UNIVERSIDAD DE CONCEPCIÓN

Centro de Investigación en Ingeniería Matemática (CI^{2}MA)

A high order HDG method for Stokes flow in curved domains

Manuel Solano, Felipe Vargas

PREPRINT 2016-12

SERIE DE PRE-PUBLICACIONES

A HIGH ORDER HDG METHOD FOR STOKES FLOW IN CURVED DOMAINS

MANUEL SOLANO * AND FELIPE VARGAS *

Abstract. We propose and analyze a high order hybridizable discontinuous Galerkin (HDG) method for the Stokes equations in a curved domain. It is based on approximating the domain by a polyhedral computational subdomain where an HDG solution is computed. To obtain a high order approximation of the Dirichlet boundary data in the computational domain, we employ a transferring technique based on integrating the approximation of the gradient. In addition, we first seek for a discrete pressure having zero-mean in the computational domain and then the zero-mean condition in the entire domain is recovered by a post-process that involves an extrapolation of the discrete pressure. We prove that the method provides optimal order of convergence for the approximations of the pressure, the velocity and its gradient. We present numerical experiments validating the method.

Key words. curved domains, unfitted methods, discontinuous Galerkin, Stokes flow

AMS subject classifications. 65N30, 65N12, 65N15

1. Introduction. We present a high order hybridizable discontinuous Galerkin (HDG) method to solve numerically the Stokes equations of an incompressible fluid flow occupying a region $\Omega \subset \mathbb{R}^d$ not necessarily polygonal ($d = 2$) or polyhedral ($d = 3$) with boundary $\Gamma := \partial\Omega$ compact and Lipschitz. More precisely, denoting by $\boldsymbol{u}$ the velocity of the fluid, p the pressure, $\nu > 0$ a constant viscosity, $\boldsymbol{f} \in [L^2(\Omega)]^d$ a source term and $\boldsymbol{g} \in [L^2(\Gamma)]^d$ the prescribed velocity at the boundary satisfying the compatibility condition $\displaystyle\int_\Gamma \boldsymbol{g} \cdot \boldsymbol{n} = 0$ ($\boldsymbol{n}$ is the outward unit normal to Ω), the equations to solve are

$$\mathrm{L} - \nabla \boldsymbol{u} = 0 \quad \text{in } \Omega, \tag{1.1a}$$

$$-\nabla \cdot (\nu \mathrm{L}) + \nabla p = \boldsymbol{f} \quad \text{in } \Omega, \tag{1.1b}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega, \tag{1.1c}$$

$$\boldsymbol{u} = \boldsymbol{g} \quad \text{on } \Gamma, \tag{1.1d}$$

$$\int_\Omega p = 0. \tag{1.1e}$$

The main idea is to approximate Ω by a polyhedral computational domain where the boundary data $\boldsymbol{g}$ is properly transferred to the computational boundary. We follow the approach proposed by [9] and analyzed in [8] that consists of integrating the approximation of the gradient along *transferring paths* connecting the computational boundary and Γ. This technique allows us to obtain a high order approximation of the boundary data in the computational domain which leads to a high order accuracy of the discrete solution. More precisely, if polynomials of degree k are used to construct the local discrete spaces, the error in the variables L, $\boldsymbol{u}$ and p, measured in L^2-norm, will be of order h^{k+1}, where h is the meshsize. This technique has been recently applied to several problems and shown to perform properly. However, to the best of our knowledge, this is the first work that applies it to the Stokes flow problem where the main difference with previous work is the treatment of the approximated pressure in the computational domain. One of the first ideas based on this transferring

*Departamento de Ingeniería Matemática and Centro de Investigación en Ingeniería Matemática (CI^{2}MA), Universidad de Concepción, Concepción, Chile.

technique was introduced by [7] for the one-dimensional case and then extended to higher dimensions for pure diffusion ([8, 9]) and convection-diffusion [10] equations. In all these work, Dirichlet boundary data was considered and the mesh does not fit the domain. It was shown that high order accuracy is obtained if the distance between the computational domain and the boundary is of only order h. Recently, a similar approach was proposed and studied numerically for Neumann boundary conditions and elliptic interface problems [28]. There, the Neumann data cannot be treated in the same way as we treat the Dirichlet data and the computational boundary/interface in [28] must be order h^2 away from the true boundary/interface in order to obtain optimal results. Roughly speaking, the Neumann data in [28] is imposed by extrapolating the discrete gradient, whereas the Dirichlet data is imposed by performing a line integration of the discrete gradient. This integral is over a segment of length proportional to h, hence it provides a power of h that allows to overcome the fact that the distance between Γ and the computational boundary is of only of order h.

Let us briefly discuss some of the literature related to numerical methods for differential equations involving curved boundaries or interfaces. In general, these methods can be classified as *fitted* or *unfitted*. In *fitted* methods, the discretization of the domain resolves the boundary/interface, up to some degree of accuracy. For instance, this is the case of isoparametric finite elements [20] where the mapping from the reference element to a physical element is a polynomial whose degree is the same as the polynomial degree of the finite element space. In this direction, [1] considers a finite element approximation for smooth elliptic interfaces where optimal convergence rates are obtained if the mesh isoparemetrically fits the interface. For piecewise linear approximation of a Dirichlet boundary value problem, [3] approximates the boundary data at the computational boundary Γ_h by transferring the boundary data g in a *natural way*. That is, for $x \in \Gamma_h$ and a suitably defined $\bar{x} \in \Gamma$ such that $|x - \bar{x}|$ is of order h^2, then the computational boundary data, $\tilde{g}$, is set as $\tilde{g}(x) := g(\bar{x})$. The same idea was extended later to interface problems [4]. As we will see in (2.2), in some sense our technique can be seen as a generalization where we add an additional term to the *natural way* of transferring. For a high order H-div-conforming method to solve elliptic interface problems we refer to [21]. There, the curved interface is locally interpolated by splines and optimal convergence is obtained for a suitable chosen spline degree. Fitted methods have been also applied to control problems in curved domains [13], where only polynomials of degree one are considered. In general, one of the main advantages of fitted methods is that the prescribed data at the boundary/interface can be easily imposed. However, the construction of the meshes might be difficult, especially in complicated geometries. On the other hand, the attractive feature of unfitted methods is that the mesh is not adjusted to the domain and even Cartesian grids can be considered. However, it is not straightforward to develop a high order unfitted method, mainly because of imposition of the boundary/interface data away from the true boundary/interface. In the context of finite differences, one of the most popular unfitted method is the Immersed Boundary (IB) method introduced by [25] in 1972. For the two dimensional case, [22] showed that it is a first order method for the velocity of a Stokes flow and second order accuracy can be achieved away from the interface. Later, in the same direction, LeVeque and Li developed the Immersed Interface method [19], which is second order accurate. In the context of finite element method, [2] proposed an unfitted method based on Nitsche's approach [24] with the polygonal domain approximation method [29]. There, the numerical scheme is posed

in a polygonal domain and a correction term is added to Nitsche's bilinear form in order to obtain high order accuracy. Recently, a high-order finite element method was proposed to solve elliptic interface problems [16]. It is also based on a term involving a piecewise polynomial correction function which is added to the functional at the right hand side. This correction function is suitable constructed a priori using the information provided by the equation and the transmission conditions at the interface. Similarly, an HDG method based also on pre–computing a correction function was proposed and analyzed in [14]. Finally, we would like to mention an alternative approach, the Composite Finite Elements [26, 27], that deals with complicated domains involving rough boundaries or small holes inside. However, to the best of our knowledge, it has not been developed for high order approximations.

Our aim is to develop a method that combines the flexibility of the mesh construction of unfitted methods with the high order accuracy of fitted methods. In fact, the main advantage of our technique is that the computational domain is easy to build and there is no need to adjust it to the actual domain. Moreover, the way we transfer the Dirichlet data allows us to consider high order polynomials in the discrete spaces and obtain high order accuracy. On the other hand, one of the drawbacks of our method compared to others is that it relies on two aspects of the PDE: (1) $\nabla \boldsymbol{u}$ must be part of the equation and (2) $\boldsymbol{u} = \boldsymbol{g}$ at Γ. In addition, the resulting global matrix is not symmetric in our case. Symmetrizing the method is subject of future work. We believe our technique is, in principle, independent on the numerical method, as long the gradient is properly approximated and the PDE has the aforementioned characteristics.

In this paper we consider a standard HDG method ([5, 23, 6]). The rest of the paper is organized as follows. We first describe the construction of the computational domain and transferring paths in Section 2. The method is introduced in Section 3 and its well-posedness is analyzed. In Section 4 the error estimates are stated and Section 5 is devoted to their proofs. In Section 6 we propose an approximation of the solution outside the computational domain and provide the corresponding error estimates. Section 7 shows numerical experiments validating these results. We end with concluding remarks.

2. Computational domain and transferring paths. In this section we introduce notation associated to the computational domain and to the family of paths that will allow us to transfer the boundary data from Γ to the computational boundary Γ_h. Moreover, we establish a set of assumptions under which our analysis holds.

Given $h > 0$, we denote by D_h an open polyhedral computational domain, with boundary Γ_h, meshed by a triangulation T_h of meshsize h, consisting of simplices K that are uniformly shape-regular, that is,

(D.1) there exists $\gamma > 0$, independent of h, such that $h_K \leq \gamma \rho_K$,

where ρ_K is the radius of the largest ball contained in K and h_K is the diameter of K. We assume

(D.2) $\max_{K \in \mathsf{T}_h} h_K \leq h$.

For a simplex K, we denote its outward unit normal by $\boldsymbol{n}_K$, writing $\boldsymbol{n}$ instead of $\boldsymbol{n}_K$ when there is no confusion. Similarly, for a face e, we write $\boldsymbol{n}$ instead of $\boldsymbol{n}_e$ to refer to its normal vector. We also consider, by simplicity, that the triangulation does not have hanging nodes.

In addition, we suppose that

(D.3) $\overline{\mathsf{D}_h} \subset \overline{\Omega}$.

The set of faces and boundary faces of T_h are denoted by $\mathcal{E}_h$ and $\mathcal{E}_h^\partial$, respectively, and we define the non-meshed region $\mathsf{D}_h^e := \Omega \setminus \overline{\mathsf{D}_h}$.

On the other hand, let $\mathcal{P}_r(K)$ denote the space of polynomials of total degree at most r defined on K, $\boldsymbol{\mathcal{P}}_r(K) := [\mathcal{P}_r(K)]^d$ and $\mathrm{P}_r(K) := [\mathcal{P}_r(K)]^{d\times d}$. Given a region $D \subset \mathbb{R}^d$, we denote by $(\cdot,\cdot)_D$ and $\langle\cdot,\cdot\rangle_{\partial D}$ the $L^2(D)$ and $L^2(\partial D)$ inner products, respectively. The L^2-norms over D and ∂D will be denoted by $\|\cdot\|_D$ and $\|\cdot\|_{\partial D}$.

For each scalar-valued function η and ζ, we define

$$(\eta,\zeta)_{\mathsf{T}_h} := \sum_{K\in\mathsf{T}_h} (\eta,\zeta)_K \quad \text{and} \quad \langle\eta,\zeta\rangle_{\partial\mathsf{T}_h} := \sum_{K\in\mathsf{T}_h} \langle\eta,\zeta\rangle_{\partial K}.$$

Vector-valued functions are boldfaced and, for $\boldsymbol{\eta}$ and $\boldsymbol{\zeta}$, we write

$$(\boldsymbol{\eta},\boldsymbol{\zeta})_{\mathsf{T}_h} := \sum_{i=1}^d (\eta_i,\zeta_i)_{\mathsf{T}_h} \quad \text{and} \quad \langle\boldsymbol{\eta},\boldsymbol{\zeta}\rangle_{\partial\mathsf{T}_h} := \sum_{i=1}^d \langle\eta_i,\zeta_i\rangle_{\partial\mathsf{T}_h}.$$

Tensor-valued functions are in Roman letters and, for N and Z, we write

$$(\mathrm{N},\mathrm{Z})_{\mathsf{T}_h} := \sum_{i,j=1}^d (\mathrm{N}_{ij},\mathrm{Z}_{ij})_{\mathsf{T}_h} \quad \text{and} \quad \langle\mathrm{N},\mathrm{Z}\rangle_{\partial\mathsf{T}_h} := \sum_{i,j=1}^d \langle\mathrm{N}_{ij},\mathrm{Z}_{ij}\rangle_{\partial\mathsf{T}_h}.$$

We use the standard notation for Sobolev spaces and their associated norms and seminorms. Finally, to avoid proliferation of constants, we will write $a \lesssim b$ instead of $a \le Cb$, where C is a constant independent of h.

Now, since the problem will be solved in D_h, we must specify a suitable boundary data on the computational boundary Γ_h. To this end, we consider the idea proposed by [9] and transfer the boundary data $\boldsymbol{g}$ from Γ to Γ_h through transferring paths. That is, let $e \in \mathcal{E}_h^\partial$ with normal unit vector $\boldsymbol{n}$. For each $\boldsymbol{x} \in e$, we set $\bar{\boldsymbol{x}} \in \Gamma$ as the closest intersection between Γ and the ray of tangent vector $\boldsymbol{n}$ starting at $\boldsymbol{x}$. We denote by $\sigma_{\boldsymbol{n}}(\boldsymbol{x})$ the segment joining $\bar{\boldsymbol{x}}$ and $\boldsymbol{x}$, which is referred as *transferring path.* The length of this segment, $l(\boldsymbol{x}) := |\bar{\boldsymbol{x}} - \boldsymbol{x}|$, is assumed to satisfy

(D.4) $l(\boldsymbol{x}) \lesssim h$.

We notice that this assumption implies that $\mathrm{dist}(\Gamma_h,\Gamma) \lesssim h$. In addition, we observe that if $\boldsymbol{x}$ is a vertex in Γ_h, then it has associated two different transferring paths. In other words, the functions $\bar{\boldsymbol{x}} = \bar{\boldsymbol{x}}(\boldsymbol{x})$ and $l(\boldsymbol{x})$ are double-valued on the boundary vertices.

Let $e \in \mathcal{E}_h^\partial$ and K^e the element where it belongs. We define

$$K_{ext}^e := \{\boldsymbol{x} + s\boldsymbol{n} : 0 \le s \le l(\boldsymbol{x}),\, \boldsymbol{x} \in e\}$$

and assume that,

(D.5) for each point $\boldsymbol{x} \in e$, the intersection of the ray $\{\boldsymbol{x} + \eta\boldsymbol{n} : \eta > 0\}$ and Γ is unique.

We denote by $H_e^\perp$ the largest distance of a point in K_{ext}^e to the plane determined by e and $h_e^\perp$ be the distance between e and the vertex of K^e opposite to e, as Fig. 1 shows. We set $r_e = H_e^\perp/h_e^\perp$ and $R = \max_{e\in\mathcal{E}_h^\partial} r_e$. This ratio will be important in our estimates and assumptions, since it indicates how far is Γ from the computational

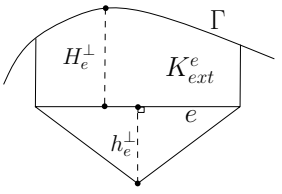

Fig. 1: Geometric definitions

boundary relative to the mesh size. If the domain were polygonal and the mesh fits its boundary, this ratio would be zero.

Through the rest of this manuscript, Assumptions (D) will be assumed valid, without mention it explicitly.

We are now in position to deduce the boundary data at Γ_h. Let $e \in \mathcal{E}_h^\partial$ with outward normal unit vector $\boldsymbol{n}$ and $\boldsymbol{x} \in e$. Integrating component-wise $\mathrm{L} = \nabla \boldsymbol{u}$ along $\sigma_{\boldsymbol{n}}(\boldsymbol{x})$ we obtain

$$u(\boldsymbol{x}) = u(\bar{\boldsymbol{x}}) - \int_0^{l(\boldsymbol{x})} \mathrm{L}(\boldsymbol{x} + \boldsymbol{n}s)\, \boldsymbol{n}\, ds.$$

Denoting by $\widetilde{\boldsymbol{g}}$ the trace of $\boldsymbol{u}$ at e and noticing that $\boldsymbol{u}(\bar{\boldsymbol{x}}) = \boldsymbol{g}(\bar{\boldsymbol{x}})$, from previous identity we get

$$\widetilde{\boldsymbol{g}}(\boldsymbol{x}) = \boldsymbol{g}(\bar{\boldsymbol{x}}) - \int_0^{l(\boldsymbol{x})} \mathrm{L}(\boldsymbol{x} + \boldsymbol{n}s)\, \boldsymbol{n}\, ds. \tag{2.1}$$

In other words, (2.1) provides an expression to obtain the boundary data at Γ_h in terms of the unknown L. We point out that, according to Lemma B.1 in [8], $\widetilde{\boldsymbol{g}}$ is well-defined if Γ is compact, Lipschitz and piecewise $\mathcal{C}^2$.

In order to construct a discrete approximation of $\widetilde{\boldsymbol{g}}$, denoted by $\widetilde{\boldsymbol{g}}_h$, we consider a *local extrapolation* of polynomials, i.e., given G a polynomial defined on K^e, with $e \in \mathcal{E}_h^\partial$, we extrapolate it from K^e to K_{ext}^e. Thus, the expression in (2.1) suggests the following approximation

$$\widetilde{\boldsymbol{g}}_h(\boldsymbol{x}) := \boldsymbol{g}(\bar{\boldsymbol{x}}) - \int_0^{l(\boldsymbol{x})} \mathrm{L}_h(\boldsymbol{x} + \boldsymbol{n}s)\boldsymbol{n}\, ds, \tag{2.2}$$

where L_h is the local extrapolation of the HDG approximation of L.

