

A symmetric method for weakly imposing Dirichlet boundary conditions in embedded finite element meshes

Joan Baiges^{1,2,*}, Ramon Codina^{1,2}, Florian Henke^{3,4}, Shadan Shahmiri⁴ and Wolfgang A. Wall⁴

¹Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain

²Centre Internacional de Mètodes Numèrics a l'Enginyeria (CIMNE), Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain

³Emmy Noether Research Group "Computational Multiscale Methods for Turbulent Combustion"

⁴Technische Universität München/Boltzmannstrasse 15, D-85748 Garching b. München /Germany

SUMMARY

In this paper, we propose a way to weakly prescribe Dirichlet boundary conditions in embedded finite element meshes. The key feature of the method is that the algorithmic parameter of the formulation which allows to ensure stability is independent of the numerical approximation, relatively small, and can be fixed *a priori*. Moreover, the formulation is symmetric for symmetric problems. An additional element-discontinuous stress field is used to enforce the boundary conditions in the Poisson problem. Additional terms are required in order to guarantee stability in the convection–diffusion equation and the Stokes problem. The proposed method is then easily extended to the transient Navier–Stokes equations. Copyright © 2012 John Wiley & Sons, Ltd.

Received 16 May 2011; Revised 16 September 2011; Accepted 18 September 2011

KEY WORDS: finite element methods; boundary conditions; fixed meshes; Nitsche's method

1. INTRODUCTION

In this paper, we propose a new method for weakly imposing Dirichlet boundary conditions in embedded grids. The numerical approximation of boundary value problems on nonmatching grids has the obvious advantage of the freedom to generate the grid. Only a grid covering the computational domain has to be created, leaving the imposition of boundary conditions to the numerical formulation being used. Moreover, in time evolving domains, embedded grids avoid the need to remesh when the domain undergoes very large deformations, which is a drawback of the classical Arbitrary Lagrangian–Eulerian (ALE) approaches. See [1–7] for some examples of the applications of fixed-mesh formulations in time evolving domains. In contrast, the boundary of the physical domain does not coincide with the boundary of the mesh. Thus, boundary conditions need to be imposed in this *immersed boundary*. Although Neumann boundary conditions can be easily dealt with, some care is needed with Dirichlet boundary conditions.

Let us describe the problem to be solved. Consider the situation depicted in Figure 1. A domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, with boundary $\Gamma = \partial\Omega$ (red curve in Figure 1), is covered by a mesh that occupies a domain $\Omega_h = \Omega_{in} \cup \Omega_\Gamma$, where $\Omega_{in} \subset \Omega$ is formed by the elements interior to Ω and Ω_Γ is formed by a set of elements cut by Γ . In turn, let us split $\Omega_\Gamma = \Omega_{\Gamma,in} \cup \Omega_{\Gamma,out}$, where $\Omega_{\Gamma,in} = \Omega \cap \Omega_\Gamma$ and $\Omega_{\Gamma,out}$ is the interior of $\Omega_\Gamma \setminus \Omega_{\Gamma,in}$. Note that $\Omega = \Omega_{in} \cup \Omega_{\Gamma,in}$. For simplicity,

*Correspondence to: Joan Baiges, Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain.

†E-mail: jbaiges@cimne.upc.edu

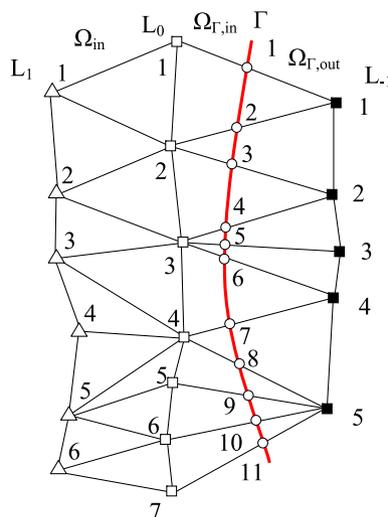


Figure 1. Setting.

we will assume that the intersection of Γ with the element domains is a piecewise polynomial curve (in 2D) or surface (in 3D) of the same order as the finite element interpolation. The problem we are facing is how to impose Dirichlet boundary conditions on the immersed boundary Γ .

Several strategies to impose Dirichlet boundary conditions in nonmatching grids have been developed. The widely used *immersed boundary method* in its original form [8] consists of adding point-wise penalty forces in the domain boundary so that the boundary conditions are fulfilled. The method is first-order accurate even if second-order approximation schemes are used, although formal second-order accuracy has been reported in [9]. The more recent *immersed interface method* achieves higher order accuracy by avoiding the use of the Dirac delta distribution to define the forcing terms (see [10–12]).

Another approach is the use of Lagrange multipliers to enforce the boundary conditions. However, the finite-element subspaces for the bulk and Lagrange multiplier fields must satisfy the classical inf-sup condition proposed by Babuška [13], which usually leads to the need for stabilization (see [14–16]). Moreover, additional DOF must be added to the problem. The use of Lagrange multipliers is the basis of the *fictitious domain method* [17, 18] (see also Chapter VIII in [19]). The use of stabilized Lagrange multiplier techniques in the fictitious domain method has been analyzed in [20].

Methods which use the grid nodes closest to the boundary to enforce boundary conditions have been developed in [21–23], for example. In [24], a method for strongly imposing Dirichlet boundary conditions in immersed boundary methods is proposed. The method is second-order accurate, does not require of user-defined parameters, and it guarantees that the difference between the solution and prescribed value on the boundary is minimized. However, it is nonsymmetric even for symmetric problems.

A *DG-based immersed boundary method* is proposed in [25], which consists of switching elements intersected by the boundary to a DG approximation and imposing the Dirichlet boundary conditions strongly. Although optimal-order accuracy is achieved, the method requires additional DOF.

Finally, several variations of Nitsche’s method can be found in the literature [26–28]. These methods are symmetric for symmetric problems and do not need additional DOF to impose boundary conditions. However, a user-defined stabilization parameter is required. Choosing this stabilization parameter is not straightforward: if the parameter is not large enough, the problem becomes unstable; if it is too large, the resulting system of equations becomes ill-conditioned. This drawback can be addressed by using the inverse estimates in order to define the minimum value for the stabilization parameter (see [29], where the stabilization parameter for the heat transfer problem is

studied). However, there are still some nondimensional constants to be defined in the inverse estimates, and it remains to be seen how to apply the method to nonsymmetric problems such as the convection–diffusion equation.

A list of desired properties for our strategy for imposing Dirichlet boundary conditions in nonmatching grids can be extracted from the previously described methods

- No additional DOF should be needed in the final system of equations in order to enforce boundary conditions.
- The method should be free of user-defined penalty or stabilization parameters which might ill-condition the resulting system of equations.
- The resulting variational form should be symmetric for symmetric problems but also capable of dealing with flow problems such as the convection–diffusion or the Navier–Stokes equations.
- A rate of convergence as close as possible to optimal should be observed when applying the method to the problems of interest.

As we will see, the method developed in this work fulfills the three first requirements of the previous list. Regarding the last point, no convergence proof is presented here, but convergence order is tested in the numerical examples section with quite satisfactory results.

