 (2009), pp. 1319–1365.

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

[8] J. D. Cole, *On a quasilinear parabolic equation occurring in aerodynamics*, Quart. Appl. Math., 9 (1951), pp. 225–236.

[9] A. Demlow, *Suboptimal and optimal convergence in mixed finite element methods*, SIAM J. Numer. Anal., 39 (2002), pp. 1938–1953.

[10] J. Douglas, Jr. and J. E. Roberts, *Global estimates for mixed methods for second order elliptic equations*, Math. Comp., 44 (1985), pp. 39–52.

[11] L. P. Franca, S. L. Frey, and T. J. R. Hughes, *Stabilized finite element methods:* I. *Application to the advective-diffusive model*, Comput. Methods Appl. Mech. Engrg., 95 (1992), pp. 253–276.

[12] E. Hopf, *The partial differential equation* $u_t + uu_x = u_{xx}$, Comm. Pure Appl. Math., 3 (1950), pp. 201–230.

[13] J. W. Jerome, *Analysis of Charge Transport. A Mathematical Study of Semi-conductor Devices*, Springer-Verlag, Berlin, 1996.

[14] J. J. H. Miller and S. Wang, *A new nonconforming Petrov–Galerkin finite-element method with triangular elements for a singularly perturbed advection-diffusion problem*, IMA J. Numer. Anal., 14 (1994), pp. 257–276.

[15] P. A. Raviart and J. M. Thomas, *A mixed finite element method for second order elliptic problems*, Mathematical Aspects of Finite Element Methods, Lecture Notes in Math. 606, I. Galligani and E. Magenes, eds., Springer, Berlin, 1977, pp. 292–315.

[16] H.-G. Roos, M. Stynes, and L. Tobiska, *Numerical Methods for Singularly Perturbed Differential Equations*, Springer-Verlag, Berlin, 1996.

[17] I. Rubinstein, *Electro-Diffusion of Ions*, SIAM Stud. Appl. Math. 11, SIAM, Philadelphia, 1990.

[18] R. M. Smith and A. G. Hutton, *The numerical treatment of advection: A performance comparison of current methods*, Numer. Heat Transfer, Part A: Applications, 5 (1982), pp. 439–461.