We end this section be defining the norms

$$\|\zeta\|_{\mathsf{D}_h^c, h^\perp} := \left\{ \sum_{e \in \mathcal{E}_h^\partial} h_e^\perp \|\zeta\|_{K_{ext}^e}^2 \right\}^{1/2}, \quad \|\boldsymbol{\mu}\|_{\partial \mathsf{T}_h, \alpha} := \left\{ \sum_{K \in \mathsf{T}_h} \langle \alpha_{\partial K} \zeta, \zeta \rangle_{\partial K} \right\}^{1/2},$$

$$\|\zeta\|_{\Gamma_h, \alpha} := \left\{ \sum_{e \in \mathcal{E}_h^\partial} \alpha_e \|\zeta\|_e^2 \right\}^{1/2}, \quad \text{and} \quad \|\boldsymbol{\mu}\|_h := \left\{ \sum_{K \in \mathsf{T}_h} h_K \langle \boldsymbol{\mu}, \boldsymbol{\mu} \rangle_{\partial K} \right\}^{1/2},$$

where $\alpha_e := \alpha|_e$ and $\alpha_{\partial K} := \alpha|_{\partial K}$.

3. The HDG method. We proceed now to introduce the numerical method on the computational domain D_h. First of all, we decompose

$$p = \overline{p}^{\mathsf{D}_h} + \widetilde{p}, \tag{3.1}$$

where $\overline{p}^{\mathsf{D}_h} := \dfrac{1}{|\mathsf{D}_h|}\displaystyle\int_{\mathsf{D}_h} p$ and $\widetilde{p} \in L^2_0(\mathsf{D}_h)$ ($L^2(\mathsf{D}_h)$-functions with zero mean in D_h). Since now the problem is posed in a polyhedral domain, we consider the standard HDG method ([5, 23]) that seeks an approximation $(\mathrm{L}_h, \boldsymbol{u}_h, \widetilde{p}_h, \widehat{\boldsymbol{u}}_h)$ of the solution $(\mathrm{L}, \boldsymbol{u}, \widetilde{p}, \boldsymbol{u}|_{\mathcal{E}_h})$ in the space $\mathrm{G}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h$ given by

$$\mathrm{G}_h = \{\mathrm{G} \in \mathrm{L}^2(\mathsf{T}_h) : \quad G|_K \in \mathrm{P}_k(K) \ \ \forall K \in \mathsf{T}_h\}, \tag{3.2a}$$

$$\boldsymbol{V}_h = \{\boldsymbol{v} \in \boldsymbol{L}^2(\mathsf{T}_h) : \quad \boldsymbol{v}|_K \in \boldsymbol{\mathcal{P}}_k(K) \ \ \forall K \in \mathsf{T}_h\}, \tag{3.2b}$$

$$P_h = \{q \in L^2(\mathsf{T}_h) : \quad q|_K \in \mathcal{P}_k(K) \ \ \forall K \in \mathsf{T}_h\}, \tag{3.2c}$$

$$\boldsymbol{M}_h = \{\boldsymbol{\mu} \in \boldsymbol{L}^2(\mathcal{E}_h) : \quad \boldsymbol{\mu}|_e \in \boldsymbol{\mathcal{P}}_k(e) \ \ \forall e \in \mathcal{E}_h\}. \tag{3.2d}$$

As we will see in Section 3.1, this approximation is the only solution of

$$(\mathrm{L}_h, \mathrm{G})_{\mathsf{T}_h} + (\boldsymbol{u}_h, \nabla \cdot \mathrm{G})_{\mathsf{T}_h} - \langle \widehat{\boldsymbol{u}}_h, \mathrm{G}\boldsymbol{n}\rangle_{\partial\mathsf{T}_h} = 0, \tag{3.3a}$$

$$(\nu \mathrm{L}_h, \nabla \boldsymbol{v})_{\mathsf{T}_h} - (\widetilde{p}_h, \nabla \cdot \boldsymbol{v})_{\mathsf{T}_h} - \langle \nu\widehat{\mathrm{L}}_h\boldsymbol{n} - \widehat{p}_h\boldsymbol{n}, \boldsymbol{v}\rangle_{\partial\mathsf{T}_h} = (\boldsymbol{f}, \boldsymbol{v})_{\mathsf{T}_h}, \tag{3.3b}$$

$$-(\boldsymbol{u}_h, \nabla q)_{\mathsf{T}_h} + \langle \widehat{\boldsymbol{u}}_h \cdot \boldsymbol{n}, q\rangle_{\partial\mathsf{T}_h} = 0, \tag{3.3c}$$

$$\langle \widehat{\boldsymbol{u}}_h, \boldsymbol{\mu}\rangle_{\Gamma_h} = \langle \widetilde{\boldsymbol{g}}_h, \boldsymbol{\mu}\rangle_{\Gamma_h}, \tag{3.3d}$$

$$\langle \nu\widehat{\mathrm{L}}_h\boldsymbol{n} - \widehat{p}_h\boldsymbol{n}, \boldsymbol{\mu}\rangle_{\partial\mathsf{T}_h\setminus\Gamma_h} = 0, \tag{3.3e}$$

$$(\widetilde{p}_h, 1)_{\mathsf{D}_h} = 0, \tag{3.3f}$$

for all $(\mathrm{G}, \boldsymbol{v}, q, \boldsymbol{\mu}) \in \mathrm{G}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h$, where we emphasize that $\widetilde{\boldsymbol{g}}_h$ depends on L_h (c.f. (2.2)),

$$\nu\widehat{\mathrm{L}}_h\boldsymbol{n} - \widehat{p}_h\boldsymbol{n} = \nu\mathrm{L}_h\boldsymbol{n} - \widetilde{p}_h\boldsymbol{n} - \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h) \qquad \text{on } \partial\mathsf{T}_h, \tag{3.3g}$$

and τ is a non-negative piecewise constant stabilization parameter defined on $\partial\mathsf{T}_h$. In order to simplify notation, we assume τ to be constant on $\partial\mathsf{T}_h$. The choice of τ is not crucial in our method. For other values of τ we refer to [5].

3.1. Well-posedness of the method. In this section we show that the discrete scheme (3.3) with the approximated boundary data (2.2) is well-posed. The steps that will lead to this statement will be very similar to those needed to obtain the error estimates in Section 5. First of all, we define an auxiliary function and a norm that will help to carry out several of the estimates in this work. Let $e \in \mathcal{E}_h^\partial$, a point $\boldsymbol{x}$ lying on e and a tensor-valued function G. We define the function

$$\delta_{\mathrm{G}}(\boldsymbol{x}) := \frac{1}{l(\boldsymbol{x})}\int_0^{l(\boldsymbol{x})} \big(\mathrm{G}(\boldsymbol{x} + s\boldsymbol{n}) - \mathrm{G}(\boldsymbol{x})\big)\boldsymbol{n}\ \mathrm{d}s. \tag{3.4}$$

If G is a polynomial, the estimates in Lemma 5.2 of [8] states that

$$\|l^{1/2}\delta_{\mathrm{G}}\|_e \le 3^{-1/2} r_e^{3/2} C_{ext}^e C_{inv}^e \|\mathrm{G}\|_{K^e}, \tag{3.5}$$

where

$$C_{ext}^e := \frac{1}{\sqrt{r_e}} \sup_{\mathrm{G}\in\mathrm{P}_k(K^e)\setminus\{0\}} \frac{\|\mathrm{G}\|_{K_{ext}^e}}{\|\mathrm{G}\|_{K^e}} \tag{3.6}$$

and

$$C^e_{inv} := h_e^{\perp} \sup_{\mathrm{G}\in \mathrm{P}_k(K^e)\setminus\{0\}} \frac{\|\partial_{\boldsymbol{n}}\mathrm{G}\|_e}{\|\mathrm{G}\|_{K^e}}. \tag{3.7}$$

The constants C^e_{ext} and C^e_{inv} do not depend on the meshsize, but depend on the polynomial degree and shape-regularity constant as can be deduced from Lemma A.2 in [8]. C^e_{ext} is what we call *extrapolation constant* because it allows us to bound the L^2-norm of the extrapolation of the polynomial G in terms of $\|\mathrm{G}\|_{K^e}$. Similarly, C^e_{inv} can be seen as the constant of an inverse-type estimate that allows us to control the L^2-norm of the directional derivative of the extrapolation of the polynomial G in terms of $\|\mathrm{G}\|_{K^e}$. Finally, we recall the discrete trace inequality (Lemma 1.46 in [15]): If φ is a scalar, vector or tensor-valued polynomial in K^e, then

$$\|\varphi\|_e \le C^e_{tr} h_e^{-1/2} \|\varphi\|_{K^e}, \tag{3.8}$$

where C^e_{tr} is independent of the meshsize but depends on the polynomial degree.

The next two lemmas will lead to the well-posedness of the method.

LEMMA 3.1. *Let* $(\mathrm{L}_h, \boldsymbol{u}_h, \widetilde{p}_h, \widehat{\boldsymbol{u}}_h) \in \mathrm{G}_h \times \boldsymbol{V}_h \times P_h \times \boldsymbol{M}_h$ *be the solution of* (3.3) *with the boundary data in* (2.2). *Let us suppose that* $\boldsymbol{f} = \boldsymbol{0}$ *and* $\boldsymbol{g} = \boldsymbol{0}$. *In addition, for every* $e \in \mathcal{E}_h^{\partial}$, *we assume that*

(A.1) $r_e^3 (C^e_{ext})^2 (C^e_{inv})^2 \le 1/8$ *and*

(A.2) $r_e h_e^{\perp} \tau \le 1/8$.

Then,

$$\|\mathrm{L}_h\|_{\mathsf{D}_h} + \|\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h\|_{\partial\mathsf{T}_h,\tau} + \|\widetilde{\boldsymbol{g}}_h\|_{\Gamma_h, l^{-1}} \le 2\sqrt{6}\nu^{-1}\|\widetilde{p}_h\|_{\Gamma_h, l}. \tag{3.9}$$

Before going through the proof, let us comment on these assumptions. If the domain were polygonal and the mesh fits its boundary, then $r_e = 0$ and all the assumptions are satisfied. In that case and we recover the standard HDG method [5]. On the other hand, the constants C^e_{ext}, C^e_{inv} and C^e_{tr} are proportional to $(k+1)^{8/3}$ as stated in Lemma A.2 of [8]. Then, if the ratio r_e is small enough, (A.1) and (A.2) hold. In other words, the larger is the polynomial degree, the smaller this ratio must be to guarantee well-posedness of the scheme.

Let us continue with the proof of Lemma 3.1.

Proof. By taking $\mathrm{G} = \nu\mathrm{L}_h$, $\boldsymbol{v} = \boldsymbol{u}_h$, $q = \widetilde{p}_h$ and $\boldsymbol{\mu} = \widehat{\boldsymbol{u}}_h$ in the first four equations of (3.3) and adding them up we obtain

$$\|\mathrm{L}_h\|^2_{\mathsf{D}_h} + \|\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h\|^2_{\partial\mathsf{T}_h,\tau} = \nu^{-1}\langle \nu\widehat{\mathrm{L}}_h\boldsymbol{n} - \widehat{p}_h\boldsymbol{n}, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h}. \tag{3.10}$$

Now, let $e \in \mathcal{E}_h^{\partial}$ and $\boldsymbol{x} \in e$. According to (2.2) and (3.4) we can write

$$l(\boldsymbol{x})^{-1}\widetilde{\boldsymbol{g}}_h(\boldsymbol{x}) = -\delta_{\mathrm{L}_h}(\boldsymbol{x}) - \mathrm{L}_h(\boldsymbol{x})\boldsymbol{n}.$$

From this inequality we obtain an expression for $\mathrm{L}_h(\boldsymbol{x})\boldsymbol{n}$ and, by (3.3g), we write:

$$\begin{aligned}
\nu^{-1}\langle \nu\widehat{\mathrm{L}}_h\boldsymbol{n} - \widehat{p}_h\boldsymbol{n}, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} &= \langle \mathrm{L}_h\boldsymbol{n}, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} - \nu^{-1}\langle \widetilde{p}_h\boldsymbol{n}, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} - \langle \tau(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} \\
&= -\langle l^{-1}\widetilde{\boldsymbol{g}}_h, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} - \langle \delta_{\mathrm{L}_h}, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} \\
&\quad -\nu^{-1}\langle \widetilde{p}_h\boldsymbol{n}, \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h} - \langle \tau(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \widetilde{\boldsymbol{g}}_h\rangle_{\Gamma_h}.
\end{aligned}$$

Replacing this in (3.10), using Cauchy-Schwarz and Young's inequalities; and moving to the left hand side the terms involving $\widetilde{\boldsymbol{g}}_h$, we get

$$\begin{aligned} &\|\mathrm{L}_h\|^2_{\mathsf{D}_h} + \|\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h\|^2_{\partial\mathsf{T}_h,\tau} + \|\widetilde{\boldsymbol{g}}_h\|^2_{\Gamma_h,l^{-1}} \\ &\qquad \le 4\|\delta_{\mathrm{L}_h}\|^2_{\Gamma_h,l} + \frac{4}{\nu^2}\|\widetilde{p}_h\|^2_{\Gamma_h,l} + 4\|\tau(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h)\|^2_{\Gamma_h,l}. \end{aligned} \tag{3.11}$$

By (3.8), (3.5), assumptions $(A.1)$ and $(A.2)$, the fact that $l(\boldsymbol{x}) \le H_e^{\perp} = r_e h_e^{\perp}$ and $h_e^{\perp} \le \gamma h_e$, we obtain

$$\|\delta_{\mathrm{L}_h}\|^2_{\Gamma_h,l} \le \frac{1}{3}\max_{e\in\mathcal{E}_h^{\partial}} r_e^3 (C_{ext}^e)^2 (C_{inv}^e)^2 \|\mathrm{L}_h\|^2_{\mathsf{D}_h} \le \frac{1}{24}\|\mathrm{L}_h\|^2_{\mathsf{D}_h}$$

and

$$\|\tau(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h)\|^2_{\Gamma_h,l} \le \max_{e\in\mathcal{E}_h^{\partial}} r_e h_e^{\perp}\tau \|\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h\|^2_{\partial\mathsf{T}_h,\tau} \le \frac{1}{8}\|\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h\|^2_{\partial\mathsf{T}_h,\tau}.$$

These two inequalities, together with (3.11), imply (3.9). □

In order to deal with the term at the right hand side of (3.9), we consider the L^2-projection onto $\boldsymbol{M}_h$, denoted by $\mathbf{P}_M$, and the following assumption: for each $e \in \mathcal{E}_h^{\partial}$.

(A.3) $2\sqrt{6}\beta M \gamma^{1/2} C_{tr}^e r_e^{1/2} < \dfrac{1}{2}$, where $M := \max\left\{1, (\tau h)^{1/2}\right\}$

and $\beta > 0$ is the constant, independent of h, appearing in next lemma.

LEMMA 3.2. *Suppose assumptions of Lemma 3.1 and (A.3) hold. Then* $\widetilde{p}_h = 0$.

Proof. We adapt the proof of Proposition 3.4 in [5] to our setting. Since $\widetilde{p}_h \in L_0^2(\mathsf{D}_h)$, there exists $\beta > 0$, independent of h, such that

$$\|\widetilde{p}_h\|_{\mathsf{D}_h} \le \beta \sup_{\boldsymbol{w}\in\boldsymbol{H}_0^1(\mathsf{D}_h)\setminus\{\boldsymbol{0}\}} \frac{(\widetilde{p}_h, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h}}{\|\boldsymbol{w}\|_{\boldsymbol{H}^1(\mathsf{D}_h)}}. \tag{3.12}$$

On the other hand, let $\boldsymbol{P} : \boldsymbol{H}^1(\mathsf{T}_h) \to \boldsymbol{V}_h$ be any projection such that $(\boldsymbol{P}\boldsymbol{w} - \boldsymbol{w}, \boldsymbol{v})_K = 0$ for all $\boldsymbol{v} \in \boldsymbol{\mathcal{P}}_{k-1}(K)$, for all $K \in \mathsf{T}_h$.