The starting point of the strategy we propose is the method presented in [30]. This method imposes Dirichlet boundary conditions weakly but does not require any user-defined stabilization or penalty parameter (as we shall see, the parameter on which the formulation depends can be set *a priori* and independently of the finite element mesh). In order to do so, a hybrid formulation which introduces an additional element-wise discontinuous flux field is used. However, this additional flux field is only required in the elements which are cut by the immersed boundary, and because it is discontinuous across interelement boundaries, it can be condensed prior to solving the resulting system of equations. The method shows optimal order of convergence and satisfies the design condition of not needing additional DOF in order to impose boundary conditions. However, it is nonsymmetric even for symmetric problems.

In Section 2, a symmetric version of the method proposed in [30] for Poisson's problem is presented. The main idea is again to use a hybrid formulation with an additional element-wise discontinuous flux field. However, some additional terms are added so that the method is symmetric. A stability analysis is performed in order to ensure that the method is stable under a certain condition on the parameter on which it depends, and therefore, there is no need to tune it by the user. This condition happens to be independent of the finite element mesh. In Section 3, we extend the method to the convection–diffusion equation. Additional terms are required to further enforce boundary conditions in order to guarantee the stability of the method in the case of convection-dominated flows. The stability analysis shows that boundary conditions can be given a different treatment in the inflow and outflow boundary, which justifies the chosen weighting term for the boundary conditions enforcement. In Section 4, we deal with the treatment of boundary conditions in the case of the Stokes problem, and the stability of the proposed method for this particular problem is shown. Additional terms which enforce the velocity in the direction normal to the immersed boundary are required to keep the symmetry of the problem. Finally, in Section 5, we put together the terms which define our method for the convection–diffusion equation and the Stokes problem, and we describe the strategy to impose boundary conditions in the transient incompressible Navier–Stokes equations. Numerical examples illustrate the behavior of the proposed method in a number of situations in Section 6, and some conclusions close the paper in Section 7.

2. A SYMMETRIC METHOD FOR POISSON'S PROBLEM

In this section, a symmetric method for imposing boundary conditions for Poisson's problem is presented. In the following sections, the method will be extended to other symmetric and nonsymmetric problems.

2.1. Problem statement

Let us consider the problem of finding $u : \Omega \rightarrow \mathbb{R}$ such that

$$\begin{aligned} -k\Delta u &= f && \text{in } \Omega, \\ u &= \bar{u} && \text{on } \Gamma = \partial\Omega, \end{aligned}$$

where $k > 0$, f is a given forcing function, and \bar{u} is the given Dirichlet boundary condition. We assume that the subdomain Ω is covered by the domain Ω_h , as explained in Section 1.

We can now consider a two-field formulation in which we introduce an additional flux unknown σ to the previous problem. The problem can now be written as

$$-k\Delta u = f \quad \text{in } \Omega, \tag{1}$$

$$\frac{1}{k}\sigma = \nabla u \quad \text{in } \Omega_{\Gamma,\text{in}}, \tag{2}$$

$$u = \bar{u} \quad \text{on } \Gamma = \partial\Omega. \tag{3}$$

2.2. Discrete weak form

Let $\mathcal{P}_h = \{K\}$ be a finite element partition of Ω_h from which we construct the finite element space $V_h \subset H^1(\Omega_h)$ (we will consider V_h made of continuous functions) and $S_h \subset L^2(\Omega_h)^d$. Functions in S_h will be considered *piecewise discontinuous in the elements cut by Γ and zero elsewhere*.

Our symmetric variational form of the problem consists of finding $u_h \in V_h$ and $\sigma_h \in S_h$ such that

$$k(\nabla u_h, \nabla v_h) - \langle \sigma_h \cdot \mathbf{n}, v_h \rangle_\Gamma + \frac{1}{n}(\nabla v_h, \sigma_h)_{\Omega_{\Gamma,\text{in}}} - \frac{1}{n}k(\nabla v_h, \nabla u_h)_{\Omega_{\Gamma,\text{in}}} = \langle f v_h \rangle_\Omega, \quad \forall v_h \in V_h, \tag{4}$$

$$-\frac{1}{nk}(\boldsymbol{\tau}_h, \sigma_h)_{\Omega_{\Gamma,\text{in}}} + \frac{1}{n}(\boldsymbol{\tau}_h, \nabla u_h)_{\Omega_{\Gamma,\text{in}}} - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, u_h \rangle_\Gamma = -\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{u} \rangle_\Gamma, \quad \forall \boldsymbol{\tau}_h \in S_h, \tag{5}$$

where n is a free parameter for which we will propose an expression in the following sections. Here and below, (\cdot, \cdot) denotes the L^2 product in Ω . In general, the integral of two functions g_1 and g_2 over a domain ω will be denoted by $\langle g_1, g_2 \rangle_\omega$, the $L^2(\omega)$ inner product by $(\cdot, \cdot)_\omega$, and the norm in a function space X by $\|\cdot\|_X$, with the simplifications $\|\cdot\|_{L^2(\Omega)} \equiv \|\cdot\|$ and $(\cdot, \cdot)_\Omega \equiv (\cdot, \cdot)$.

It is readily seen that problems (4)–(5) correspond to the discrete version of the optimality conditions of the functional

$$F(u, \sigma) := \frac{1}{2}k\|\nabla u\|^2 - \langle f, u \rangle_\Omega - \langle \sigma \cdot \mathbf{n}, u - \bar{u} \rangle_\Gamma - \frac{1}{2nk}\|\sigma - k\nabla u\|_{L^2(\Omega_{\Gamma,\text{in}})}^2.$$

The first two terms correspond to the functional associated to the Poisson problem, whereas the third yields the Lagrangian obtained by imposing the boundary condition, $\lambda = \sigma \cdot \mathbf{n}$ being the Lagrange multiplier. The last term imposes in a least squares sense that $\sigma = k\nabla u$. Therefore, a possible interpretation of the method we propose is that *it is a Lagrange multiplier technique with the condition that the multiplier is the normal trace of the flux of the unknown in a least squares sense*. This additional condition has important consequences at the discrete level, because we will be able to eliminate σ_h and end up with a problem posed for u_h alone, as we shall see. Contrary to the classical Lagrange multiplier technique, no additional DOF will be needed in the final system of equations to be solved.

Note that for the exact solution u, σ of problems (1)–(3), there holds

$$k(\nabla u, \nabla v_h) - \langle \sigma \cdot \mathbf{n}, v_h \rangle_\Gamma = \langle f, v_h \rangle_\Omega, \tag{6}$$

$$\frac{1}{n}(\nabla v_h, \sigma)_{\Omega_{\Gamma,\text{in}}} - \frac{1}{n}k(\nabla v_h, \nabla u)_{\Omega_{\Gamma,\text{in}}} = 0, \tag{7}$$

$$-\frac{1}{nk}(\boldsymbol{\tau}_h, \sigma)_{\Omega_{\Gamma,\text{in}}} + \frac{1}{n}(\boldsymbol{\tau}_h, \nabla u)_{\Omega_{\Gamma,\text{in}}} = 0, \tag{8}$$

$$-\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, u \rangle_\Gamma = -\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{u} \rangle_\Gamma. \tag{9}$$

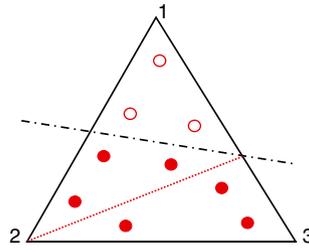


Figure 2. Splitting of elements.