Now, we will work on the numerator of the right hand side of (3.12). Integrating by parts and using the projection $\boldsymbol{P}$, we have

$$\begin{aligned} (\widetilde{p}_h, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h} &= -(\nabla\widetilde{p}_h, \boldsymbol{w})_{\mathsf{T}_h} + \langle \widetilde{p}_h\boldsymbol{n}, \boldsymbol{w}\rangle_{\partial\mathsf{T}_h} \\ &= -(\nabla\widetilde{p}_h, \boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} + \langle \widetilde{p}_h\boldsymbol{n}, \boldsymbol{w}\rangle_{\partial\mathsf{T}_h}. \end{aligned} \tag{3.13}$$

Integrating by parts again and using (3.3b) with $\boldsymbol{v} := \boldsymbol{P}\boldsymbol{w}$, we obtain

$$\begin{aligned} -(\nabla\widetilde{p}_h, \boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} &= (\widetilde{p}_h, \nabla\cdot\boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} - \langle \widetilde{p}_h\boldsymbol{n}, \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h} \\ &= (\nu\mathrm{L}_h, \nabla\boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} - \langle \nu\widehat{\mathrm{L}}_h\boldsymbol{n} - \widehat{p}_h\boldsymbol{n}, \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h} - \langle \widetilde{p}_h\boldsymbol{n}, \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}. \end{aligned}$$

Using the definition of the numerical flux (3.3g), we can write

$$-(\nabla\widetilde{p}_h, \boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} = (\nu\mathrm{L}_h, \nabla\boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} - \langle \nu\mathrm{L}_h\boldsymbol{n}, \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h} + \langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}.$$

Integrating by parts we observe that

$$(\nu\mathrm{L}_h, \nabla\boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} - \langle \nu\mathrm{L}_h\boldsymbol{n}, \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h} = -(\nu\nabla\cdot\mathrm{L}_h, \boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h}.$$

Using the projection $\boldsymbol{P}$ and integrating by parts once again, we obtain

$$-(\nu\nabla\cdot \mathrm{L}_h, \boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} = -(\nu\nabla\cdot \mathrm{L}_h, \boldsymbol{w})_{\mathsf{T}_h} = (\nu \mathrm{L}_h, \nabla\boldsymbol{w})_{\mathsf{T}_h} - \langle \nu \mathrm{L}_h\boldsymbol{n}, \boldsymbol{w}\rangle_{\partial\mathsf{T}_h}.$$

Hence, we have that

$$-(\nabla\widetilde{p}_h, \boldsymbol{P}\boldsymbol{w})_{\mathsf{T}_h} = (\nu \mathrm{L}_h, \nabla\boldsymbol{w})_{\mathsf{T}_h} - \langle \nu \mathrm{L}_h\boldsymbol{n}, \boldsymbol{w}\rangle_{\partial\mathsf{T}_h} + \langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}.$$

Replacing this into (3.13), we conclude that

$$(\widetilde{p}_h, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h} = (\nu \mathrm{L}_h, \nabla\boldsymbol{w})_{\mathsf{T}_h} + \langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{P}\boldsymbol{w}\rangle_{\partial\mathsf{T}_h} - \langle \nu \mathrm{L}_h\boldsymbol{n} - \widetilde{p}_h\boldsymbol{n}, \boldsymbol{w}\rangle_{\partial\mathsf{T}_h}.$$

Adding and subtracting the term $\langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{w}\rangle_{\partial\mathsf{T}_h}$, we can rewrite this equation as

$$\begin{aligned}(\widetilde{p}_h, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h} = (\nu \mathrm{L}_h, \nabla\boldsymbol{w})_{\mathsf{T}_h} &+ \langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{P}\boldsymbol{w} - \boldsymbol{w}\rangle_{\partial\mathsf{T}_h}\\ &- \langle \nu \mathrm{L}_h\boldsymbol{n} - \widetilde{p}_h\boldsymbol{n} - \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{w}\rangle_{\partial\mathsf{T}_h}.\end{aligned}$$

Since $\nu \mathrm{L}_h, \widetilde{p}_h$ and $\tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h)$ are polynomials, we can use the definition of $\mathbf{P}_M$ to rewrite the last equation as

$$\begin{aligned}(\widetilde{p}_h, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h} = (\nu \mathrm{L}_h, \nabla\boldsymbol{w})_{\mathsf{T}_h} &+ \langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{P}\boldsymbol{w} - \mathbf{P}_M\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}\\ &- \langle \nu \mathrm{L}_h\boldsymbol{n} - \widetilde{p}_h\boldsymbol{n} - \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \mathbf{P}_M\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}.\end{aligned}$$

The last term vanishes thanks to (3.3e) with $\boldsymbol{\mu} = \mathbf{P}_M\boldsymbol{w}$ and the fact that $\boldsymbol{w} \in \boldsymbol{H}_0^1(\mathsf{D}_h)$. Thus,

$$(\widetilde{p}_h, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h} = (\nu \mathrm{L}_h, \nabla\boldsymbol{w})_{\mathsf{T}_h} + \langle \tau\nu(\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h), \boldsymbol{P}\boldsymbol{w} - \mathbf{P}_M\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}, \tag{3.14}$$

On the other hand, Proposition 3.9 in [5], states that

$$\max\left\{1, \sup_{\boldsymbol{w}\in\boldsymbol{H}_0^1(\mathsf{D}_h)\setminus\{\mathbf{0}\}} \frac{\|\boldsymbol{P}\boldsymbol{w} - \mathbf{P}_M\boldsymbol{w}\|_{\partial\mathsf{T}_h,\tau}}{\|\boldsymbol{w}\|_{\boldsymbol{H}^1(\mathsf{D}_h)}}\right\} \leq \max\left\{1, \max_{K\in\mathsf{T}_h}(\tau h_K)^{1/2}\right\}. \tag{3.15}$$

Combining (3.12), (3.14), Cauchy-Schwarz inequality and (3.15), we arrive at

$$\|\widetilde{p}_h\|_{\mathsf{D}_h} \leq \beta\nu M(\|\mathrm{L}_h\|_{\mathsf{D}_h} + \|\boldsymbol{u}_h - \widehat{\boldsymbol{u}}_h\|_{\partial\mathsf{T}_h,\tau}) \leq \beta M 2\sqrt{6}\|\widetilde{p}_h\|_{\Gamma_h,l},$$

where the last inequality is provided by Lemma 3.1. Thus, the result follows from this estimate, discrete trace inequality (3.8) and Assumption (A.3). □

Finally, from Lemmas 3.1 and 3.2 we conclude:

THEOREM 3.3. *Under the assumptions of Lemmas 3.1 and 3.2, the scheme* (3.3) *with the boundary data* (2.2) *has a unique solution.*

4. Error estimates. In this section we summarize the main error estimates in this work. First, we make the following assumption that allows us to estimate the L^2-norm of $\boldsymbol{u}$ by using a duality argument.

For any given $\boldsymbol{\theta} \in \boldsymbol{L}^2(\Omega)$, let $(\Phi, \boldsymbol{\phi}, \phi)$ be the solution of

$$\Phi + \nabla\boldsymbol{\phi} = 0 \quad \text{in } \Omega, \tag{4.1a}$$

$$\nabla\cdot(\nu\Phi) - \nabla\phi = \boldsymbol{\theta} \quad \text{in } \Omega, \tag{4.1b}$$

$$-\nabla\cdot\boldsymbol{\phi} = 0 \quad \text{in } \Omega, \tag{4.1c}$$

$$\boldsymbol{\phi} = \mathbf{0} \quad \text{on } \partial\Omega. \tag{4.1d}$$

We suppose

Assumption B.

$$\nu\|\Phi\|_{\mathrm{H}^1(\Omega)} + \nu\|\boldsymbol{\phi}\|_{\boldsymbol{H}^2(\Omega)} + \|\phi\|_{H^1(\Omega)} \leq C\|\boldsymbol{\theta}\|_{\Omega}. \tag{4.2}$$

This is satisfied, for example, for the two-dimensional case if the domain is convex [17] and for any convex polyhedron in three dimensions [11].

THEOREM 4.1. *Suppose that Assumptions A hold. If* $(\mathrm{L}, \boldsymbol{u}, \widetilde{p}) \in H^{k+1}(\Omega) \times \boldsymbol{H}^{k+1}(\Omega) \times H^{k+1}(\Omega)$*, then, for* $h < 1$ *and* $k \geq 1$*, we have*

$$\|\mathrm{L} - \mathrm{L}_h\|_{\mathsf{D}_h} + \|\widetilde{\boldsymbol{g}} - \widetilde{\boldsymbol{g}}_h\|_{\Gamma_h, l^{-1}} + \nu^{-1}\|\widetilde{p} - \widetilde{p}_h\|_{\mathsf{D}_h} \lesssim C_{reg}\, h^{k+1}.$$

Moreover, if Assumption B holds, then $\|\boldsymbol{u} - \boldsymbol{u}_h\|_{\mathsf{D}_h} \lesssim C_{reg}\, h^{k+1}$ *and*

$$\|\mathbf{P}_M\boldsymbol{u} - \widehat{\boldsymbol{u}}_h\|_{\partial\mathsf{T}_h, h} + \|\boldsymbol{u} - \boldsymbol{u}_h^*\|_{\mathsf{D}_h} \lesssim C_{reg}\, h^{k+3/2}.$$

Here, $C_{reg} := \big(|\mathrm{L} - \nu^{-1}\widetilde{p}\mathrm{I}|_{\mathrm{H}^{k+1}(\mathsf{D}_h)} + \tau|\boldsymbol{u}|_{\boldsymbol{H}^{k+1}(\mathsf{D}_h)} + |\mathrm{L}|_{\mathrm{H}^{k+1}(\mathsf{D}_h)}\big)$ and $\boldsymbol{u}_h^*$ is an element-by-element postprocessing of $\boldsymbol{u}_h$ computed as follows. For each element $K \in \mathsf{T}_h$, we seek $\boldsymbol{u}_h^* \in \boldsymbol{\mathcal{P}}_{k+1}^0(K) := \{\boldsymbol{w} \in \boldsymbol{\mathcal{P}}_{k+1}(K) : \int_K \boldsymbol{w} = 0\}$ such that

$$(\nabla\boldsymbol{u}_h^*, \nabla\boldsymbol{w}_h)_K = (\mathrm{L}_h, \nabla\boldsymbol{w}_h)_K \quad \forall \boldsymbol{w}_h \in \boldsymbol{\mathcal{P}}_{k+1}^0(K), \tag{4.3a}$$

$$\int_K \boldsymbol{u}_h^* = \int_K \boldsymbol{u}_h. \tag{4.3b}$$

In other words, if the solution is smooth enough and τ is properly chosen (order one, for instance), then the L^2-norm of the errors in L, $\boldsymbol{u}$ and $\widetilde{p}$ are of order h^{k+1}, which is the same as in the case of a polyhedral domain [5]. In addition, the above estimate indicates that the error in the numerical trace $\widehat{\boldsymbol{u}}_h$ and the post processed solution $\boldsymbol{u}_h^*$ is of only of order $h^{k+3/2}$, which is half a power of h less than the estimates provided in the case of a polyhedral domain [5]. However, in our numerical experiments we observe an experimental order of convergence of h^{k+2} for the latter variables, which suggests that the order $h^{k+3/2}$ provided by the analysis might not be sharp. The same behavior has been observed for the Poisson's equation in curved domains [8].

5. Proofs. This section is devoted to prove the error estimates stated in Theorem 4.1. It follows from the standard procedure for analyzing HDG schemes using the projection-based approach. That is, a suitable projection operator is defined and then an energy argument helps to bound the projection of the errors of all variables except $\boldsymbol{u}$. A duality argument is employed to obtain an L^2-estimate of the projection of the error in $\boldsymbol{u}$. The main difference compared to [5] is the presence of additional terms at Γ_h due to the approximation of the boundary data (2.2).

5.1. An energy argument. First of all, let us recall the projection defined in [5] which will be used in our analysis. If $(\mathrm{L}, \boldsymbol{u}, \widetilde{p}) \in \mathrm{H}^1(\mathsf{T}_h) \times \boldsymbol{H}^1(\mathsf{T}_h) \times H^1(\mathsf{T}_h)$, we take its projection $\mathit{\Pi}_h(\mathrm{L}, \boldsymbol{u}, \widetilde{p}) := (\Pi\mathrm{L}, \boldsymbol{\mathit{\Pi}}\boldsymbol{u}, \mathit{\Pi}\widetilde{p})$ as the element of $\mathrm{G}_h \times \boldsymbol{V}_h \times P_h$ defined as follows. On an arbitrary element K of the triangulation T_h, the values of the projected function on the simplex K are determined by requiring that

$$(\Pi\mathrm{L}, \mathrm{G})_K = (\mathrm{L}, \mathrm{G})_K \qquad \forall\ \mathrm{G} \in \mathrm{P}_{k-1}(K), \tag{5.1a}$$

$$(\boldsymbol{\mathit{\Pi}}\boldsymbol{u}, \boldsymbol{v})_K = (\boldsymbol{u}, \boldsymbol{v})_K \qquad \forall\ \boldsymbol{v} \in \boldsymbol{\mathcal{P}}_{k-1}(K), \tag{5.1b}$$

$$(\mathit{\Pi}\widetilde{p}, q)_K = (\widetilde{p}, q)_K \qquad \forall\ q \in \mathcal{P}_{k-1}(K), \tag{5.1c}$$

$$(\mathrm{tr}\ \Pi\mathrm{L}, q)_K = (\mathrm{tr}\ \mathrm{L}, q)_K \qquad \forall\ q \in \mathcal{P}_k(K), \tag{5.1d}$$

$$\langle \nu\Pi\mathrm{L}\boldsymbol{n} - \mathit{\Pi}\widetilde{p}\boldsymbol{n} - \tau\nu\boldsymbol{\mathit{\Pi}}\boldsymbol{u}, \boldsymbol{\mu}\rangle_e = \langle \nu\mathrm{L}\boldsymbol{n} - \widetilde{p}\boldsymbol{n} - \tau\nu\boldsymbol{u}, \boldsymbol{\mu}\rangle_e, \quad \forall\ \boldsymbol{\mu} \in \boldsymbol{\mathcal{P}}_k(e) \tag{5.1e}$$

for all faces e of the simplex K. Thus, we define the projection of the errors $\mathrm{E}^{\mathrm{L}} := \Pi\mathrm{L} - \mathrm{L}_h$, $\boldsymbol{\varepsilon}^{\boldsymbol{u}} := \boldsymbol{\Pi}\boldsymbol{u} - \boldsymbol{u}_h$, $\varepsilon^p := \Pi\widetilde{p} - \widetilde{p}_h$, $\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}} := \mathbf{P}_M\boldsymbol{u} - \widehat{\boldsymbol{u}}_h$; and the interpolation errors $\mathrm{I}^{\mathrm{L}} := \mathrm{L} - \Pi\mathrm{L}$, $\boldsymbol{I}^{\boldsymbol{u}} := \boldsymbol{u} - \boldsymbol{\Pi}\boldsymbol{u}$, $I^p := \widetilde{p} - \Pi\widetilde{p}$. If $(\mathrm{L}, \boldsymbol{u}, \widetilde{p}) \in \mathrm{H}^{k+1}(K) \times \boldsymbol{H}^{k+1}(K) \times H^{k+1}(K)$ on each element $K \in \mathsf{T}_h$, it is known (Theorem 2.1 in [5]) that the above defined projection satisfies the following properties:

$$\|\boldsymbol{I}^{\boldsymbol{u}}\|_K \lesssim h_K^{k+1}|\boldsymbol{u}|_{\boldsymbol{H}^{k+1}(K)} + h_K^{k+1}(\tau\nu)^{-1}|\nabla\cdot(\nu\mathrm{L} - \mathrm{I}\widetilde{p})|_{\mathrm{H}^k(K)}, \tag{5.2a}$$

$$\|\nu\mathrm{I}^{\mathrm{L}}\|_K + \|I^p\|_K \lesssim h_K^{k+1}|\nu\mathrm{L} - \widetilde{p}\mathrm{I}|_{\mathrm{H}^{k+1}(K)} + h_K^{k+1}\tau\nu|\boldsymbol{u}|_{\boldsymbol{H}^{k+1}(K)} + \tau\nu\|\boldsymbol{I}^{\boldsymbol{u}}\|_K. \tag{5.2b}$$

Moreover, by a standard scaling argument and the fact that $h_e^{\perp} \leq h_K$, we obtain

$$\|\mathrm{I}^{\mathrm{L}}\boldsymbol{n}\|_{e,h_e^{\perp}} \lesssim \|\mathrm{I}^{\mathrm{L}}\|_K, \quad \|\boldsymbol{I}^{\boldsymbol{u}}\|_{e,h_e^{\perp}} \lesssim \|\boldsymbol{I}^{\boldsymbol{u}}\|_K, \quad \text{and} \quad \|I^p\|_{e,h_e^{\perp}} \lesssim \|I^p\|_K. \tag{5.2c}$$

We observe that $\int_{\mathsf{D}_h}\varepsilon^p = 0$ for $k > 0$, thanks to (5.1c) and (3.3f). Since we are interested in a high order method, we assume $k > 0$ to avoid technicalities.

Let $\partial_{\boldsymbol{n}}(\mathrm{I}^{\mathrm{L}}\boldsymbol{n})$ be the directional derivative of each component of $\mathrm{I}^{\mathrm{L}}\boldsymbol{n}$. We define the following quantity that will appear in the right hand side of next estimates:

$$\Theta_{\mathrm{I}^{\mathrm{L}}} := \big(4R^2\|\partial_n(\mathrm{I}^{\mathrm{L}}\boldsymbol{n})\|^2_{\mathsf{D}_h^c,(h^{\perp})^2} + 12R\|\mathrm{I}^{\mathrm{L}}\boldsymbol{n}\|^2_{\Gamma_h,h^{\perp}} + 5\|\mathrm{I}^{\mathrm{L}}\|^2_{\mathsf{D}_h}\big)^{1/2}. \tag{5.3}$$

If $(\mathrm{L}, \boldsymbol{u}, \widetilde{p}) \in \mathrm{H}^{k+1}(\Omega) \times \boldsymbol{H}^{k+1}(\Omega) \times H^{k+1}(\Omega)$, by Lemma 3.8 in [8] and (5.2), we have

$$\Theta_{\mathrm{I}^{\mathrm{L}}} \lesssim h^{k+1}\left(|\mathrm{L} - \nu^{-1}\widetilde{p}\mathrm{I}|_{\mathrm{H}^{k+1}(\Omega)} + \tau|\boldsymbol{u}|_{\boldsymbol{H}^{k+1}(\Omega)} + |\mathrm{L}|_{\mathrm{H}^{k+1}(\Omega)}\right). \tag{5.4}$$

Lemma 5.1. *Let us suppose Assumptions A hold. Then*

$$|\!|\!|\,(\mathrm{E}^{\mathrm{L}}, \boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \varepsilon^p)\,|\!|\!| \leq \left(4\sqrt{3}\beta M + 1\right)\Theta_{\mathrm{I}^{\mathrm{L}}}, \quad \textit{where}$$

$$|\!|\!|\,(\mathrm{E}^{\mathrm{L}}, \boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \varepsilon^p)\,|\!|\!| := \big(\|\mathrm{E}^{\mathrm{L}}\|^2_{\mathsf{D}_h} + \|\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}\|^2_{\partial\mathsf{T}_h,\tau} + \|\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h\|^2_{\Gamma_h,l^{-1}} + \nu^{-2}\|\varepsilon^p\|^2_{\mathsf{D}_h}\big)^{1/2}.$$

We now proceed to detail the steps that lead to the statement of Lemma 5.1. As we will see, we will employ the ideas behind the proofs of Lemmas 3.1 and 3.2.

Step 1: The projection of the errors. The following lemma establishes the equations satisfied by the projection of the errors.

Lemma 5.2.

$$(\mathrm{E}^{\mathrm{L}}, \mathrm{G})_{\mathsf{T}_h} + (\boldsymbol{\varepsilon}^{\boldsymbol{u}}, \nabla\cdot\mathrm{G})_{\mathsf{T}_h} - \langle\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \mathrm{G}\boldsymbol{n}\rangle_{\partial\mathsf{T}_h} = -(\mathrm{I}^{\mathrm{L}}, \mathrm{G})_{\mathsf{T}_h}, \tag{5.5a}$$

$$-(\nabla\cdot(\nu\mathrm{E}^{\mathrm{L}}), \boldsymbol{v})_{\mathsf{T}_h} + (\nabla\varepsilon^p, \boldsymbol{v})_{\mathsf{T}_h} + \langle\tau\nu(\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}), \boldsymbol{v}\rangle_{\partial\mathsf{T}_h} = 0, \tag{5.5b}$$

$$-(\boldsymbol{\varepsilon}^{\boldsymbol{u}}, \nabla q)_{\mathsf{T}_h} + \langle\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, q\boldsymbol{n}\rangle_{\partial\mathsf{T}_h} = 0, \tag{5.5c}$$

$$\langle\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \boldsymbol{\mu}\rangle_{\Gamma_h} = \langle\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \boldsymbol{\mu}\rangle_{\Gamma_h}, \tag{5.5d}$$

$$\langle\hat{\mathrm{E}}\boldsymbol{n}, \boldsymbol{\mu}\rangle_{\partial\mathsf{T}_h\setminus\Gamma_h} = 0, \tag{5.5e}$$

$$(\varepsilon^p, 1)_{\mathsf{D}_h} = 0, \tag{5.5f}$$

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

$$\begin{aligned}\hat{\mathrm{E}}\boldsymbol{n} &:= \mathbf{P}_M(\nu\mathrm{L}\boldsymbol{n} - \widetilde{p}\boldsymbol{n} - \nu\widehat{\mathrm{L}}_h\boldsymbol{n} + \widehat{p}_h\boldsymbol{n})\\ &= \nu\mathrm{E}^{\mathrm{L}}\boldsymbol{n} - \varepsilon^p\boldsymbol{n} - \tau\nu(\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}) \text{ on } \partial\mathsf{T}_h.\end{aligned} \tag{5.5g}$$

Proof. All the equations, except (5.5d), follow directly from Lemma 3.1 in [5]. To show (5.5d), let $\boldsymbol{\mu} \in \boldsymbol{M}_h$. Thanks to (3.3d) and the fact that $\widetilde{\boldsymbol{g}} = \boldsymbol{u}$ on Γ_h, we get $\langle \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \boldsymbol{\mu} \rangle_{\Gamma_h} = \langle \mathbf{P}_M \boldsymbol{u} - \widehat{\boldsymbol{u}}_h, \boldsymbol{\mu} \rangle_{\Gamma_h} = \langle \boldsymbol{u}, \boldsymbol{\mu} \rangle_{\Gamma_h} - \langle \widehat{\boldsymbol{u}}_h, \boldsymbol{\mu} \rangle_{\Gamma_h} = \langle \widetilde{\boldsymbol{g}}, \boldsymbol{\mu} \rangle_{\Gamma_h} - \langle \widetilde{\boldsymbol{g}}_h, \boldsymbol{\mu} \rangle_{\Gamma_h}$. ∎

COROLLARY 5.3. *Let* $\mathbb{T}_{\mathrm{L},h} := \nu^{-1} \langle \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \hat{\mathrm{E}} \boldsymbol{n} \rangle_{\Gamma_h}$. *Then,*

$$\|\mathrm{E}^{\mathrm{L}}\|^2_{\mathsf{D}_h} + \|\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}\|^2_{\partial \mathsf{T}_h, \tau} = -(\mathrm{I}^{\mathrm{L}}, \mathrm{E}^{\mathrm{L}})_{\mathsf{T}_h} + \mathbb{T}_{\mathrm{L},h}.$$

Proof. It is consequence of previous lemma by taking $\mathrm{G} = \nu \mathrm{E}^{\mathrm{L}}$, $\boldsymbol{v} = \boldsymbol{\varepsilon}^{\boldsymbol{u}}$, $q = \varepsilon^p$, $\boldsymbol{\mu} = \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}$ and adding the equations. ∎

If Ω were polygonal and the mesh is fitted to it, then $\widetilde{\boldsymbol{g}}$ would be equal to $\widetilde{\boldsymbol{g}}_h$. Thus, the right hand side of (5.5e) would be zero and, as a consequence, $\mathbb{T}_{\mathrm{L},h}$ would vanish, recovering Lemma 3.1 and Proposition 3.2 in [5].

Step 2: A bound for $\mathbb{T}_{\mathrm{L},h}$. Following the approach by [8], we need to rewrite the term $\mathbb{T}_{\mathrm{L},h}$ in a suitable manner.

LEMMA 5.4.

$$\begin{aligned}\mathbb{T}_{\mathrm{L},h} \leq & - \|\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h\|^2_{\Gamma_h, l^{-1}} + \|\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h\|_{\Gamma_h, l^{-1}} \big\{ \|\delta_{\mathrm{I}^{\mathrm{L}}}\|_{\Gamma_h, l} + \|\mathrm{I}^{\mathrm{L}} \boldsymbol{n}\|_{\Gamma_h, l} \\ & + \|\delta_{\mathrm{E}^{\mathrm{L}}}\|_{\Gamma_h, l} + \|\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}\|_{\Gamma_h, \tau^2 l} + \nu^{-1} \|\varepsilon^p\|_{\Gamma_h, l} \big\}.\end{aligned}$$

Proof. Proceeding exactly as in Lemma 5.1 of [8], it is not difficult to see that, according to (2.1), (2.2) and (3.4), we can decompose $(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)(\boldsymbol{x}) = -l(\boldsymbol{x}) \big\{ \delta_{\mathrm{I}^{\mathrm{L}}} + \mathrm{I}^{\mathrm{L}} \boldsymbol{n} + \delta_{\mathrm{E}^{\mathrm{L}}} + \mathrm{E}^{\mathrm{L}} \boldsymbol{n} \big\}(\boldsymbol{x})$. Then, by (5.5g), we obtain

$$\hat{\mathrm{E}} \boldsymbol{n} = -\nu \big\{ (\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l + \delta_{\mathrm{I}^{\mathrm{L}}} + \mathrm{I}^{\mathrm{L}} \boldsymbol{n} + \delta_{\mathrm{E}^{\mathrm{L}}} \big\} - \varepsilon^p \boldsymbol{n} - \tau \nu (\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}). \tag{5.6}$$

On the other hand, by the definition of $\mathbb{T}_{\mathrm{L},h}$ and (5.5d),

$$\mathbb{T}_{\mathrm{L},h} := \nu^{-1} \langle \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \hat{\mathrm{E}} \boldsymbol{n} \rangle_{\Gamma_h} = \nu^{-1} \langle \mathbf{P}_M (\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h), \hat{\mathrm{E}} \boldsymbol{n} \rangle_{\Gamma_h} = \nu^{-1} \langle (\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h), \hat{\mathrm{E}} \boldsymbol{n} \rangle_{\Gamma_h},$$

which, according to (5.6), leads to the decomposition $\mathbb{T}_{\mathrm{L},h} = \sum_{i=1}^{6} \mathbb{T}^i_{\mathrm{L},h}$, where

$$\begin{array}{llll} \mathbb{T}^1_{\mathrm{L},h} = -\langle (\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l, (\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h) \rangle_{\Gamma_h}, & \mathbb{T}^2_{\mathrm{L},h} = -\langle \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \delta_{\mathrm{I}^{\mathrm{L}}} \rangle_{\Gamma_h}, \\ \mathbb{T}^3_{\mathrm{L},h} = -\langle \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \mathrm{I}^{\mathrm{L}} \boldsymbol{n} \rangle_{\Gamma_h}, & \mathbb{T}^4_{\mathrm{L},h} = -\langle \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \delta_{\mathrm{E}^{\mathrm{L}}} \rangle_{\Gamma_h}, \\ \mathbb{T}^5_{\mathrm{L},h} = -\langle \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \tau (\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}) \rangle_{\Gamma_h}, & \mathbb{T}^6_{\mathrm{L},h} = -\nu^{-1} \langle \tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \varepsilon^p \boldsymbol{n} \rangle_{\Gamma_h}. \end{array}$$

The result is obtained by applying Cauchy-Schwarz inequality to each of these terms. ∎

Step 3: A first energy estimate. We recall the estimate in Lemma 5.2 of [8]:

$$\|l^{1/2} \delta_{\mathrm{I}^{\mathrm{L}}}\|_e \leq 3^{-\frac{1}{2}} r_e \|h_e^{\perp} \partial_n (\mathrm{I}^{\mathrm{L}} \boldsymbol{n})\|_{K^e_{ext}}. \tag{5.7}$$

Thus, gathering the result stated in Corollary 5.3, estimate in Lemma 5.4, inequality (5.7), (3.5) with $\mathrm{G} = \mathrm{E}^{\mathrm{L}}$, Young's inequality, definition (5.3) and noticing that $\|\mathrm{I}^{\mathrm{L}} \boldsymbol{n}\|^2_{\Gamma_h, l} \leq R \|\mathrm{I}^{\mathrm{L}} \boldsymbol{n}\|^2_{\Gamma_h, h^{\perp}}$, we conclude:

LEMMA 5.5. $\|\mathrm{E}^{\mathrm{L}}\|^2_{\mathsf{D}_h} + \|\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}\|^2_{\partial \mathsf{T}_h, \tau} + \|\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h\|^2_{\Gamma_h, l^{-1}} \leq \Theta^2_{\mathrm{I}^{\mathrm{L}}} + 12 \nu^{-2} \|\varepsilon^p\|^2_{\Gamma_h, l}$.

Comparing this estimate with the corresponding result in [5] for polyhedral domains, we see in our case the presence of the terms $\|\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h\|^2_{\Gamma_h, l^{-1}}$ and $12 \nu^{-2} \|\varepsilon^p\|^2_{\Gamma_h, l}$ coming from $\mathbb{T}_{\mathrm{L},h}$. In other words, for a polyhedral domain, we would have $\mathbb{T}_{\mathrm{L},h} = 0$

and recover the estimates in [5]. We also observe that the estimate in Lemma 5.5 is similar to the one obtained in Lemma 3.1, except that the right hand side involves terms related to the interpolation operator instead of the source term and boundary data. Similarly, we must bound the term that contains ε^p and proceed as in Lemma 3.2.

LEMMA 5.6. *If Assumption (A.3) holds, then* $\|\varepsilon^p\|_{\mathsf{D}_h} \leq 3\sqrt{2}\beta\nu M\Theta_{\mathrm{I^L}}$.

Proof. The proof of this lemma is very similar to that of Lemma 3.2. Since $\varepsilon^p \in L^2_0(\mathsf{D}_h)$, there exists $\beta > 0$ such that

$$\|\varepsilon^p\|_{\mathsf{D}_h} \leq \beta \sup_{\boldsymbol{w}\in\boldsymbol{H}^1_0(\mathsf{D}_h)\setminus\{\mathbf{0}\}} \frac{(\varepsilon^p, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h}}{\|\boldsymbol{w}\|_{\boldsymbol{H}^1(\mathsf{D}_h)}}. \tag{5.8}$$

On the other hand, similarly to the proof of Lemma 3.2, using (5.5e), we get

$$(\varepsilon^p, \nabla\cdot\boldsymbol{w})_{\mathsf{D}_h} = (\nu \mathrm{E^L}, \nabla\boldsymbol{w})_{\mathsf{T}_h} + \langle \tau\nu(\boldsymbol{\varepsilon^u} - \boldsymbol{\varepsilon^{\hat{u}}}), \boldsymbol{P}\boldsymbol{w} - \mathbf{P}_M\boldsymbol{w}\rangle_{\partial\mathsf{T}_h}, \tag{5.9}$$

for $\boldsymbol{w} \in \boldsymbol{H}^1_0(\mathsf{D}_h)$. Using Cauchy-Schwarz inequality in this expression, combining (5.8), (3.15), Lemma 5.5, trace inequality (3.8) and Assumption (A.4), the result follows. □

If Ω were polyhedral, this result reduces to the estimate of Proposition 3.4 in [5].

Step 4: Conclusion of the proof of Lemma 5.1. Gathering the bounds obtained in Lemmas 5.5 and 5.6, we obtain the estimate stated in Lemma 5.1.

5.2. A duality argument. We recall ([5]) properties of the projection $\mathit{\Pi}_h$ that will be used to obtain estimates for the velocity. Let $(\Phi, \boldsymbol{\phi}, \phi) \in \mathrm{H}^1(\mathsf{T}_h) \times \boldsymbol{H}^1(\mathsf{T}_h) \times H^1(\mathsf{T}_h)$. Then, for all $(\mathrm{G}, \boldsymbol{v}, q) \in \mathrm{G}_h \times \boldsymbol{V}_h \times P_h$,

$$(\boldsymbol{v}, \nabla\cdot\Phi)_{\mathsf{T}_h} = (\boldsymbol{v}, \nabla\cdot\Pi\Phi)_{\mathsf{T}_h} + \langle \boldsymbol{v}, (\Phi - \Pi\Phi)\boldsymbol{n}\rangle_{\partial\mathsf{T}_h}, \tag{5.10a}$$

$$(\mathrm{G}, \nabla\boldsymbol{\phi})_{\mathsf{T}_h} = -(\nabla\cdot\mathrm{G}, \boldsymbol{\mathit{\Pi}}\boldsymbol{\phi})_{\mathsf{T}_h} + \langle \mathrm{G}\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\partial\mathsf{T}_h}, \tag{5.10b}$$

$$(q, \nabla\cdot\boldsymbol{\phi})_{\mathsf{T}_h} = -(\nabla q, \boldsymbol{\mathit{\Pi}}\boldsymbol{\phi})_{\mathsf{T}_h} + \langle q\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\partial\mathsf{T}_h}, \tag{5.10c}$$

$$(\boldsymbol{v}, \nabla\phi)_{\mathsf{T}_h} = (\boldsymbol{v}, \nabla\mathit{\Pi}\phi)_{\mathsf{T}_h} + \langle \boldsymbol{v}, (\phi - \mathit{\Pi}\phi)\boldsymbol{n}\rangle_{\partial\mathsf{T}_h}. \tag{5.10d}$$

The main estimate of this section is the following.

LEMMA 5.7. *Let* $H(R,h) := h^{1/2}(1+\tau) + R + R^{1/2} + h^{1/2}\tau^{1/2}R + R^{3/2}$. *If Assumptions A and B hold, and* $h \leq 1$, *then*

$$\|\boldsymbol{\varepsilon^u}\|_{\mathsf{D}_h} \lesssim h^{1/2}\big(H(R,h)\Theta_{\mathrm{I^L}} + R\nu^{-1}\|I^p\|_{\mathsf{D}_h} + R\tau\|\boldsymbol{I^u}\|_{\mathsf{D}_h}\big).$$

We observe that, for a smooth solution, if τ and R are of order one, then $\|\boldsymbol{\varepsilon^u}\|_{\mathsf{D}_h} \lesssim h^{k+3/2}$ since the interpolation errors are of order h^{k+1}. In the case of a polyhedral domain where Γ_h fits Γ, we would have $R = 0$, $\tilde{\boldsymbol{g}} = \tilde{\boldsymbol{g}}_h$. As a consequence, $\|\boldsymbol{\varepsilon^u}\|_{\mathsf{D}_h}$ would be of order h^{k+2} which agrees with the estimates in [5]. We now detail the steps that construct the proof of Lemma 5.7.

Step 1: Estimate of the velocity. We first obtain an identity for the projection of the error in the velocity by using a duality argument.