Equation (6) is weakly enforcing (1) tested against v_h ; (7) and (8) are weakly enforcing (2) tested against $\frac{1}{n}\nabla v_h$ and $-\frac{1}{n}\boldsymbol{\tau}_h$, respectively, and (9) is weakly enforcing (3) tested against $-\boldsymbol{\tau}_h \cdot \mathbf{n}$. Problems (4)–(5) is obtained by adding up (6)–(9) and replacing the continuous solution by the approximate one. Note that the main difference between the presented method and the method described in [30] when applied to Poisson’s problem are the terms coming from (8), which do not appear in [30] and are the terms which make the presented method symmetric.

Let us finally remark that the volume integrals in (4)–(5) are performed over $\Omega = \Omega_{\text{in}} \cup \Omega_{\Gamma,\text{in}}$. This means that elements which are cut by the boundary of the domain need to be split for integration purposes. In the case of 2D linear elements, Figure 2 shows how that the splitting can be done and the numerical integration points (red points) required in each triangle resulting from this splitting. The DOF of the problem to be solved continue to be the ones of the original mesh, because the splitting is only for integration purposes. See [31] for an extensive description on how to perform numerical subintegration in 3D high-order finite elements.

2.3. Stability

In this subsection, we prove that the formulation given by (4)–(5) is stable and as a consequence has a unique solution. We define the norm

$$\| [u, \boldsymbol{\sigma}] \|^2 = k \|\nabla u\|^2 + \frac{k}{h} \|u\|_{L^2(\Gamma)}^2 + \frac{1}{k} \|\boldsymbol{\sigma}\|_{L^2(\Omega_{\Gamma,\text{in}})}^2, \tag{10}$$

where h is the element size. For simplicity, we will assume that \mathcal{P}_h is a uniform finite element partition. Note that the subscript $L^2(\Omega_{\Gamma,\text{in}})$ in the last term can be omitted for $\boldsymbol{\sigma}_h \in S_h$, because these functions vanish outside $\Omega_{\Gamma,\text{in}}$.

We define the bilinear form on $[V_h \times S_h]^2$

$$\begin{aligned} B([u_h, \boldsymbol{\sigma}_h], [v_h, \boldsymbol{\tau}_h]) &= k(\nabla u_h, \nabla v_h) - \langle \boldsymbol{\sigma}_h \cdot \mathbf{n}, v_h \rangle_{\Gamma} + \frac{1}{n}(\nabla v_h, \boldsymbol{\sigma}_h) - \frac{1}{n}k(\nabla v_h, \nabla u_h)_{\Omega_{\Gamma,\text{in}}} \\ &\quad - \frac{1}{nk}(\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) + \frac{1}{n}(\boldsymbol{\tau}_h, \nabla u_h) - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, u_h \rangle_{\Gamma}. \end{aligned} \tag{11}$$

An important technical aspect of the stability estimate we will obtain is that it relies on the following condition: we suppose that V_h and S_h are such that

$$\forall v_h \in V_h \quad \exists \boldsymbol{\tau}_h \in S_h \mid \delta_1 \|v_h\|_{L^2(\Gamma)}^2 \leq \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, v_h \rangle_{\Gamma} + \delta_0 h \|\nabla v_h\|^2, \tag{12}$$

$$\|\boldsymbol{\tau}_h\|_{L^2(\Gamma)} = \|v_h\|_{L^2(\Gamma)}, \quad \|\boldsymbol{\tau}_h\|^2 \leq \delta_2 h \|v_h\|_{L^2(\Gamma)}^2, \tag{13}$$

where $\delta_0, \delta_1, \delta_2$ are positive nondimensional constants which depend on the geometry of the mesh.

Conditions (12)–(13) are an assumption of our formulation, but let us check that they hold in two particular cases of interest, in both of which we assume that $\Gamma_K := \Gamma \cap K$ is a straight segment

Case 1

Equal order interpolation. This case is the simplest. τ_h can be defined within each element by

$$\tau_h|_{\Gamma_K} = \mathbf{n}v_h|_{\Gamma_K},$$

and then extended by parallel prolongation from Γ_K to the whole element domain K (recall that S_h is assumed to be made of discontinuous functions). It is obvious in this case that (12)–(13) hold, in fact with $\delta_0 = 0$ and $\delta_1 = 1$.

Case 2

V_h made of piecewise linears and S_h of piecewise constants. This is the situation considered in the numerical examples. Let us start defining $v_{lm,K}$ and $v_{qm,K}$ as the mean values of v_h and of v_h^2 , respectively in Γ_K

$$v_{lm,K} = \frac{\int_{\Gamma_K} v_h}{\int_{\Gamma_K} 1}, \quad v_{qm,K} = \sqrt{\frac{\int_{\Gamma_K} v_h^2}{\int_{\Gamma_K} 1}}.$$

A straightforward application of Schwarz' inequality yields $v_{qm} \geq |v_{lm}|$. Note also that for any $v_h \in V_h$, the following Poincaré-type inequality holds

$$\delta_1 \int_{\Gamma_K} v_h^2 \leq \int_{\Gamma_K} v_{lm,K}^2 + \delta_0 h \int_K |\nabla v_h|^2,$$

for $\delta_0 > 0, \delta_1 > 0$ that depend on the element shape. Thus, if we define $\tau_h^*|_K = \mathbf{n}v_{lm,K}$, we immediately get

$$\delta_1 \int_{\Gamma_K} v_h^2 \leq \int_{\Gamma_K} \tau_h^* \cdot \mathbf{n} v_h + \delta_0 h \int_K |\nabla v_h|^2. \tag{14}$$

Let us now introduce τ defined by

$$\tau_h|_K = \mathbf{n} v_{qm,K} \operatorname{sgn}(v_{lm,K}). \tag{15}$$

Because

$$\int_{\Gamma_K} \tau_h \cdot \mathbf{n} v_h = v_{qm,K} \int_K |v_h| \geq \int_{\Gamma_K} \tau_h^* \cdot \mathbf{n} v_h,$$

from (14), we get

$$\delta_1 \int_{\Gamma_K} v_h^2 \leq \int_{\Gamma_K} \tau_h \cdot \mathbf{n} v_h + \delta_0 h \int_K |\nabla v_h|^2.$$

Summing for all K , we have that τ_h defined in (15) satisfies (12). By its definition, it satisfies also the first part of (13), and the second is obvious, being τ_h piecewise constant.

We can proceed now to prove stability of the formulation (4)–(5) in the form of an inf-sup condition for the bilinear form (11) in the norm (10). Let us take $[v_h, \tau_h] = [u_h, -\sigma_h - \frac{\beta}{h} k \tilde{\tau}_h]$, where $\tilde{\tau}_h$ is the element in S_h that makes conditions (12)–(13) hold for u_h , and β is a dimensionless constant to be defined.