LEMMA 5.8. *Let* $\mathbb{T}_{\boldsymbol{u},h} := \langle \hat{\mathrm{E}}\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h} + \langle \boldsymbol{\varepsilon^{\hat{u}}}, \nu\Phi\boldsymbol{n} - \phi\boldsymbol{n}\rangle_{\Gamma_h}$ *and* P_{k-1} *the* L^2*-projection over the space of piecewise polynomials of degree at most* $k-1$. *Then,*

$$(\boldsymbol{\varepsilon^u}, \boldsymbol{\theta})_{\mathsf{T}_h} = \nu(\mathrm{L}_h - \mathrm{L}, \Pi\Phi - \Phi)_{\mathsf{T}_h} - \nu(\mathrm{I^L}, \Phi - \mathrm{P}_{k-1}\Phi)_{\mathsf{T}_h} + \mathbb{T}_{\boldsymbol{u},h}.$$

Proof. Proceeding as in the proof of Lemma 3.6 of [5], but keeping the boundary terms since in our case they are not zero, we can write $(\boldsymbol{\varepsilon}^{\boldsymbol{u}},\boldsymbol{\theta})_{\mathsf{T}_h} = \sum_{i=1}^{5} T_i$, where

$$
\begin{aligned}
&T_1 := \nu(\mathrm{L}_h - \mathrm{L}, \Pi\Phi - \Phi)_{\mathsf{T}_h}, \quad T_2 := -\nu(\mathrm{I}^{\mathrm{L}}, \Phi)_{\mathsf{T}_h},\\
&T_3 := \langle \boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \nu(\Phi - \Pi\Phi)\boldsymbol{n} - (\phi - \mathit{\Pi}\phi)\boldsymbol{n} + \tau\nu(\boldsymbol{\phi} - \boldsymbol{\mathit{\Pi}}\boldsymbol{\phi})\rangle_{\partial\mathsf{T}_h},\\
&T_4 := \langle \nu\mathrm{E}^{\mathrm{L}}\boldsymbol{n} - \varepsilon^p\boldsymbol{n} - \tau\nu(\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}), \boldsymbol{\phi}\rangle_{\partial\mathsf{T}_h}, \quad T_5 := \langle \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \nu\Phi\boldsymbol{n} - \phi\boldsymbol{n}\rangle_{\partial\mathsf{T}_h}.
\end{aligned}
$$

By the property of the projection (5.1a), we have that $T_2 = -\nu(\mathrm{I}^{\mathrm{L}}, \Phi - \mathrm{P}_{k-1}\Phi)_{\mathsf{T}_h}$, and by (5.1e) with $\boldsymbol{\mu} = \boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}$, we have that $T_3 = 0$. Moreover, using the error equation (5.5e) we obtain that $T_4 = \langle \hat{\mathrm{E}}\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h}$. Finally, since $\nu\Phi - \phi\mathrm{I} \in \mathrm{H}(\mathrm{div}, \Omega)$ and $\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}$ is single-valued, then $T_5 = \langle \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \nu\Phi\boldsymbol{n} - \phi\boldsymbol{n}\rangle_{\Gamma_h}$, which concludes the proof. ∎

We point out that $\phi = 0$ on Γ, but not in Γ_h. Thus, if Γ and Γ_h were the same (as in the polyhedral case), then T_4 would be zero. Moreover, for a polyhedral domain Ω, T_5 would also vanish thanks to (5.5e) since $\widetilde{\boldsymbol{g}} = \widetilde{\boldsymbol{g}}_h$, recovering the equality stated in Lemma 3.6 of [5].

Step 2: A new expression for $\mathbb{T}_{\boldsymbol{u},h}$. As we did previously with $\mathbb{T}_{\mathrm{L},h}$, we proceed now to rewrite $\mathbb{T}_{\boldsymbol{u},h}$ in a suitable manner.

LEMMA 5.9. *We have that* $\mathbb{T}_{\boldsymbol{u},h} = \sum_{i=1}^{10} \mathbb{T}^i_{\boldsymbol{u},h}$, *where*

$$
\begin{aligned}
&\mathbb{T}^1_{\boldsymbol{u},h} = -\nu\langle(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l, \phi + \ l\nabla\boldsymbol{\phi}\boldsymbol{n}\rangle_{\Gamma_h}, \quad \mathbb{T}^2_{\boldsymbol{u},h} = \nu\langle\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \nabla\boldsymbol{\phi}\boldsymbol{n} - \mathbf{P}_M(\nabla\boldsymbol{\phi}\boldsymbol{n})\rangle_{\Gamma_h},\\
&\mathbb{T}^3_{\boldsymbol{u},h} = -\nu\langle\delta_{\mathrm{I}^{\mathrm{L}}}, \boldsymbol{\phi}\rangle_{\Gamma_h}, \quad \mathbb{T}^4_{\boldsymbol{u},h} = -\nu\langle\mathrm{I}^{\mathrm{L}}\boldsymbol{n}, \boldsymbol{\phi} - \mathbf{P}_M\boldsymbol{\phi}\rangle_{\Gamma_h}, \quad \mathbb{T}^5_{\boldsymbol{u},h} = -\langle I^p\boldsymbol{n}, \mathbf{P}_M\boldsymbol{\phi}\rangle_{\Gamma_h},\\
&\mathbb{T}^6_{\boldsymbol{u},h} = -\nu\langle\mathbf{P}_M\tau\boldsymbol{I}^{\boldsymbol{u}}, \boldsymbol{\phi}\rangle_{\Gamma_h}, \quad \mathbb{T}^7_{\boldsymbol{u},h} = -\nu\langle\delta_{E^{\mathrm{L}}}, \boldsymbol{\phi}\rangle_{\Gamma_h}, \quad \mathbb{T}^8_{\boldsymbol{u},h} = -\langle\varepsilon^p\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h},\\
&\mathbb{T}^9_{\boldsymbol{u},h} = -\nu\langle\tau(\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}), \boldsymbol{\phi}\rangle_{\Gamma_h}, \quad \mathbb{T}^{10}_{\boldsymbol{u},h} = -\langle\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \mathbf{P}_M(\phi\boldsymbol{n})\rangle_{\Gamma_h}.
\end{aligned}
$$

Proof. We recall (5.5g):

$$
\hat{\mathrm{E}}\boldsymbol{n} = -\nu\left\{(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l + \delta_{\mathrm{I}^{\mathrm{L}}} + \mathrm{I}^{\mathrm{L}}\boldsymbol{n} + \delta_{\mathrm{E}^{\mathrm{L}}}\right\} - \varepsilon^p\boldsymbol{n} - \nu\tau(\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}).
$$

Then, since $\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}} = \mathbf{P}_M(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)$ on Γ_h, we obtain

$$
\begin{aligned}
\mathbb{T}_{\boldsymbol{u},h} &= \langle\hat{\mathrm{E}}\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h} + \langle\boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}, \nu\Phi\boldsymbol{n} - \phi\boldsymbol{n}\rangle_{\Gamma_h}\\
&= \nu\langle(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l, \ l\mathbf{P}_M(\Phi\boldsymbol{n}) - \boldsymbol{\phi}\rangle_{\Gamma_h} - \nu\langle\delta_{\mathrm{I}^{\mathrm{L}}}, \boldsymbol{\phi}\rangle_{\Gamma_h} - \nu\langle\mathrm{I}^{\mathrm{L}}\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h}\\
&\quad -\nu\langle\delta_{\mathrm{E}^{\mathrm{L}}}, \boldsymbol{\phi}\rangle_{\Gamma_h} - \langle\varepsilon^p\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h} - \nu\langle\tau(\boldsymbol{\varepsilon}^{\boldsymbol{u}} - \boldsymbol{\varepsilon}^{\hat{\boldsymbol{u}}}), \boldsymbol{\phi}\rangle_{\Gamma_h} - \langle\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \mathbf{P}_M(\phi\boldsymbol{n})\rangle_{\Gamma_h}.
\end{aligned}
$$

Using the first equation of the dual problem (4.1a), adding and subtracting $l\nabla\boldsymbol{\phi}\boldsymbol{n}$,

$$
\begin{aligned}
\nu\langle(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l, \ l\mathbf{P}_M(\Phi\boldsymbol{n}) - \boldsymbol{\phi}\rangle_{\Gamma_h} &= -\nu\langle(\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h)/l, \ l\nabla\boldsymbol{\phi}\boldsymbol{n} + \boldsymbol{\phi}\rangle_{\Gamma_h}\\
&\quad +\nu\langle\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h, \nabla\boldsymbol{\phi}\boldsymbol{n} - \mathbf{P}_M(\nabla\boldsymbol{\phi}\boldsymbol{n})\rangle_{\Gamma_h}.
\end{aligned}
$$

On the other hand, adding and subtracting $\mathbf{P}_M\boldsymbol{\phi}$, and using equation (5.1e), we get $\nu\langle\mathrm{I}^{\mathrm{L}}\boldsymbol{n}, \boldsymbol{\phi}\rangle_{\Gamma_h} = \nu\langle\mathrm{I}^{\mathrm{L}}\boldsymbol{n}, \boldsymbol{\phi} - \mathbf{P}_M\boldsymbol{\phi}\rangle_{\Gamma_h} + \langle I^p\boldsymbol{n}, \mathbf{P}_M\boldsymbol{\phi}\rangle_{\Gamma_h} + \nu\langle\tau\boldsymbol{I}^{\boldsymbol{u}}, \mathbf{P}_M\boldsymbol{\phi}\rangle_{\Gamma_h}$. The result follows by gathering the above equalities. ∎

Step 3: Estimate of $\mathbb{T}_{\boldsymbol{u},h}$. In order to obtain the estimate of $\mathbb{T}_{\boldsymbol{u},h}$, we need the following lemma, which provides bounds related to the solution of the dual problem (4.1a)-(4.1d) that will be used later.

LEMMA 5.10. *Suppose that the elliptic regularity inequality (4.2) holds. Then*

$$
\begin{aligned}
&\nu\|\boldsymbol{\phi} - \mathbf{P}_M\boldsymbol{\phi}\|_{\Gamma_h,(h^{\perp})^{-1}} \lesssim h\|\boldsymbol{\theta}\|_{\Omega}, \quad \nu\|\nabla\boldsymbol{\phi}\boldsymbol{n} - \mathbf{P}_M(\nabla\boldsymbol{\phi}\boldsymbol{n})\|_{\Gamma_h,l} \lesssim hR\|\boldsymbol{\theta}\|_{\Omega},\\
&\nu\|\boldsymbol{\phi} + \ l\nabla\boldsymbol{\phi}\boldsymbol{n}\|_{\Gamma_h,l^{-3}} \lesssim \|\boldsymbol{\theta}\|_{\Omega}, \quad \nu\|\boldsymbol{\phi}\|_{\Gamma_h,l^{-2}} \lesssim \|\boldsymbol{\theta}\|_{\Omega}, \quad \|\phi\boldsymbol{n}\|_{\Gamma_h,l} \lesssim h^{1/2}R^{1/2}\|\boldsymbol{\theta}\|_{\Omega}.
\end{aligned}
$$

Proof. The first four estimates follow by applying Lemma 5.5 of [8] to each component of a tensor- or a vector-valued function. For the last one, we observe that for $e \in \mathcal{E}_h^\partial$, $\sum_{e\in\mathcal{E}_h^\partial} \|\phi\boldsymbol{n}\|_{e,l}^2 \leq \sum_{e\in\mathcal{E}_h^\partial} r_e h_e^\perp \|\phi\|_e^2 \lesssim h\|\phi\|_{\Gamma_h}^2 \max_{e\in\mathcal{E}_h^\partial} r_e$. Now, since $\phi \in H^1(\Omega)$, by trace inequality we have that $\|\phi\|_{\Gamma_h} \lesssim \|\phi\|_{\mathsf{D}_h} \leq \|\phi\|_\Omega$, and the result follows. ∎

We are now ready to obtain the estimate of $\mathbb{T}_{\boldsymbol{u},h}$.

LEMMA 5.11. *We have that*

$$|\mathbb{T}_{\boldsymbol{u},h}| \lesssim h^{1/2}\Big\{(R^{3/2}+h^{1/2}+R+R^{1/2}+h^{1/2}\tau^{1/2}R)\Theta_{\mathrm{I}^{\mathrm{L}}} + R\nu^{-1}\|I^p\|_{\mathsf{D}_h} + R\tau\|\boldsymbol{I}^{\boldsymbol{u}}\|_{\mathsf{D}_h}\Big\}\|\boldsymbol{\theta}\|_\Omega.$$

Proof. By Lemma 5.9, we know that $\mathbb{T}_{\boldsymbol{u},h} = \sum_{i=1}^{10} \mathbb{T}_{\boldsymbol{u},h}^i$. We first apply the Cauchy-Schwarz inequality to each term $\mathbb{T}_{\boldsymbol{u},h}^i$. The result follows from estimate in (3.7), Lemma 5.10, Assumptions A, the fact that $\|I^p\boldsymbol{n}\|_{\Gamma_h,l^2} \leq Rh^{1/2}\|I^p\boldsymbol{n}\|_{\Gamma_h,h^\perp}$, the interpolation properties (5.2c) and Lemma 5.1 . ∎

Step 4: Proof of Lemma 5.7. From Lemma 5.8, we have

$$(\boldsymbol{\varepsilon}^{\boldsymbol{u}},\boldsymbol{\theta})_{\mathsf{T}_h} \leq \|\mathrm{L}_h - \mathrm{L}\|_{\mathsf{D}_h}\|\nu\Pi\Phi - \nu\Phi\|_{\mathsf{D}_h} + \|\mathrm{I}^{\mathrm{L}}\|_{\mathsf{D}_h}\|\nu\Phi - \nu\mathrm{P}_{k-1}\Phi\|_{\mathsf{D}_h} + |\mathbb{T}_{\boldsymbol{u},h}|.$$

According to the regularity Assumption B, (5.2a) implies that $\|\nu\Pi\Phi - \nu\Phi\|_{\mathsf{D}_h} \lesssim h(1+\tau h)\|\boldsymbol{\theta}\|_\Omega$. Also, by the approximation properties of the L^2-projection, we have $\|\nu\Phi - \nu\mathrm{P}_{k-1}\Phi\|_{\mathsf{D}_h} \lesssim h\|\boldsymbol{\theta}\|_\Omega$. We recall we are considering $k \geq 1$. Hence,

$$(\boldsymbol{\varepsilon}^{\boldsymbol{u}},\boldsymbol{\theta})_{\mathsf{T}_h} \lesssim h(1+\tau h)\left(\|\mathrm{E}^{\mathrm{L}}\|_{\mathsf{D}_h} + \|\mathrm{I}^{\mathrm{L}}\|_{\mathsf{D}_h}\right)\|\boldsymbol{\theta}\|_\Omega + |\mathbb{T}_{\boldsymbol{u},h}|.$$

Thus, using the estimate for $|\mathbb{T}_{\boldsymbol{u},h}|$ from Lemma 5.11, recalling that $\|\mathrm{E}^{\mathrm{L}}\|_{\mathsf{D}_h} \leq \Theta_{\mathrm{I}^{\mathrm{L}}}$ from Lemma 5.1, dividing by $\|\boldsymbol{\theta}\|_\Omega$, $\boldsymbol{\theta} \neq \mathbf{0}$, and considering the supremum over $\boldsymbol{\theta}$, we obtain the result.

5.3. Conclusion of the proof of Theorem 4.1. First of all, adding and subtracting $\Pi\mathrm{L}$ and $\Pi\widetilde{p}$, using the triangle inequality, Lemma 5.1 and recalling the definition of $\Theta_{\mathrm{I}^{\mathrm{L}}}$ in (5.3), we get

$$\|\mathrm{L} - \mathrm{L}_h\|_{\mathsf{D}_h} + \|\tilde{\boldsymbol{g}} - \tilde{\boldsymbol{g}}_h\|_{\Gamma_h,l^{-1}} + \nu^{-1}\|\widetilde{p} - \widetilde{p}_h\|_{\mathsf{D}_h} \lesssim \Theta_{\mathrm{I}^{\mathrm{L}}} + \nu^{-1}\|I^p\|_{\mathsf{D}_h}.$$

Moreover, if Assumption B holds, by adding and subtracting $\boldsymbol{\Pi u}$, using triangle inequality and Lemma 5.7, we obtain

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_{\mathsf{D}_h} \lesssim h^{1/2}\big(H(R,h)\Theta_{\mathrm{I}^{\mathrm{L}}} + R\nu^{-1}\|I^p\|_{\mathsf{D}_h} + R\tau\|\boldsymbol{I}^{\boldsymbol{u}}\|_{\mathsf{D}_h}\big) + \|\boldsymbol{I}^{\boldsymbol{u}}\|_{\mathsf{D}_h}.$$

Lemma 3.7 in [5] states that $\|\boldsymbol{\varepsilon}^{\widehat{\boldsymbol{u}}}\|_h \lesssim h\|\mathrm{I}^{\mathrm{L}}\|_{\mathsf{D}_h} + h\|\mathrm{E}^{\mathrm{L}}\|_{\mathsf{D}_h} + \|\boldsymbol{\varepsilon}^{\boldsymbol{u}}\|_{\mathsf{D}_h}$ which, together with Lemmas 5.1 and 5.7, implies

$$\|\mathbf{P}_M\boldsymbol{u} - \widehat{\boldsymbol{u}}_h\|_h \lesssim h^{1/2}\big((h^{1/2}+H(R,h))\Theta_{\mathrm{I}^{\mathrm{L}}} + R\nu^{-1}\|I^p\|_{\mathsf{D}_h} + R\tau\|\boldsymbol{I}^{\boldsymbol{u}}\|_{\mathsf{D}_h}\big).$$

The error estimate $\|\boldsymbol{u} - \boldsymbol{u}_h^*\|_{\mathsf{D}_h} \leq \|\boldsymbol{\varepsilon}^{\boldsymbol{u}}\|_{\mathsf{D}_h} + Ch\|\mathrm{L} - \mathrm{L}_h\|_{\mathsf{D}_h} + Ch^{k+2}|\mathrm{L}|_{\mathrm{H}^{k+1}(\Omega)}$ can be found in [6] and, from Lemma 5.7, follows that

$$\|\boldsymbol{u} - \boldsymbol{u}_h^*\|_{\mathsf{D}_h} \lesssim h^{1/2}\big(H(R,h)\Theta_{\mathrm{I}^{\mathrm{L}}} + R\nu^{-1}\|I^p\|_{\mathsf{D}_h} + R\tau\|\boldsymbol{I}^{\boldsymbol{u}}\|_{\mathsf{D}_h}\big) + h\|\mathrm{L} - \mathrm{L}_h\|_{\mathsf{D}_h} + h^{k+2}|\mathrm{L}|_{\mathrm{H}^{k+1}(\Omega)}.$$

Finally, according to assumption $(D.4)$, we have $R \leq Ch \leq C$, which implies that $H(R,h) \lesssim 1 + \tau + \tau^{1/2}$ and the estimates stated in Theorem 4.1 follow.

6. Approximation in D_h^c and recovering p_h. In this section we provide a way to extend the solution $(\mathrm{L}_h, \boldsymbol{u}_h, \widetilde{p}_h)$ of (3.3) to the non-meshed region D_h^c in the two-dimensional case. The case of three dimensions can be treated similarly. In addition, since the scheme (3.3) provides an approximation of $\widetilde{p}_h$, here we will explain a procedure to recover p_h. To that end, we write D_h^c as the union of disjoint regions. More precisely, let $e \in \mathcal{E}_h^\partial$ that belongs to $K^e \in \mathsf{T}_h$. For $j \in \{1, 2\}$, we denote by $\boldsymbol{y}_j$, the j-th vertex of e and by $\overline{\boldsymbol{y}}_j$ its closest point in Γ. We remark that the closest point is unique if we assume

(D.6) the mesh in fine enough and Γ is $\mathcal{C}^2$.

We define $\widetilde{K}_{ext}^e$ as the region enclosed by Γ, e and the transferring segments associated to $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$, as Fig. 2 shows.

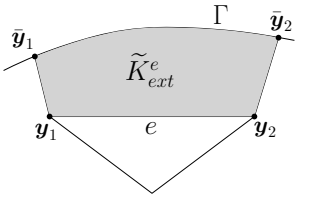

Fig. 2: Example of region $\widetilde{K}_{ext}^e$.

Now, the polynomials $\mathrm{L}_h|_{K^e}$ and $\widetilde{p}_h|_{K^e}$ are extrapolated from K^e to $\widetilde{K}_{ext}^e$ and they constitute an approximation of L and $\widetilde{p}$ in $\widetilde{K}_{ext}^e$. We can also approximate $\boldsymbol{u}$ in $\widetilde{K}_{ext}^e$ by extrapolating $u_h|_{K^e}$, however we construct a better approximation based on the fact that L is the gradient of $\boldsymbol{u}$. Let $\boldsymbol{y} \in \widetilde{K}_{ext}^e$ and $\overline{\boldsymbol{y}}$ its closest point in Γ. Let $\boldsymbol{x} \in e$ be the intersection between e and the line determined by $\boldsymbol{y}$ and $\overline{\boldsymbol{y}}$. Then, denoting by $\boldsymbol{t}(\boldsymbol{x})$ the unit tangent vector of the segment joining $\boldsymbol{x}$ and $\overline{\boldsymbol{x}}$, we can write $\boldsymbol{y} = \boldsymbol{x} + \eta \boldsymbol{t}(\boldsymbol{x})$ and $\bar{\boldsymbol{y}} := \boldsymbol{x} + \bar{\eta}\boldsymbol{t}(\boldsymbol{x})$, for some η and $\bar{\eta}$ positive. Thus, based on the procedure to obtain (2.1), we integrate $\mathrm{L} = \nabla \boldsymbol{u}$ along the segment connecting $\boldsymbol{y}$ and $\bar{\boldsymbol{y}}$, and propose the following approximation of the velocity in $\boldsymbol{y}$,

$$\boldsymbol{u}_h(\boldsymbol{y}) := \boldsymbol{g}(\bar{\boldsymbol{y}}) - \int_\eta^{\bar{\eta}} \mathrm{L}_h(\boldsymbol{y} + \boldsymbol{t}(\boldsymbol{x})s\,)\boldsymbol{t}(\boldsymbol{x})\, ds. \tag{6.1}$$

On the other hand, since $0 = \displaystyle\int_\Omega p = |\Omega|\,\overline{p}^{\mathsf{D}_h} + \int_{\mathsf{D}_h^c} \widetilde{p}$, we suggest the following approximation

$$\overline{p}^{\mathsf{D}_h} \approx \overline{p_h}^{\mathsf{D}_h} := -\frac{1}{|\Omega|}\int_{\mathsf{D}_h^c} \widetilde{p}_h, \tag{6.2}$$

thereby

$$p_h := \overline{p_h}^{\mathsf{D}_h} + \widetilde{p}_h \tag{6.3}$$

is an approximation of p. In the Appendix we explain how to approximate an integral over D_h^c.

Thus, we have defined approximations of L, $\widetilde{p}$ and $\boldsymbol{u}$ in $\mathsf{D}_h^c = \displaystyle\bigcup_{e \in \mathcal{E}_h^\partial} \widetilde{K}_{ext}^e$. Even though the variables L_h, $\widetilde{p}_h$ and $\boldsymbol{u}_h$ denote the approximations delivered by the HDG

method in D_h, we use the same notation to represent the approximations in D_h^c proposed in this section.

Finally, from Lemmas 3.7 and 3.8 in [8], it can be obtained that

$$\|\mathrm{L}-\mathrm{L}_h\|_{\mathsf{D}_h^c} \lesssim \|\mathrm{I}^{\mathrm{L}}\|_{\mathsf{D}_h} + \|\mathrm{E}^{\mathrm{L}}\|_{\mathsf{D}_h} + h^{k+1}|\mathrm{L}|_{\mathrm{H}^{k+1}(\Omega)}, \tag{6.4a}$$

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathsf{D}_h^c} \lesssim h\|\mathrm{L}-\mathrm{L}_h\|_{\mathsf{D}_h^c} \quad \text{and} \tag{6.4b}$$

$$\|\widetilde{p}-\widetilde{p}_h\|_{\mathsf{D}_h^c} \lesssim \big(\|\mathrm{I}^p\|_{\mathsf{D}_h} + \|\varepsilon^p\|_{\mathsf{D}_h}\big) + h^{k+1}|\widetilde{p}|_{H^{k+1}(\mathsf{D}_h)}. \tag{6.4c}$$

Actually the estimates of Lemmas 3.7 and 3.8 in [8] were obtained based on a different projection of a point into Γ. However, if the closest projection is considered instead, the same results hold under minor modifications in the proofs.

THEOREM 6.1. *Suppose assumptions of Theorem 4.1 hold, then*

$$\|\mathrm{L}-\mathrm{L}_h\|_{\mathsf{D}_h^c} \lesssim C_{reg}\, h^{k+1}, \tag{6.5a}$$

$$\|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathsf{D}_h^c} \lesssim h\|\mathrm{L}-\mathrm{L}_h\|_{\mathsf{D}_h^c}, \tag{6.5b}$$

$$\|p-p_h\|_{\mathsf{D}_h} \lesssim h^{k+1}\max\Big\{1,(\tau h)^{1/2}\Big\}\big(C_{reg}+|\widetilde{p}|_{H^{k+1}(\Omega)}\big) \quad \textit{and} \tag{6.5c}$$

$$\|p-p_h\|_{\mathsf{D}_h^c} \lesssim h^{k+1}\max\Big\{1,(\tau h)^{1/2}\Big\}\big(C_{reg}+|\widetilde{p}|_{H^{k+1}(\Omega)}\big), \tag{6.5d}$$

where we recall that C_{reg} has been defined in Theorem 4.1.

Proof.

The first two inequalities are obtained from (6.4), Lemma 5.1, estimates in (6.4) and the approximation properties (5.2). Now, by (3.1) and (6.3), we have that

$$\begin{aligned}\|p-p_h\|_{\mathsf{D}_h} &\le \|\widetilde{p}-\widetilde{p}_h\|_{\mathsf{D}_h} + \|\overline{p}^{\mathsf{D}_h} - \overline{p_h}^{\mathsf{D}_h}\|_{\mathsf{D}_h}\\ &\le \|\widetilde{p}-\widetilde{p}_h\|_{\mathsf{D}_h} + |\mathsf{D}_h||\Omega|^{-1}\|\widetilde{p}-\widetilde{p}_h\|_{\mathsf{D}_h^c}.\end{aligned}$$

This inequality, together with Theorem 4.1 and (6.4c) imply (6.5c).

Similarly $\|p-p_h\|_{\mathsf{D}_h^c} \le \|\widetilde{p}-\widetilde{p}_h\|_{\mathsf{D}_h^c} + |\mathsf{D}_h^c||\Omega|^{-1}\|\widetilde{p}-\widetilde{p}_h\|_{\mathsf{D}_h^c}$ and (6.5d) follows from (6.4c). ∎

We remark that, if the solution of the problem is smooth enough, these estimates indicate that the L^2-norm of the error for L_h and p_h converge with optimal order h^{k+1} also in the region D_h^c. Moreover, $\boldsymbol{u}_h$ converges with order h^{k+2}. This additional power of h is due to the the fact that in (6.1) the length of the integration segment is of order h.

7. Numerical results. In this section we present two-dimensional numerical experiments to validate the theoretical orders of convergence of the approximations provided by the HDG method. For $\nu = 1$ and $\tau \equiv 1$, we compute $e_p := \|p-p_h\|_\Omega$, $e_{\boldsymbol{u}} := \|\boldsymbol{u}-\boldsymbol{u}_h\|_\Omega$, $e_{\mathrm{L}} := \|\mathrm{L}-\mathrm{L}_h\|_\Omega$, $e_{\widehat{\boldsymbol{u}}} := \big(\sum_{K\in\mathsf{D}_h} h_K\|\mathbf{P}_M\boldsymbol{u}-\widehat{\boldsymbol{u}}_h\|_{\partial K}\big)^{1/2}$ and $e_{\boldsymbol{u}^*} := \big(\|\boldsymbol{u}-\boldsymbol{u}_h^*\|_{\mathsf{D}_h}^2 + \|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathsf{D}_h^c}^2\big)^{1/2}$. In addition, for each variable, we calculate the experimental order of convergence e.o.c. $= -2\dfrac{\log\left(e_{\mathcal{T}_1}/e_{\mathcal{T}_2}\right)}{\log(N_{\mathcal{T}_1}/N_{\mathcal{T}_2})}$, where $e_{\mathcal{T}_1}$ and $e_{\mathcal{T}_2}$ are the errors associated to the corresponding variable considering two consecutive meshes with $N_{\mathcal{T}_1}$ and $N_{\mathcal{T}_2}$ elements, respectively.

7.1. Example 1: $\mathrm{dist}(\Gamma_h,\Gamma)$ of order h^2. In this first example, we consider the annular domain $\Omega = \{(x,y)\in\mathbb{R}^2 : 0.5^2 < x^2+y^2 < 2^2\}$. The computational boundary Γ_h is constructed by interpolating $\partial\Omega$ by a piecewise linear function and

D_h is the domain enclosed by Γ_h. In this case all the assumptions are satisfied, except (D.3) because the domain is not convex. However, if we assume that the PDE is also valid in the $\mathsf{D}_h \cap \Omega^c$, under minor modification our analysis is also valid.

The source term $\boldsymbol{f}$ and boundary data $\boldsymbol{g}$ are such that the exact solution is

$$p(x,y) = \sin(x)\sin(y), \quad \boldsymbol{u}(x,y) = \begin{bmatrix} \sin(x)\sin(y) \\ \cos(x)\cos(y) \end{bmatrix}.$$

In Table **??** we observe optimal convergence rate for all the variables as Theorems 4.1 and 6.1 predict, i.e., order of h^{k+1} for the errors of e_p, $e_{\boldsymbol{u}}$ and e_{L}; and superconvergence with order h^{k+2} for the numerical trace of the velocity and post-processed solution. Moreover, in Fig. 3 we display the approximation of the second component of $\boldsymbol{u}$ for $k = 1$ and 2 obtained with meshes of sizes $N = 150$ and 2396 elements (first column).

k	N	e_p	order	$e_{\boldsymbol{u}}$	order	e_{L}	order	$e_{\widehat{\boldsymbol{u}}}$	order	$e_{\boldsymbol{u}^*}$	order
1	150	1.83e-02	–	2.01e-02	–	3.76e-02	–	7.68e-03	–	3.21e-03	–
	608	4.38e-03	2.04	5.10e-03	1.96	8.92e-03	2.06	1.04e-03	2.86	4.20e-04	2.91
	2396	1.05e-03	2.08	1.29e-03	2.00	2.23e-03	2.02	1.34e-04	2.99	5.51e-05	2.96
	5842	4.19e-04	2.07	5.32e-04	1.99	8.84e-04	2.07	3.35e-05	3.10	1.39e-05	3.10
	15480	1.57e-04	2.01	2.02e-04	1.98	3.35e-04	1.99	7.98e-06	2.94	3.28e-06	2.96
2	150	1.29e-03	–	1.35e-03	–	1.99e-03	–	2.79e-04	–	1.48e-04	–
	608	1.43e-04	3.14	1.62e-04	3.04	2.43e-04	3.00	1.75e-05	3.95	9.49e-06	3.93
	2396	1.70e-05	3.11	2.07e-05	3.00	3.02e-05	3.04	1.18e-06	3.93	6.05e-07	4.01
	5842	4.33e-06	3.07	5.44e-06	2.99	7.74e-06	3.06	1.99e-07	3.99	1.01e-07	4.01
	15480	1.01e-06	2.98	1.29e-06	2.96	1.82e-06	2.97	2.97e-08	3.91	1.49e-08	3.93
3	150	4.49e-05	–	5.54e-05	–	1.10e-04	–	9.42e-06	–	4.58e-06	–
	608	2.94e-06	3.90	3.80e-06	3.83	6.45e-06	4.05	3.19e-07	4.84	1.50e-07	4.88
	2396	1.81e-07	4.06	2.40e-07	4.03	3.99e-07	4.06	9.85e-09	5.07	5.08e-09	4.94
	5842	2.93e-08	4.09	3.98e-08	4.03	6.46e-08	4.08	1.05e-09	5.03	5.42e-10	5.02
	15480	4.27e-09	3.95	5.83e-09	3.94	9.48e-09	3.94	9.75e-11	4.87	5.01e-11	4.89

Table 1: History of convergence of Example 1.

7.2. Example 2: $\mathrm{dist}(\Gamma_h, \Gamma)$ of order h. We consider $\Omega := \{(x,y) \in \mathbb{R}^2_+ : 1.4 < \sqrt{x^2+y^2} < 2\}$ and the exact solution

$$p(x,y) = e^{x^2+y^2} - (e^{2^2} - e^{1.4^2})/(2^2 - 1.4^2), \quad \boldsymbol{u}(x,y) = \begin{bmatrix} \sin(3x)e^y \\ -3\cos(3x)e^y \end{bmatrix}.$$

The computational D_h is set in such a way that $r_e = 1$ for all $e \in \Gamma_h \setminus \{(x,y) : x = 0 \vee y = 0\}$ as seen in Fig. 4.

Table 2 shows optimal rate of convergence for the error all variables in Ω, that is, order $k+1$ for the errors of the pressure p, the velocity $\boldsymbol{u}$ and the gradient of the velocity L. In addition, the rate of convergence of the errors $e_{\widehat{\boldsymbol{u}}}$ and $e_{\boldsymbol{u}^*}$ seems to be slightly larger than the order $k + 3/2$ predicted by Theorem 4.1.

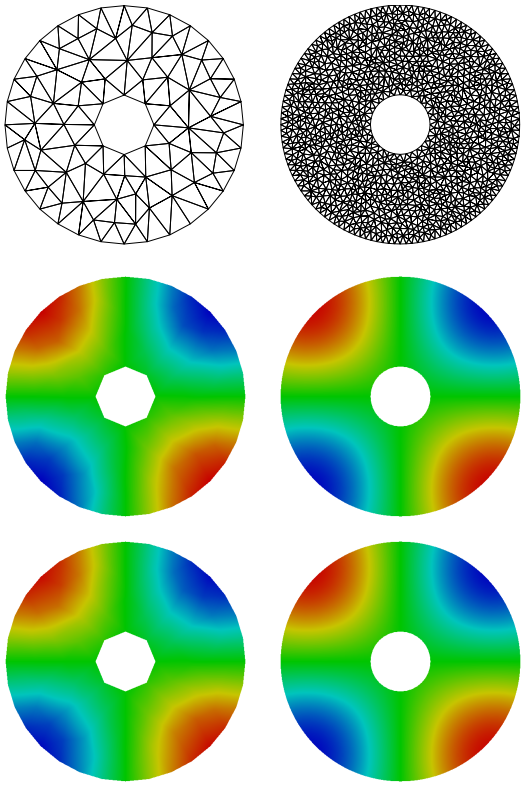

Fig. 3: Approximation of the first component of $\boldsymbol{u}$ in Example 1. Columns: $N = 150$ and 2396. Rows: Polynomial of degree $k = 1$ and 2.