Using assumptions (12)–(13),

$$\begin{aligned}
 & B([u_h, \sigma_h], [u_h, -\sigma_h - \frac{\beta}{h}k\tilde{\tau}_h]) \\
 &= k(\nabla u_h, \nabla u_h) - \langle \sigma_h \cdot \mathbf{n}, u_h \rangle_\Gamma + \frac{1}{n}(\nabla u_h, \sigma_h) - \frac{1}{n}k(\nabla u_h, \nabla u_h)_{\Omega_{\Gamma, \text{in}}} \\
 &\quad + \frac{1}{nk}(\sigma_h, \sigma_h) - \frac{1}{n}(\sigma_h, \nabla u_h) + \langle \sigma_h \cdot \mathbf{n}, u_h \rangle_\Gamma \\
 &\quad + \frac{\beta}{nh}(\tilde{\tau}_h, \sigma_h) - \frac{\beta k}{nh}(\tilde{\tau}_h, \nabla u_h) + \frac{\beta k}{h}\langle \tilde{\tau}_h \cdot \mathbf{n}, u_h \rangle_\Gamma \\
 &\geq (1 - \frac{1}{n})k\|\nabla u_h\|^2 + \frac{1}{nk}\|\sigma_h\|^2 + \frac{\beta\delta_1 k}{h}\|u_h\|_{L^2(\Gamma)}^2 \\
 &\quad - \frac{\beta\delta_2^{1/2}}{nh^{1/2}}\|u_h\|_{L^2(\Gamma)}\|\sigma_h\| - \frac{\beta\delta_2^{1/2}k}{nh^{1/2}}\|u_h\|_{L^2(\Gamma)}\|\nabla u_h\| - \beta\delta_0 k\|\nabla u_h\|^2 \\
 &\geq (1 - \frac{1}{n})k\|\nabla u_h\|^2 + \frac{1}{nk}\|\sigma_h\|^2 + \frac{\beta\delta_1 k}{h}\|u_h\|_{L^2(\Gamma)}^2 \\
 &\quad - \frac{\delta_2^{1/2}}{2\gamma h^{1/2}}\frac{\beta k\delta_2^{1/2}}{nh^{1/2}}\|u_h\|_{L^2(\Gamma)}^2 - \frac{\gamma h^{1/2}}{2\delta_2^{1/2}}\frac{\beta\delta_2^{1/2}}{kn h^{1/2}}\|\sigma_h\|^2 \\
 &\quad - \frac{\delta_2^{1/2}}{2\gamma h^{1/2}}\frac{\beta k\delta_2^{1/2}}{nh^{1/2}}\|u_h\|_{L^2(\Gamma)}^2 - \frac{\gamma h^{1/2}}{2\delta_2^{1/2}}\frac{\beta k\delta_2^{1/2}}{nh^{1/2}}\|\nabla u_h\|^2 - \beta\delta_0 k\|\nabla u_h\|^2 \\
 &\geq \left(1 - \frac{1}{n} - \beta\left(\frac{\gamma}{2n} + \delta_0\right)\right)k\|\nabla u_h\|^2 + \frac{1}{n}\left(1 - \frac{\beta\gamma}{2}\right)\frac{1}{k}\|\sigma_h\|^2 + \beta\left(\delta_1 - \frac{\delta_2}{\gamma n}\right)\frac{k}{h}\|u_h\|_{L^2(\Gamma)}^2
 \end{aligned} \tag{16}$$

where γ is an arbitrary dimensionless constant. Choosing

$$n > 1, \quad \gamma > \frac{\delta_2}{\delta_1 n}, \quad \beta < \min\left(\frac{1 - \frac{1}{n}}{\left(\frac{\gamma}{2n} + \delta_0\right)}, \frac{2}{\gamma}\right),$$

and taking into account that

$$\begin{aligned}
 \|(u_h, -\sigma_h - \frac{\beta}{h}k\tilde{\tau}_h)\|^2 &= k\|\nabla u_h\|^2 + \frac{k}{h}\|u_h\|_{L^2(\Gamma)}^2 + \frac{1}{k}\|-\sigma_h - \frac{\beta}{h}k\tilde{\tau}_h\|^2 \\
 &\leq k\|\nabla u_h\|^2 + \frac{k}{h}\|u_h\|_{L^2(\Gamma)}^2 + \frac{2}{k}\|\sigma_h\|^2 + \frac{2k\beta^2}{h^2}\|\tilde{\tau}_h\|^2 \\
 &\leq k\|\nabla u_h\|^2 + \frac{k}{h}\|u_h\|_{L^2(\Gamma)}^2 + \frac{2}{k}\|\sigma_h\|^2 + \frac{2k\beta^2\delta_2}{h}\|u_h\|_{L^2(\Gamma)}^2 \\
 &\leq k\|\nabla u_h\|^2 + \frac{k}{h}(1 + 2\beta^2\delta_2)\|u_h\|_{L^2(\Gamma)}^2 + \frac{2}{k}\|\sigma_h\|^2 \\
 &\leq \max(1 + 2\beta^2\delta_2, 2)\|(u_h, \sigma_h)\|^2,
 \end{aligned}$$

we obtain the result we wished to prove.

Theorem 1

Suppose that conditions (12)–(13) are satisfied and that $n > 1$. Then, the bilinear form (11) satisfies that for all $[u_h, \sigma_h]$, there exist $[v_h, \tau_h]$ and $\alpha > 0$ such that

$$B([u_h, \sigma_h], [v_h, \tau_h]) \geq \alpha\|(u_h, \sigma_h)\|\|(v_h, \tau_h)\|.$$

We have proved that our symmetric bilinear form is stable for the Poisson problem for any $n > 1$. We consider $n = 2$ in the following.

The previous analysis allows us to very clearly understand the behavior of the method in terms of the mesh-dependent constants δ_0 , δ_1 , and δ_2 . The worst case scenario corresponds to $\delta_2 \gg \delta_1$

and $\delta_0 \gg 1$. Both cases correspond to ill-conditioned elements; that is, elements which are very elongated or in the case of embedded methods, suffer from bad intersections with the boundary of the physical domain. However, we must stress that no matter how bad the intersections are, we will always be able to find a sufficiently large γ and a sufficiently small β such that the method is stable as soon as $n > 1$. The stability estimate will be obviously weaker in these cases, leading to an ill-conditioned but still stable, system of equations. This shortcoming is not particular of our formulation but shared by most methods designed to prescribe boundary conditions on nonmatching meshes.

The analysis also shows the importance of taking S_h defined on the element interiors instead of taking it defined on Γ (which would lead to a method similar to the one proposed in [15]). In this case, 7 would also be defined on Γ , leading to a stability estimate of the type

$$B([u_h, \sigma_h], [u_h, -\sigma_h - \frac{\beta}{h}k\tilde{\tau}_h]) \geq \left(1 - \frac{C_{tr}}{n} - \beta(\dots)\right) k \|\nabla u_h\|^2 + \dots$$

where C_{tr} is the constant in the trace inequality $\|v_h\|_{L^2(\Gamma)}^2 \leq \frac{C_{tr}}{h} \|v_h\|^2$. In this case, the stability parameter n would depend on C_{tr} , which in turn depends on the geometry of the finite element mesh, whereas in the method we propose, it is sufficient to take $n > 1$.