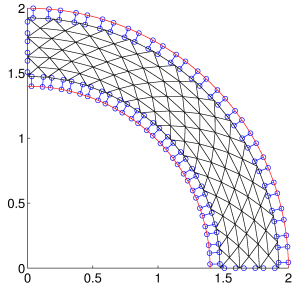

Fig. 4: Domain Ω (red solid line) and D_h in Example 2. Transferring paths (blue solid lines) associated to the quadrature points when $k = 1$.

7.3. Example 3: Other choice of transferring paths. In this last set of examples, we explore the capabilities of the method in a more general setting where some of the assumption are not necessarily satisfied. We consider a kidney-shaped domain whose boundary satisfies the equation $(2[(x + 0.5)^2 + y^2] - x - 0.5)^2 - [(x + 0.5)^2 + y^2] + 0.1 = 0$ and a triangulation of a background domain $\mathcal{B}$ such that $\Omega \subset \mathcal{B}$.

k	N	e_p	order	$e_{\boldsymbol{u}}$	order	e_{L}	order	$e_{\widehat{\boldsymbol{u}}}$	order	$e_{\boldsymbol{u}^*}$	order
1	180	2.35e+00	–	1.13e-01	–	2.29e+00	–	1.96e-01	–	5.46e-02	–
	868	4.20e-01	2.19	2.05e-02	2.17	2.79e-01	2.67	2.62e-02	2.55	7.76e-03	2.48
	3780	9.37e-02	2.04	4.95e-03	1.93	6.90e-02	1.90	3.86e-03	2.61	1.15e-03	2.60
	15748	2.16e-02	2.06	1.24e-03	1.95	1.72e-02	1.95	5.35e-04	2.77	1.61e-04	2.76
	64260	5.05e-03	2.07	3.10e-04	1.96	4.23e-03	1.99	7.18e-05	2.86	2.18e-05	2.85
2	180	1.95e-01	–	5.49e-03	–	7.69e-02	–	1.22e-02	–	3.66e-03	–
	868	2.14e-02	2.81	4.76e-04	3.11	9.84e-03	2.61	1.03e-03	3.14	3.09e-04	3.14
	3780	2.28e-03	3.04	4.78e-05	3.13	1.22e-03	2.83	7.56e-05	3.55	2.28e-05	3.54
	15748	2.48e-04	3.11	5.45e-06	3.04	1.52e-04	2.92	5.20e-06	3.75	1.58e-06	3.74
	64260	2.76e-05	3.13	6.65e-07	2.99	1.87e-05	2.98	3.47e-07	3.85	1.06e-07	3.85
3	180	4.74e-02	–	1.77e-03	–	4.85e-02	–	3.17e-03	–	9.36e-04	–
	868	1.31e-03	4.57	2.02e-05	5.69	5.54e-04	5.69	5.75e-05	5.10	1.72e-05	5.08
	3780	6.97e-05	3.98	7.68e-07	4.45	3.51e-05	3.75	2.11e-06	4.50	6.34e-07	4.49
	15748	3.79e-06	4.08	3.20e-08	4.45	2.21e-06	3.87	7.28e-08	4.72	2.21e-08	4.70

Table 2: History of convergence of Example 2. Errors measured in the Computational domain D_h.

We set D_h as the union of all the elements inside Ω, as it is shown in Fig. 5 (most-left). In this case, the family of transferring paths is constructed by the procedure in Section 2.4.1 of [9]. We point out that now the tangent vector associated to a transferring path is not, in general, normal to a boundary edge. An example is depicted in Fig. 5. In this case, instead of (2.2), for $\boldsymbol{x}$ in a boundary vertex e, we set

$$\widetilde{\boldsymbol{g}}_h(\boldsymbol{x}) := \boldsymbol{g}(\bar{\boldsymbol{x}}) - \int_0^{l(\boldsymbol{x})} \mathrm{L}_h(\boldsymbol{x} + \boldsymbol{t}(\boldsymbol{x})s\,)\boldsymbol{t}(\boldsymbol{x})\,ds, \tag{7.1}$$

where $\boldsymbol{t}(\boldsymbol{x})$ is the unit vector joining $\boldsymbol{x}$ and $\bar{\boldsymbol{x}}$.

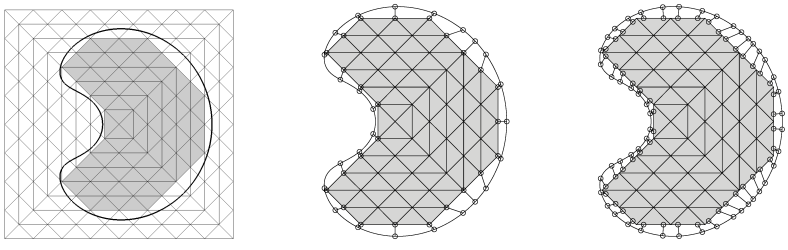

Fig. 5: Left: Example of a domain Ω (kidney-shaped), background domain (square) and polygonal subdomain (gray). Middle: *transferring paths* (segments with starting and ending points marked with $\circ$) associated to boundary vertices. Right: *transferring paths* associated to two points on each boundary edge.

In all the simulations the source term $\boldsymbol{f}$ and boundary data $\boldsymbol{g}$ are such that the

exact solution is

$$p(x,y) = \sin(x^2+y^2) - c_\Omega \quad \text{and} \quad \boldsymbol{u}(x,y) = \begin{bmatrix} \sin(x)\sin(y) \\ \cos(x)\cos(y) \end{bmatrix},$$

where $c_\Omega := \dfrac{1}{|\Omega|}\displaystyle\int_\Omega \sin(x^2+y^2)\,dx\,dy$ was computed numerically considering a extremely fine triangulation that fits the domain.

7.3.1. h-convergence. In Table 3 we observe, for $k=1$, convergence of order $k+1$ for the errors in p, $\boldsymbol{u}$ and L; and order $k+2$ for the errors of the numerical trace and post-processed solution. This agrees with the estimates provided in Theorems 4.1 and 6.1. We recall that Theorem 6.1 predicts an order $h+3/2$ for the numerical trace and post-processed solution, but for an HDG method in polyhedral domains the order of convergence is $k+2$ for these variables. For the case $k=2$, the convergence rate deteriorates in the fourth mesh, but then the optimal order seems to be recovered. When $k=3$, optimal rates are also observed. In the last mesh $e_{\boldsymbol{u}^*}$ and $e_{\widehat{\boldsymbol{u}}}$ seem to be affected by round-off errors. On the other hand, Fig.6 shows the approximation of the first component of the tensor L obtained with $k=2$ and 712 elements. Here we display the computational domain, the approximations in D_h and D_h^c, and the two of them together in Ω.

7.4. k-convergence and condition number. Even though our estimates provide convergence with respect to the meshsize h, we numerically explore the performance of the method when the polynomial degree increases, since the constants in Assumptions A depend on k. Fig. 7 (left) shows the behavior of the log of errors for a fixed mesh with 154 elements. We observe that they linearly decrease for $k=1$, 2, 3 and 4. For $k=5$ they deteriorate but decrease again for $k=6$.

On the other hand, we denote by κ the condition number of the global matrix. For comparison purpose, let κ_{poly} the condition number of the global matrix where the boundary data on Γ_h is being imposed exactly (we can do this since in this example we know the exact solution). We define the ratios $r^\kappa := \kappa/((k+1)^2h^{-2})$ and $r^\kappa_{poly} := \kappa_{poly}/((k+1)^2h^{-2})$. We observe in Fig. 7 (right) that r_{poly} remains constant, which means it behaves like a constant times $(k+1)^2h^{-2}$. However, r^κ increases with k. This fact shows one of the limitations of our method which has been also observed in the Poisson problem [8]. Developing efficient preconditioners is subject of future work. Meanwhile, we solve the system using UMFPACK [12], which is an LU factorization for sparse matrices that is a built-in function in MATLAB.

8. Conclusions. We have proposed and analyzed a high order HDG method for the Stokes equations in a curved domain Ω approximated by a polyhedral subdomain. We showed that if Assumptions A are satisfied, the method is well-posed and provides order of convergence $k+1$ for the pressure, the velocity and its gradient; and order $k+3/2$ for the numerical trace of the velocity and for the element-by-element post-processed velocity. We have also provided numerical evidence validating the error estimates and showing that the method performs optimally even if Assumption A can not be totally guaranteed. One of the advantages of our approach is that the mesh does not need to fit the domain which is convenient for complicated geometries or evolving boundary. In addition, it can be easily coupled to an standard HDG code for polygonal domains and handling the curved boundary reduces to compute line integrals of locally extrapolated polynomials. On the other hand, we think the main

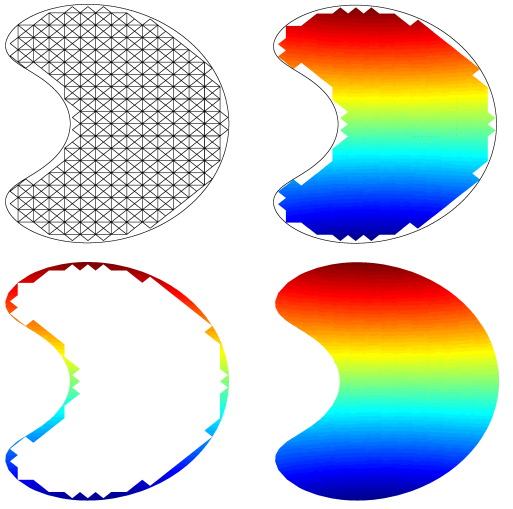

Fig. 6: Approximation of the first component of L in Example 3 for $k = 2$ and 712 elements.

k	N	e_p	order	$e_{\boldsymbol{u}}$	order	e_{L}	order	$e_{\boldsymbol{u}^*}$	order	$e_{\widehat{\boldsymbol{u}}}$	order
1	28	1.82e−02	–	1.24e−03	–	1.10e−02	–	2.20e−03	–	3.10e−03	–
	154	2.78e−03	2.21	2.77e−04	1.75	1.97e−03	2.01	1.83e−03	0.22	3.74e−04	2.48
	712	8.09e−04	1.61	8.35e−05	1.57	4.77e−04	1.85	3.14e−05	5.31	6.71e−05	2.24
	3054	1.41e−04	2.40	2.06e−05	1.92	1.29e−04	1.80	3.24e−06	3.12	7.54e−06	3.00
	12579	3.44e−05	1.99	5.12e−06	1.97	3.31e−05	1.92	3.51e−07	3.14	8.55e−07	3.08
	50877	8.39e−06	2.02	1.28e−06	1.99	8.20e−06	2.00	4.43e−08	2.96	1.09e−07	2.95
2	28	2.21e−03	–	6.20e−05	–	7.75e−04	–	9.84e−05	–	1.18e−04	–
	154	2.25e−04	2.68	7.40e−06	2.49	1.03e−04	2.37	1.21e−05	2.46	2.20e−05	1.97
	712	2.84e−05	2.70	6.18e−07	3.24	1.73e−05	2.33	8.27e−07	3.50	1.73e−06	3.32
	3054	1.88e−05	0.57	1.88e−07	1.63	1.15e−05	0.56	2.38e−07	1.71	5.31e−07	1.62
	12579	8.11e−08	7.69	4.66e−09	5.23	6.97e−08	7.22	7.68e−10	8.11	1.85e−09	8.00
	50877	8.43e−09	3.24	5.82e−10	2.98	7.61e−09	3.17	4.60e−11	4.03	1.12e−10	4.01
3	28	1.33e−02	–	4.74e−04	–	5.33e−03	–	8.67e−04	–	1.06e−03	–
	154	2.06e−05	7.59	5.62e−07	7.90	1.13e−05	7.22	8.40e−07	8.14	1.43e−06	7.75
	712	9.09e−07	4.08	1.50e−08	4.73	5.43e−07	3.97	2.00e−08	4.88	4.04e−08	4.66
	3054	2.53e−08	4.92	3.05e−10	5.35	1.47e−08	4.95	2.98e−10	5.78	6.65e−10	5.64
	12579	1.40e−09	4.10	1.53e−11	4.23	1.13e−09	3.63	1.23e−11	4.51	3.00e−11	4.38
	50877	8.13e−11	4.07	1.36e−12	3.47	6.43e−11	4.10	1.46e−12	3.04	3.53e−12	3.06

Table 3: History of convergence of Example 3.

limitation is the bad conditioning of the linear system when the polynomial degree increases. The development of a suitable preconditioner is subject of future work.

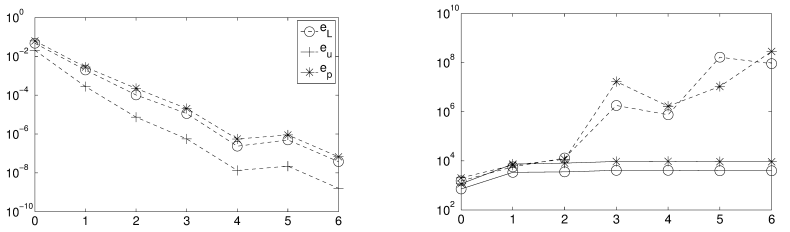

Fig. 7: Left: Errors e_{L} ($\circ$), $e_{\boldsymbol{u}}$ ($+$) and e_p ($*$) versus $k = 0, ..., 6$ in semi-log scale, for 154 elements. Right: Ratios r^{κ} (dashed line) and r^{κ}_{poly} (solid line) in semi-log scale for 28 ($\circ$) and 154 ($*$) elements.

Appendix A. HDG implementation. The HDG method (3.3) can be implemented similarly to the standard HDG method for polygonal domains [23]. The boundary data $\widetilde{\boldsymbol{g}}_h$ defined in (2.2) produces a slight modification in the global system, as we will see. We first construct the local matrices associated to an element $K \in \mathsf{T}_h$ and the boundary data will be imposed after the global matrix is assembled. To fix ideas we briefly describe the two dimensional case. Let $K \in \mathsf{T}_h$ with edges e_1, e_2 and e_3, $N_{\mathrm{L}} = \dim(\mathrm{P}_k(K))$, $N_{\boldsymbol{u}} = \dim(\boldsymbol{\mathcal{P}}_k(K))$, $N_p = \dim(\mathcal{P}_k(K))$ and $N_{\widehat{\boldsymbol{u}}} = \dim(\boldsymbol{\mathcal{P}}_k(e_1)) = \dim(\boldsymbol{\mathcal{P}}_k(e_2)) = \dim(\boldsymbol{\mathcal{P}}_k(e_3))$. Let $\{\Psi_j\}_{j=1}^{N_{\mathrm{L}}}$, $\{\boldsymbol{\psi}_j\}_{j=1}^{N_{\boldsymbol{u}}}$, $\{\psi_j\}_{j=1}^{N_p}$ and $\{\boldsymbol{\xi}_j^{e_l}\}_{j=1}^{N_{\widehat{\boldsymbol{u}}}}$ ($l = 1$, 2 and 3) the basis function of $\mathrm{P}_k(K)$, $\boldsymbol{\mathcal{P}}_k(K)$, $\mathcal{P}_k(K)$ and $\mathcal{P}_k(e_l)$, respectively. In our numerical experiments we considered Dubiner basis [18] for the polynomial spaces in K and Legendre basis for the polynomial space on the edges. We write now the unknowns restricted to K as linear combination of the basis functions, i.e., $\mathrm{L}_h|_K = \sum_{j=1}^{N_{\mathrm{L}}} \alpha_j^K \Psi_j$, $\boldsymbol{u}_h|_K = \sum_{j=1}^{N_{\boldsymbol{u}}} \beta_j^K \boldsymbol{\psi}_j$, $\widetilde{p}_h|_K = \sum_{j=1}^{N_p} \gamma_j^K \psi_j$ and $\widehat{\boldsymbol{u}}_h|_{e_l} = \sum_{j=1}^{N_{\widehat{\boldsymbol{u}}}} \lambda_j^{e_l} \boldsymbol{\xi}_j^{e_l}$. We decompose $\widetilde{p}_h|_K = \overline{p_h}^K + \rho_h$, where $\overline{p_h}^K \in \mathcal{P}_0(K)$ and $\rho_h \in \mathcal{P}_k(K) \cap L_0^2(K)$. Since Dubiner basis are orthogonal on K and ψ_1 is a constant, then $\rho_h = \sum_{j=2}^{N_p} \gamma_j^K \psi_j$. Thus, the system associated to (3.3a)-(3.3c) is of the form

$$\text{(A.1)}\quad \mathbf{A}^K \begin{bmatrix} \boldsymbol{\alpha}^K \\ \boldsymbol{\beta}^K \\ \boldsymbol{\gamma}^K \end{bmatrix} = \mathbf{E}^K \begin{bmatrix} \boldsymbol{\lambda}^{e_1} \\ \boldsymbol{\lambda}^{e_2} \\ \boldsymbol{\lambda}^{e_3} \end{bmatrix} + \mathbf{F}^K, \quad \text{where } \mathbf{A}^K \in \mathbb{R}^{N_T \times N_T}, \mathbf{E}^K \in \mathbb{R}^{N_T \times 3N_{\widehat{\boldsymbol{u}}}},$$