2.4. Implementation and comparison to Nitsche’s method

A key feature of the presented method is that because the flux field is discontinuous across interelement edges, it can be eliminated from the final equations. We will show in this section that after eliminating the flux variables, some of the terms cancel out, and the final expression of the terms to be implemented is very similar to that of Nitsche’s method but with the important feature that we have been able to obtain, an estimate for the parameter n of the formulation independent of the finite element mesh. We will see later how a similar procedure can be applied to Nitsche’s method, although in this case, the algorithmic parameter is mesh (and geometry) dependent. A similar analysis for general stabilized Lagrange multiplier methods can be found in [32]. Let us also recall that we have also related the method proposed to the Lagrange multiplier technique. A comparison between this, our approach and Nitsche’s method will be summarized in Section 7.

Let U and Σ be the arrays of nodal unknowns of u_h and σ_h , respectively, and let us consider the matrices and vectors introduced in Table I for the algebraic version of problem (4)–(5). This problem written in matrix form is

$$\begin{bmatrix} (1 - \frac{1}{n})K_{uu} & K_{u\sigma} + G_{u\sigma} \\ K_{\sigma u} + G_{\sigma u} & K_{\sigma\sigma} \end{bmatrix} \begin{bmatrix} U \\ \Sigma \end{bmatrix} = \begin{bmatrix} f \\ g_{\sigma\bar{u}} \end{bmatrix}.$$

We can then compute the fluxes as

$$\Sigma = K_{\sigma\sigma}^{-1}(- (K_{\sigma u} + G_{\sigma u})U + g_{\sigma\bar{u}}).$$

In elements cut by Γ , $K_{\sigma\sigma}$ is block diagonal and therefore easy to invert due to the element-wise discontinuous flux approximation. This allows for the condensation of the flux unknowns at the

Table I. Matrices and vectors for the algebraic version of (4)–(5).

Discrete variational eqn.	$k(\nabla v_h, \nabla u_h)$	$-\frac{1}{nk}(\tau_h, \sigma_h)$	$\frac{1}{n}(\tau_h, \nabla u_h)$	$\frac{1}{n}(\nabla v_h, \sigma_h)$
Algebraic form	$K_{uu}U$	$K_{\sigma\sigma}\Sigma$	$K_{\sigma u}U$	$K_{u\sigma}\Sigma$
Discrete variational eqn.	$-\langle \mathbf{n} \cdot \sigma_h, v_h \rangle_\Gamma$	$-\langle \mathbf{n} \cdot \tau_h, u_h \rangle_\Gamma$	$-\langle \mathbf{n} \cdot \tau_h, \bar{u} \rangle_\Gamma$	$\langle f, v_h \rangle_\Omega$
Algebraic form	$G_{u\sigma}\Sigma$	$G_{\sigma u}U$	$g_{\sigma\bar{u}}$	f

element level, and we are left with only the original unknowns of the problem

$$\left[\left(1 - \frac{1}{n} \right) \mathbf{K}_{uu} - (\mathbf{G}_{u\sigma} + \mathbf{K}_{u\sigma}) \mathbf{K}_{\sigma\sigma}^{-1} (\mathbf{K}_{\sigma u} + \mathbf{G}_{\sigma u}) \right] \mathbf{U} = [\mathbf{f} - (\mathbf{G}_{u\sigma} + \mathbf{K}_{u\sigma}) \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}]. \quad (17)$$

This is the matrix version of the method we propose.

Let us consider now Nitsche’s method for the Poisson problem, which consists of finding $u_h \in V_h$ such that

$$\begin{aligned} & k(\nabla u_h, \nabla v_h) - k\langle \partial_n u_h, v_h \rangle_\Gamma - k\langle \partial_n v_h, u_h \rangle_\Gamma + k \frac{\alpha}{h} \langle u_h, v_h \rangle_\Gamma \\ & = \langle f, v_h \rangle_\Omega - k\langle \partial_n v_h, \bar{u} \rangle_\Gamma + k \frac{\alpha}{h} \langle \bar{u}, v_h \rangle_\Gamma, \end{aligned} \quad (18)$$

for all $v_h \in V_h$, where $\alpha > 0$ is now the algorithmic parameter of the formulation. First of all, let us observe that if we take $v_h = u_h$ in the right-hand side of this discrete variational equation, we obtain

$$\begin{aligned} & k\|\nabla u_h\|^2 - 2k\langle \partial_n u_h, u_h \rangle_\Gamma + k \frac{\alpha}{h} \|u_h\|_{L^2(\Gamma)}^2 \\ & \gtrsim k\|\nabla u_h\|^2 - 2k\|\partial_n u_h\|_{L^2(\Gamma)} \|u_h\|_{L^2(\Gamma)} + k \frac{\alpha}{h} \|u_h\|_{L^2(\Gamma)}^2 \\ & \gtrsim k\|\nabla u_h\|^2 - \frac{k\beta}{h} C_{tr} \|\nabla u_h\|^2 - \frac{k}{\beta} \|u_h\|_{L^2(\Gamma)}^2 + k \frac{\alpha}{h} \|u_h\|_{L^2(\Gamma)}^2 \end{aligned}$$

where $\beta > 0$ can be chosen, and C_{tr} is the constant in the trace inequality $\|v_h\|_{L^2(\Gamma)}^2 \leq \frac{C_{tr}}{h} \|v_h\|^2$. We have used the symbol \gtrsim to denote \geq up to positive constants. Choosing $\beta = \frac{h}{2C_{tr}}$, we observe that the bilinear form in (18) is coercive in the norm $k\|\nabla u_h\|^2 + k \frac{\alpha}{h} \|u_h\|_{L^2(\Gamma)}^2$ provided $\alpha > 2C_{tr}$. This bound depends on the geometry of the computational domain as well as on the finite element space V_h , whereas for the formulation, we propose the sufficient condition for stability $n > 1$ does not suffer from these dependencies.

Let us move our attention now to the matrix form of (18) in order to compare it with the matrix form of (4)–(5). Using the matrices introduced in Table II, this equation reads

$$[\mathbf{K}_{uu} + \mathbf{G}_{uu} + \mathbf{G}_{uu}^T + \mathbf{G}_{uu}^\alpha] \mathbf{U} = [\mathbf{f} + \mathbf{g}_{u\bar{u}} + \mathbf{g}_{u\bar{u}}^\alpha]. \quad (19)$$

In order to compare (17) with (19), we can significantly simplify the former with the help of the following identities:

$$\frac{1}{n} \mathbf{K}_{uu} \mathbf{U} = -[\mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{K}_{\sigma u}] \mathbf{U}, \quad (20)$$

$$\mathbf{G}_{uu}^T \mathbf{U} = -[\mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u}] \mathbf{U}, \quad (21)$$

$$\mathbf{g}_{u\bar{u}} = -[\mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}], \quad (22)$$

$$\mathbf{G}_{uu} \mathbf{U} = -[\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{K}_{\sigma u}] \mathbf{U}. \quad (23)$$

Let us prove that (20) holds if S_h is made of piecewise polynomials of one order less than the elements in V_h or higher. The proofs of the other three identities are similar.