$\mathbf{F}^K \in \mathbb{R}^{N_T}$ and $N_T := N_{\mathrm{L}} + N_{\boldsymbol{u}} + N_p$. On the other hand, the local contribution of the global equation (3.3g) can be written in the following matrix form

$$\text{(A.2)}\quad \mathbf{B}^K \begin{bmatrix} \boldsymbol{\alpha}^K \\ \boldsymbol{\beta}^K \\ \boldsymbol{\gamma}^K \end{bmatrix} - \mathbf{H}^K \overline{p_h}^K = \boldsymbol{\Lambda}^K \begin{bmatrix} \boldsymbol{\lambda}^{e_1} \\ \boldsymbol{\lambda}^{e_2} \\ \boldsymbol{\lambda}^{e_3} \end{bmatrix}, \text{ where } \mathbf{B}^K \in \mathbb{R}^{3N_{\widehat{\boldsymbol{u}}} \times N_T}, \mathbf{H}^K \in \mathbb{R}^{3N_{\widehat{\boldsymbol{u}}}}$$

and $\boldsymbol{\Lambda}^K \in \mathbb{R}^{3N_{\widehat{\boldsymbol{u}}} \times 3N_{\widehat{\boldsymbol{u}}}}$. Since the matrix on the left hand side of (A.1) is invertible, we express the unknowns $\boldsymbol{\alpha}^K$, $\boldsymbol{\beta}^K$ and $\boldsymbol{\gamma}^K$ in terms of $\boldsymbol{\lambda}^{e_l}$ and substitute them in (A.2) to obtain a local system of the form

$$\text{(A.3)}\quad \mathbf{M}^K \begin{bmatrix} \boldsymbol{\lambda}^{e_1} \\ \boldsymbol{\lambda}^{e_2} \\ \boldsymbol{\lambda}^{e_3} \end{bmatrix} + \mathbf{H}^K \overline{p_h}^K = \mathbf{b}^K, \quad \text{where } \mathbf{M}^K \in \mathbb{R}^{N_{\widehat{\boldsymbol{u}}} \times N_{\widehat{\boldsymbol{u}}}} \text{ and } \mathbf{b}^K \in \mathbb{R}^{N_{\widehat{\boldsymbol{u}}}}.$$

Now, let N_e and N_K be the number of edges and elements of the triangulation. We denote by $\boldsymbol{\lambda}^l$ ($l = 1, \ldots, N_e$) the global unknowns associated to the l-th edge. We define the global vectors $\boldsymbol{\lambda} = [(\boldsymbol{\lambda}^1)^t, ..., (\boldsymbol{\lambda}^{N_e})^t]^t$ and $\boldsymbol{\sigma} = [\overline{p}_h^{K_1}, ..., \overline{p}_h^{K_{N_K}}]^t$. Thus, once the above local matrices are computed, we assembly on the edges of the triangulation and obtain the global system of the form

$$\begin{bmatrix}\mathbf{M} & \mathbf{H}\end{bmatrix}\begin{bmatrix}\boldsymbol{\lambda}\\ \boldsymbol{\sigma}\end{bmatrix} = \mathbf{b}, \quad \text{where} \quad \mathbf{M} \in \mathbb{R}^{(N_e N_{\hat{\boldsymbol{u}}})\times(N_e N_{\hat{\boldsymbol{u}}})}, \mathbf{H} \in \mathbb{R}^{(N_e N_{\hat{\boldsymbol{u}}})\times N_K} \tag{A.4}$$

and $\mathbf{b} \in \mathbb{R}^{(N_e N_{\hat{\boldsymbol{u}}})}$. In addition, equation (3.3f) can be written as $[|K_1| \quad ... \quad |K_{N_K}|]\boldsymbol{\sigma} = 0$, which, together with (A.4) and the boundary condition, fully determines $\boldsymbol{\lambda}$ and $\boldsymbol{\sigma}$.

We proceed now to impose the boundary condition. Let $e \in \mathcal{E}_h^{\partial}$ and K^e the triangle where it belongs. We assume that the only boundary edge of K is e and denote by e_1 and e_2 the other two edges. If K had two boundary edges the procedure is analogous. From (3.3d) and (7.1) we have that

$$\boldsymbol{\Lambda}^e \boldsymbol{\lambda}^e = \mathbf{b}^{\boldsymbol{g}} - \mathbf{J}^e \boldsymbol{\alpha}^{K^e},$$

where

$$\mathbf{b}^{\boldsymbol{g}} := \left\{ \int_e \int_0^{l(\boldsymbol{x})} \boldsymbol{g}(\bar{\boldsymbol{x}}(\boldsymbol{x})) \cdot \boldsymbol{\xi}_i^e(\boldsymbol{x})\, ds\, d\boldsymbol{x} \right\}_{i=1}^{N_{\hat{\boldsymbol{u}}}}$$

and

$$\mathbf{J}^e := \left\{ \int_e \int_0^{l(\boldsymbol{x})} \boldsymbol{\Psi}_j(\boldsymbol{x} + s\boldsymbol{t}(\boldsymbol{x}))\boldsymbol{t}(\boldsymbol{x}) \cdot \boldsymbol{\xi}_i^e(\boldsymbol{x})\, ds\, d\boldsymbol{x} \right\}_{i=1}^{N_{\hat{\boldsymbol{u}}}}.$$

Here, we see how the basis functions $\boldsymbol{\Psi}_j$ (defined in K) are being evaluated (extrapolated) along the path $\boldsymbol{x} + s\boldsymbol{t}(\boldsymbol{x})$. From (A.1) we write $\boldsymbol{\alpha}^K$ in terms of $\boldsymbol{\lambda}^{e_1}$, $\boldsymbol{\lambda}^{e_2}$ and $\boldsymbol{\lambda}^e$, say $\boldsymbol{\alpha}^K = \mathbf{K}\begin{bmatrix}\boldsymbol{\lambda}^{e_1}\\ \boldsymbol{\lambda}^{e_2}\\ \boldsymbol{\lambda}^e\end{bmatrix} + \mathbf{r}$, and obtain the system

$$\left[\begin{bmatrix}\boldsymbol{\Theta}_{N_{\hat{\boldsymbol{u}}}\times 3N_{\hat{\boldsymbol{u}}}} & \boldsymbol{\Theta}_{N_{\hat{\boldsymbol{u}}}\times 3N_{\hat{\boldsymbol{u}}}} & \boldsymbol{\Lambda}^e\end{bmatrix} + \mathbf{J}^e\mathbf{K}\right]\begin{bmatrix}\boldsymbol{\lambda}^{e_1}\\ \boldsymbol{\lambda}^{e_2}\\ \boldsymbol{\lambda}^e\end{bmatrix} = \mathbf{b}^{\boldsymbol{g}} - \mathbf{J}^e\mathbf{r}. \tag{A.5}$$

Finally, we substitute the rows in (A.4) associated to the global degrees of freedom of the edge e by the equations in (A.5). This resulting global system is not symmetric (unless $\mathbf{J}^e = \boldsymbol{\Theta}$). If Ω were polygonal, then $\mathbf{J}^e = \boldsymbol{\Theta}$ and the (A.5) becomes $\boldsymbol{\Lambda}^e\boldsymbol{\lambda}^e = \mathbf{b}^{\boldsymbol{g}}$ which is one of the ways to impose Dirichlet boundary conditions.

Appendix B. Integration over curved regions. In order to compute the approximation $\overline{p_h}^{\mathsf{D}_h}$ and errors in D_h^c, integrals over $\widetilde{K}_{ext}^e$ must be calculated. We write $\widetilde{K}_{ext}^e$ as a union of three disjoint regions T_1, T_2 and S as Fig. 8 shows. The same quadrature rules considered in the calculation of the local matrices of the method are used now in the triangles T_1 and T_2. Now, let e_S and Γ_S denote the straight and curved part of the boundary of S, resp. For a point $\boldsymbol{x}$ in Γ_h, we construct $\sigma_{\boldsymbol{n}_S}(\boldsymbol{x})$, where $\boldsymbol{n}_S$ is the unit normal of e_S pointing to Γ_S. Thus, for a smooth enough function q defined on S, we approximate

$$\int_S q(\boldsymbol{x})d\boldsymbol{x} = \int_{e_S}\int_0^{l(\boldsymbol{x})} q(\boldsymbol{x} + t\boldsymbol{n}_S)\boldsymbol{n}_S\, dt\, d\boldsymbol{x}$$

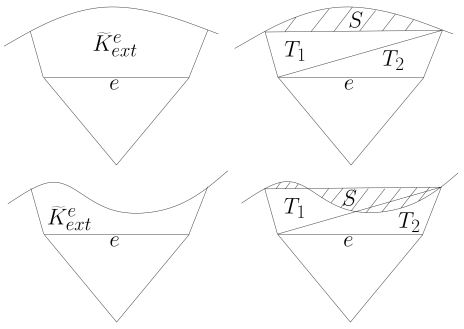

Fig. 8: Examples of $\widetilde{K}^e_{ext}$ divided in triangles T_1 and T_2, and a region S (dashed area).

using one-dimensional quadrature on both integrals.

REFERENCES

[1] Barrett. J. W. and Elliott, C. M. (1987) Fitted and Unfitted Finite-Element Methods for Elliptic Equations with Smooth Interfaces, IMA J. Numer. Anal. **7** 1987 283–300.

[2] Bramble, J. H., Dupont, T. and Thomée, V. (1994) Projection methods for Dirichlet's problem in Approximating polygonal domains with boundary-value corrections, Math. Comp., **26** (120), pp. 869–879.

[3] Bramble, J. H. and King, J. T. (1994) A robust finite element method for nonhomogeneous Dirichlet problems in domains with curved boundaries, Math. Comp., **63**, 1–17.

[4] Bramble, J. H. and King, J. T. (1996) A finite element method for interface problems in domains with smooth boundaries and interfaces. Advances in Comp. Math. **6**, 109–138.

[5] Cockburn, B., Gopalakrishnan, J., Nguyen, N. C., Peraire, J. (2010) Analysis of HDG methods for Stokes flow. Math. of Comp., **80** (274), 723–760.

[6] Cockburn, B., Gopalakrishnan, J. and Sayas, F.-J. (2010) A projection-based error analysis of HDG methods. Math. Comp., **79**, 1351–1367.

[7] Cockburn, B., Gupta, D. and Reitich, F. (2010) Boundary-conforming discontinuous Galerkin methods via extensions from subdomains. J. Sci. Comput., **42**, 144–184.
second-order elliptic problems. Math. Comp., **78**, 1–24.

[8] Cockburn, B., Qiu, W. and Solano, M. (2014) A priori error analysis for HDG methods using extensions from subdomains to achieve boundary-conformity. Math. of Comp. **83**, 665–699.

[9] Cockburn, B. and Solano, M. (2012) Solving Dirichlet boundary-value problems on curved domains by extensions from subdomains. SIAM J. Sci. Comput. **34**, A497–A519.

[10] Cockburn, B. and Solano, M. (2014) Solving convection-diffusion problems on curved domains by extensions from subdomain. J. Sci Comput. **59** (2), 512–543.

[11] Dauge, M. (1989) Stationary Stokes and Navier-Stokes systems on two- or three-dimensional domains with corners. I. Linearized equations. SIAM J. Math. Anal., **20** (1), 74–97.

[12] Davis, T. A. (2004). Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method, ACM Trans. Math. Softw., **30**:2, 196–199.

[13] Deckelnick, K., Günther, A. and Hinze M. (2009) Finite Element Approximation of Dirichlet Boundary Control for Elliptic PDEs on Two- and Three-Dimensional Curved Domains. SIAM J. Control Optim, **48**:4, 2798–2819.

[14] Dong, H., Wang, B. and Xie, Z. (2017) An unfitted hybridizable discontinuous Galerkin method for the Poisson interface problem and its error analysis, IMA J. Numer. Anal. **37**:1, 444–476

[15] Ern, A. and Di Pietro, DA (2012) Mathematical Aspects of discontinuous Galerkin methods, In: Mathématiques et Applications (Berlin) **69**. Springer, Heidelberg.

[16] Guzman, J., Sanchez, M.A. and Sarkis M. (2016) Higher-order finite element methods for elliptic problems with interfaces, **50**:5, 1561–1583.

[17] Kellogg, R. B. and Osborn, J. E. (1976) A regularity result for the Stokes problem in a convex polygon, J. Funct. Anal.m **21** (4), 397–431.

[18] Kirby, R. A. (2010) Singularity-free evaluation of collapsed–coordinate orthogonal polynomials, ACM Trans. Math. Softw. **37**, 1, Article No. 5.

[19] LeVeque R.J. and Li, Z. (1997) Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sic. Comput. **18** (3) 709–735.
[20] Lenoir, M. (1986) Optimal isoparametric finite elements and errors estimates for domains involving curved boundaries. SIAM J. Numer. Anal., **23**, 562–580.
[21] Li, J., Melenk, M., Wohlmuth, B. and Zou, J. (2010) Optimal convergence of higher order finite element methods for elliptic interface problems. Appl. Numer. Math. **60**, 19–37.
[22] Mori, Y. (2008) Convergence proof of the velocity field for a Stokes flow immersed boundary method. Comm. Pure and Appl. Math., **160**, pp. 1213–1263.
equations. J. Comput. Phys. **228**, 3232–3254.
[23] Nguyen, N. C., Peraire, J. and Cockburn, B. (2010) A hybridizable discontinuous Galerkin method for Stokes flow. Comput. Methods Appl. Mech. Engrg., **199**, 582–597.
[24] Nitsche, J. (1970–71) Über ein Variationsprinzip zur LBsung van Dirichlet–Problemen bei Verwendung van Teilrümen, die keinen Handbedingungen unterworfen sind, Abh. Math. Sem. Univ. Hamburg **36**
[25] Peskin, C.S. (1972) Flow patterns around heart valves: a numerical method. J. Comput. Phys., **10**, 252–271.
[26] Peterseim, D. and Sauter, S. A. (2008) The Composite Mini Element–Coarse Mesh Computation of Stokes Flows on Complicated Domains. SIAM J. Numer. Anal. **46**:6, 3181–3206.
[27] Peterseim, D. and Sauter, S. A. (2011) Finite element methods for the Stokes problem on complicated domains. Comput. Methods Appl. Mech. Engrg. **200** 2611–2623.
[28] Qiu, W., Solano, M. and Vega, P. (2016) A high order HDG method for curved-interface problems via approximations from straight triangulations., J. Sci Comput. **69**, 3, 1384–1407.
[29] Thomée, V. (1973) Polygonal domain approximation in Dirichlet's problem, J. Inst. Math. Appl., **11**, 33–44.

Centro de Investigación en Ingeniería Matemática (CI^{2}MA)

PRE-PUBLICACIONES 2016

2016-01 Daniele Boffi, Lucia Gastaldi, Rodolfo Rodríguez, Ivana Sebestova: *Residual-based a posteriori error estimation for the Maxwell eigenvalue problem*

2016-02 Weifeng Qiu, Manuel Solano, Patrick Vega: *A high order HDG method for curved-interface problems via approximations from straight triangulations*

2016-03 Gabriel N. Gatica: *A note on stable Helmholtz decompositions in 3D*

2016-04 Julio Aracena, Adrien Richard, Lilian Salinas: *Number of fixed points and disjoint cycles in monotone Boolean networks*

2016-05 Sophie Balemans, Raimund Bürger, Stefan Diehl, Pierre Francois, Julien Laurent, Florent Locatelli, Ingmar Nopens, Elena Torfs: *On constitutive functions for hindered settling velocity in 1-D settler models: selection of appropriate model structure*

2016-06 Anahi Gajardo, Vincent Nesme, Guillaume Theyssier: *Pre-expansivity in cellular automata*

2016-07 Eligio Colmenares, Michael Neilan: *Dual-mixed finite element methods for the stationary Boussinesq problem*

2016-08 Veronica Anaya, David Mora, Carlos Reales, Ricardo Ruiz-Baier: *Finite element methods for a stream-function – vorticity formulation of the axisymmetric Brinkman equations*

2016-09 Rommel Bustinza, Bibiana López-Rodríguez, Mauricio Osorio: *An a priori error analysis of an HDG method for an eddy current problem*

2016-10 Luis M. Castro, Víctor H. Lachos, Tsung-I Lin, Larissa A. Matos: *Heavy-tailed longitudinal regression models for censored data: A likelihood based perspective*

2016-11 Gabriel N. Gatica, Ricardo Ruiz-Baier, Giordano Tierra: *A posteriori error analysis of an augmented mixed method for the Navier-Stokes equations with nonlinear viscosity*

2016-12 Manuel Solano, Felipe Vargas: *A high order HDG method for Stokes flow in curved domains*

Para obtener copias de las Pre-Publicaciones, escribir o llamar a: Director, Centro de Investigación en Ingeniería Matemática, Universidad de Concepción, Casilla 160-C, Concepción, Chile, Tel.: 41-2661324, o bien, visitar la página web del centro: `http://www.ci2ma.udec.cl`







Centro de Investigación en
Ingeniería Matemática (CI²MA)
Universidad de Concepción

Casilla 160-C, Concepción, Chile
Tel.: 56-41-2661324/2661554/2661316
http://www.ci2ma.udec.cl