Let $\tilde{\Sigma}$ be such that $\mathbf{K}_{\sigma\sigma} \tilde{\Sigma} = \mathbf{K}_{\sigma u} \mathbf{U}$. Then, $\tilde{\Sigma}$ are the nodal unknowns of $\tilde{\sigma}_h \in S_h$ such that $-\frac{1}{nk}(\boldsymbol{\tau}_h, \tilde{\sigma}_h) = \frac{1}{n}(\boldsymbol{\tau}_h, \nabla u_h)$ for all $\boldsymbol{\tau}_h \in S_h$, that is to say, $\tilde{\sigma}_h = -kP_{S_h}(\nabla u_h) = -k\nabla u_h$, where P_{S_h} is the L^2 projection onto the space of fluxes. Thus, $\mathbf{K}_{u\sigma} \tilde{\Sigma} = \mathbf{K}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{K}_{\sigma u} \mathbf{U}$ amounts to say that $\frac{1}{n}(\nabla v_h, \tilde{\sigma}_h) = \frac{1}{n}(\nabla v_h, -k\nabla u_h)$, which implies (20).

Table II. Matrices and vectors for the algebraic version of (18.)

Discrete variational equation	$-k\langle \mathbf{n} \cdot \nabla u_h, v_h \rangle_\Gamma$	$\frac{\alpha k}{h} \langle v_h, u_h \rangle_\Gamma$	$k\langle \mathbf{n} \cdot \nabla v_h, \bar{u} \rangle_\Gamma$	$\frac{\alpha k}{h} \langle v_h, \bar{u} \rangle_\Gamma$
Algebraic form	$\mathbf{G}_{uu} \mathbf{U}$	$\mathbf{G}_{uu}^\alpha \mathbf{U}$	$\mathbf{g}_{u\bar{u}}$	$\mathbf{g}_{u\bar{u}}^\alpha$

Taking identities (20)–(23) into account, we can write the matrix form (17) as

$$[\mathbf{K}_{uu} + \mathbf{G}_{uu} + \mathbf{G}_{uu}^T - \mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u}] \mathbf{U} = [\mathbf{f} + \mathbf{g}_{u\bar{u}} - \mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}]. \tag{24}$$

Comparing (24) and (19), we can conclude that the only difference between the presented method and Nitsche’s method is that we have replaced

$$\mathbf{G}_{uu}^\alpha \mathbf{U} \text{ and } \mathbf{g}_{u\bar{u}}^\alpha, \tag{25}$$

by

$$-\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{G}_{\sigma u} \mathbf{U} \text{ and } -\mathbf{G}_{u\sigma} \mathbf{K}_{\sigma\sigma}^{-1} \mathbf{g}_{\sigma\bar{u}}. \tag{26}$$

We end up with a symmetric method which is identical to Nitsche’s method (for a rich enough discontinuous flux field), except for the so called penalty term. Note that the penalty terms in (25) involve boundary integrals, whereas in (26), they involve volume integrals. Although the system of equations is not positive definite before eliminating the stress nodal unknowns, the final matrix after condensing the element-wise discontinuous stress field is symmetric and positive definite. This can be seen by doing an analysis similar to the one presented in [33].

As we have seen, the advantage of our formulation is that the algorithmic parameter in our approach is independent of the geometry and discretization. We will also see in the numerical examples that results are less sensitive to the value of n in (4)–(5) than to the value of α in (18).

3. INTRODUCING CONVECTION

3.1. Problem statement

In this section, we deal with the convection–diffusion equation. The problem to be solved is no longer symmetric, and additional terms are needed in order to ensure the stability of the final weak form. The problem we consider in this section consists of finding $u : \Omega \rightarrow \mathbb{R}$ and $\sigma : \Omega_{\Gamma,\text{in}} \rightarrow \mathbb{R}^d$ such that

$$-\nabla \cdot \sigma + \mathbf{a} \cdot \nabla u = f \quad \text{in } \Omega, \tag{27}$$

$$\frac{1}{k} \sigma = \nabla u \quad \text{in } \Omega_{\Gamma,\text{in}}, \tag{28}$$

$$u = \bar{u} \quad \text{on } \Gamma = \partial\Omega, \tag{29}$$

where $k > 0$, \mathbf{a} is the advection velocity, f is a given forcing function and \bar{u} is the given Dirichlet boundary condition. We have already used the two-field formulation presented for Poisson’s problem. In this case, the advective flux could also be included in the definition of σ , which could be convenient if the equation is written in conservative form, although we will not pursue this option here.

3.2. Weak form

The variational form of the problem consists of finding $u_h \in V_h$ and $\sigma_h \in S_h$ such that

$$\begin{aligned} k(\nabla u_h, \nabla v_h) - \langle \sigma_h \cdot \mathbf{n}, v_h \rangle_\Gamma + (\mathbf{a} \cdot \nabla u_h, v_h) + \frac{1}{n}(\nabla v_h, \sigma_h) - \frac{1}{n}k(\nabla v_h, \nabla u_h)_{\Omega_{\Gamma,\text{in}}} + \frac{1}{2}\langle av_h, u_h \rangle_\Gamma \\ = \langle f, v_h \rangle_\Omega + \frac{1}{2}\langle av_h, \bar{u} \rangle_\Gamma, \quad \forall v_h \in V_h, \end{aligned} \tag{30}$$

$$-\frac{1}{nk}(\boldsymbol{\tau}_h, \sigma_h) + \frac{1}{n}(\boldsymbol{\tau}_h, \nabla u_h) - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, u_h \rangle_\Gamma = -\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{u} \rangle_\Gamma, \quad \forall \boldsymbol{\tau}_h \in S_h. \tag{31}$$

Note that in the previous weak form, we have replaced (6) with

$$k(\nabla u, \nabla v_h) - \langle \sigma \cdot \mathbf{n}, v_h \rangle_\Gamma + (\mathbf{a} \cdot \nabla u, v_h) = \langle f, v_h \rangle_\Omega,$$

and we have added the discrete version of

$$\frac{1}{2} \langle av_h, u \rangle_\Gamma = \frac{1}{2} \langle av_h, \bar{u} \rangle_\Gamma,$$

which is weakly enforcing (3) tested against $\frac{a}{2}v_h$. We will see how to define the parameter a in the following subsections. The need for introducing this term will be clear from the subsequent stability analysis.

It is observed that, apart from the way to impose the boundary conditions, (30)–(31) is based on the standard Galerkin method to solve the convection–diffusion reaction equation. This method is stable only for high values of the diffusion coefficient k . Even though in the examples, we will consider convection-dominated flows solved using a stabilized formulation; for the sake of conciseness, the exposition will be developed in the diffusion-dominated case. Likewise, we will consider a constant for simplicity.

3.3. Stability

In this subsection, we prove that the formulation given by (30)–(31) is stable and as a consequence has a unique solution.

We define the bilinear form on $[V_h, S_h] \times [V_h, S_h]$

$$\begin{aligned} B^c([u_h, \sigma_h], [v_h, \tau_h]) &= k(\nabla u_h, \nabla v_h) - \langle \sigma_h \cdot \mathbf{n}, v_h \rangle_\Gamma + \frac{1}{n}(\nabla v_h, \sigma_h) - \frac{1}{n}k(\nabla v_h, \nabla u_h)_{\Omega_{\Gamma, \text{in}}} \\ &+ (\mathbf{a} \cdot \nabla u_h, v_h) + \frac{1}{2} \langle av_h, u_h \rangle_\Gamma - \frac{1}{nk}(\tau_h, \sigma_h) + \frac{1}{n}(\tau_h, \nabla u_h) - \langle \tau_h \cdot \mathbf{n}, u_h \rangle_\Gamma. \end{aligned} \quad (32)$$

Taking $[v_h, \tau_h] = [u_h, -\sigma_h - \frac{\beta}{h}k\tilde{\tau}_h]$, with $\tilde{\tau}_h$ as in the proof of Theorem 1 in the previous section, we have

$$\begin{aligned} B^c([u_h, \sigma_h], [u_h, -\sigma_h - \frac{\beta}{h}k\tilde{\tau}_h]) &= k(\nabla u_h, \nabla u_h) - \langle \sigma_h \cdot \mathbf{n}, u_h \rangle_\Gamma + \frac{1}{n}(\nabla u_h, \sigma_h) - \frac{1}{n}k(\nabla u_h, \nabla u_h)_{\Omega_{\Gamma, \text{in}}} \\ &+ (\mathbf{a} \cdot \nabla u_h, u_h) + \frac{1}{2} \langle au_h, u_h \rangle_\Gamma + \frac{1}{nk}(\sigma_h, \sigma_h) - \frac{1}{n}(\sigma_h, \nabla u_h) + \langle \sigma_h \cdot \mathbf{n}, u_h \rangle_\Gamma \\ &+ \frac{\beta}{nh}(\tilde{\tau}_h, \sigma_h) - \frac{\beta k}{nh}(\tilde{\tau}_h, \nabla u_h) + \frac{\beta k}{h} \langle \tilde{\tau}_h \cdot \mathbf{n}, u_h \rangle_\Gamma \\ &\geq \left(1 - \frac{1}{n} - \beta \left(\frac{\gamma}{2n} + \delta_0\right)\right) k \|\nabla u_h\|^2 + \frac{1}{n} \left(1 - \frac{\beta \gamma}{2}\right) \frac{1}{k} \|\sigma_h\|^2 + \beta \left(\delta_1 - \frac{\delta_2}{\gamma n}\right) \frac{k}{h} \|u_h\|_{L^2(\Gamma)}^2 \\ &+ \frac{1}{2} \int_\Gamma (\mathbf{n} \cdot \mathbf{a} + a) u_h^2, \end{aligned}$$

where the same steps as in (16) have been carried out. From this, we obtain

Theorem 2

Suppose that conditions (12)–(13) are satisfied, that $n > 1$ and that the parameter a is chosen such that $a + \mathbf{a} \cdot \mathbf{n} \geq 0$ on Γ . Then, the bilinear form (32) satisfies that for all $[u_h, \sigma_h]$, there exists $[v_h, \tau_h]$ and $\alpha > 0$ such that

$$B^c([u_h, \sigma_h], [v_h, \tau_h]) \geq \alpha \| [u_h, \sigma_h] \| \| [v_h, \tau_h] \|.$$

Taking the condition $a + \mathbf{a} \cdot \mathbf{n} \geq 0$ into account, the obvious definition for a is

$$\begin{aligned} a &= -\mathbf{a} \cdot \mathbf{n}, \quad \text{if } \mathbf{a} \cdot \mathbf{n} < 0 \\ a &= 0, \quad \text{otherwise.} \end{aligned}$$

This definition of the weighting term a is very similar to the one used in the weak imposition of boundary conditions in [34], the main difference being that in our case is accompanied with a $\frac{1}{2}$ factor.

4. EXTENSION TO THE STOKES PROBLEM

In this section, we extend the previous ideas to the Stokes problem, for which we obtain a symmetric method again. Once the method is defined for the Stokes problem, we can deal with the Navier–Stokes equations just by putting together the formulation presented in Section 3 and the one in the current section.

4.1. Problem statement

Let us consider a three-field formulation for the Stokes problem, which consists of finding $\mathbf{u} : \Omega \rightarrow \mathbb{R}^d$, $p : \Omega \rightarrow \mathbb{R}$ and $\boldsymbol{\sigma} : \Omega_{\Gamma,\text{in}} \rightarrow \mathbb{R}^{d \times d}$ such that

$$-\nu \Delta \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega, \tag{33}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \tag{34}$$

$$\frac{1}{\nu} \boldsymbol{\sigma} = \nabla \mathbf{u} \quad \text{in } \Omega_{\Gamma,\text{in}}, \tag{35}$$

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma = \partial\Omega, \tag{36}$$

where $\nu > 0$, \mathbf{f} is a given forcing function and $\bar{\mathbf{u}}$ is the given Dirichlet boundary condition. Note that $\boldsymbol{\sigma}$ only accounts for the deviatoric part of the pseudo-stresses (we could also formulate the method in terms of the strain rate tensor $\nabla^s \mathbf{u}$).

4.2. Discrete weak form

Let us consider the finite element spaces $V_h \subset H^1(\Omega_h)^d$, $Q_h \subset L^2(\Omega_h)$ (we will consider V_h and Q_h made of continuous functions) and $S_h \subset L^2(\Omega_\Gamma)^{d \times d}$ (we will consider S_h made of element-wise discontinuous functions). The standard finite element approximation of the Stokes problem is not stable for an arbitrary \mathbf{u} , p interpolation. This is the reason why we add stabilization terms to the original weak form of the problem, which allow us to use equal interpolation spaces for velocity and pressure. Our stabilized approach to the problem is based on the formulation proposed in [35, 36]. In this case, it will not be necessary to stabilize the additional pseudo-stress $\boldsymbol{\sigma}_h$ because, as in the problems discussed previously, its stability relies on the way to choose S_h .

The formulation we consider here consists of finding $\mathbf{u}_h \in V_h$, $p_h \in Q_h$ and $\boldsymbol{\sigma}_h \in S_h$ such that

$$\begin{aligned} & \nu(\nabla \mathbf{u}_h, \nabla \mathbf{v}_h) - (\nabla \cdot \mathbf{v}_h, p_h) - \langle \boldsymbol{\sigma}_h \cdot \mathbf{n}, \mathbf{v}_h \rangle_\Gamma + \langle \mathbf{n} \cdot \mathbf{v}_h p_h \rangle_\Gamma \\ & + \sum_K \tau_K (\nu \Delta \mathbf{v}_h, -\nu \Delta \mathbf{u}_h + \nabla p_h)_K + \frac{1}{n} (\nabla \mathbf{v}_h, \boldsymbol{\sigma}_h) - \frac{1}{n} \nu (\nabla \mathbf{u}_h, \nabla \mathbf{v}_h)_{\Omega_{\Gamma,\text{in}}} \\ & = \langle \mathbf{f}, \mathbf{v}_h \rangle_\Omega + \sum_K \tau_K (\nu \Delta \mathbf{v}_h, \mathbf{f})_K, \quad \forall \mathbf{v}_h \in V_h, \end{aligned} \tag{37}$$

$$\begin{aligned} & - (q_h, \nabla \cdot \mathbf{u}_h) - \sum_K \tau_K (\nabla q_h, -\nu \Delta \mathbf{u}_h + \nabla p_h)_K + \langle q_h \mathbf{n} \cdot \mathbf{u}_h \rangle_\Gamma \\ & = - \sum_K \tau_K (\nabla q_h, \mathbf{f})_K + \langle q_h, \mathbf{n} \cdot \bar{\mathbf{u}} \rangle_\Gamma, \quad \forall q_h \in Q_h, \end{aligned} \tag{38}$$

$$- \frac{1}{n\nu} (\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) + \frac{1}{n} (\boldsymbol{\tau}_h, \nabla \mathbf{u}_h) - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \mathbf{u}_h \rangle_\Gamma = - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{\mathbf{u}} \rangle_\Gamma, \quad \forall \boldsymbol{\tau}_h \in S_h. \tag{39}$$

In these equation, $(\cdot, \cdot)_K \equiv (\cdot, \cdot)_{L^2(K)}$ and the so called stabilization parameters τ_K are computed as

$$\tau_K = \left(c_1 \frac{\nu}{h^2} \right)^{-1}, \tag{40}$$

in each element, where h is the element size. For the numerical experiments, we have taken $c_1 = 4$. Note that for the exact solution $\mathbf{u}, p, \boldsymbol{\sigma}$ there holds

$$\begin{aligned} \nu(\nabla \mathbf{u}, \nabla \mathbf{v}_h) - (\nabla \cdot \mathbf{v}_h, p) - \langle \boldsymbol{\sigma} \cdot \mathbf{n}, \mathbf{v}_h \rangle_\Gamma + \langle \mathbf{n} \cdot \mathbf{v}_h, p \rangle_\Gamma &= \langle \mathbf{f}, \mathbf{v}_h \rangle_\Omega, \\ \sum_K \tau_K (\nu \Delta \mathbf{v}_h, -\nu \Delta \mathbf{u} + \nabla p)_K &= \sum_K \tau_K (\nu \Delta \mathbf{v}_h, \mathbf{f})_K, \\ \frac{1}{n} (\nabla \mathbf{v}_h, \boldsymbol{\sigma})_{\Omega_{\Gamma, \text{in}}} - \frac{1}{n} \nu (\nabla \mathbf{v}_h, \nabla \mathbf{u}_h)_{\Omega_{\Gamma, \text{in}}} &= 0, \\ -(q_h, \nabla \cdot \mathbf{u}) &= 0, \\ \sum_K \tau_K (\nabla q_h, -\nu \Delta \mathbf{u} + \nabla p)_K &= \sum_K \tau_K (\nabla q_h, \mathbf{f})_K, \\ \langle q_h, \mathbf{n} \cdot \mathbf{u} \rangle_\Gamma &= \langle q_h, \mathbf{n} \cdot \bar{\mathbf{u}} \rangle_\Gamma \\ -\frac{1}{n\nu} (\boldsymbol{\tau}_h, \boldsymbol{\sigma})_{\Omega_{\Gamma, \text{in}}} + \frac{1}{n} (\boldsymbol{\tau}_h, \nabla \mathbf{u})_{\Omega_{\Gamma, \text{in}}} &= 0, \\ -\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \mathbf{u} \rangle_\Gamma &= -\langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{\mathbf{u}} \rangle_\Gamma. \end{aligned}$$

The method we propose is obtained by summing all these equations and replacing the continuous solution by the finite element approximation. Thus, it is fully consistent. Let us remark that it is important to add the terms $\langle q_h, \mathbf{n} \cdot \mathbf{u}_h \rangle_\Gamma$ and $\langle q_h, \mathbf{n} \cdot \bar{\mathbf{u}} \rangle_\Gamma$ to obtain a symmetric problem, and also, that the stabilization terms are independent of the way boundary conditions are imposed.

4.3. Stability

In this subsection, we prove that the formulation given by (37)–(39) is stable and as a consequence has a unique solution. We define the norm

$$\|[\mathbf{u}, p, \boldsymbol{\sigma}]\|^2 = \nu \|\nabla \mathbf{u}\|^2 + \frac{\nu}{h} \|\mathbf{u}\|_{L^2(\Gamma)}^2 + \frac{h^2}{\nu} \|\nabla p\|^2 + \frac{1}{\nu} \|\boldsymbol{\sigma}\|_{L^2(\Omega_{\Gamma, \text{in}})}^2. \tag{41}$$

Subscript $L^2(\Omega_{\Gamma, \text{in}})$ in the last term can be omitted for discrete functions $\boldsymbol{\sigma}_h$, because elements in S_h vanish outside $\Omega_{\Gamma, \text{in}}$. We also introduce the bilinear form on $[V_h \times Q_h \times S_h]^2$

$$\begin{aligned} B^s([\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h], [\mathbf{v}_h, q_h, \boldsymbol{\tau}_h]) &= \nu(\nabla \mathbf{u}_h, \nabla \mathbf{v}_h) - (\nabla \cdot \mathbf{v}_h, p_h) - \langle \boldsymbol{\sigma}_h \cdot \mathbf{n}, \mathbf{v}_h \rangle_\Gamma + \langle \mathbf{n} \cdot \mathbf{v}_h, p_h \rangle_\Gamma + \frac{1}{n} (\nabla \mathbf{v}_h, \boldsymbol{\sigma}_h) \\ &\quad - \frac{1}{n} \nu (\nabla \mathbf{v}_h, \nabla \mathbf{u}_h)_{\Omega_{\Gamma, \text{in}}} + \sum_K \tau_K (\nu \Delta \mathbf{v}_h - \nabla q_h, -\nu \Delta \mathbf{u}_h + \nabla p_h)_K - (q_h, \nabla \cdot \mathbf{u}_h) \\ &\quad + \langle q_h, \mathbf{n} \cdot \mathbf{u}_h \rangle_\Gamma - \frac{1}{n\nu} (\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) + \frac{1}{n} (\boldsymbol{\tau}_h, \nabla \mathbf{u}_h) - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \mathbf{u}_h \rangle_\Gamma. \end{aligned} \tag{42}$$

Similarly to the Poisson problem, we assume that V_h and S_h are such that the vector counterpart of (12)–(13) holds. With this, we can proceed to prove stability in the form of an inf-sup

condition for (42). Let $\tilde{\tau}_h$ be such that the vector counterpart of (12)–(13) holds for \mathbf{u}_h . Taking $[\mathbf{v}_h, q_h, \boldsymbol{\tau}_h] = [\mathbf{u}_h, -p_h, -\boldsymbol{\sigma}_h - \frac{\beta}{h}v\tilde{\tau}_h]$, with β as a constant to be determined, we have

$$\begin{aligned} & B^s \left([\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h], \left[\mathbf{u}_h, -p_h, -\boldsymbol{\sigma}_h - \frac{\beta}{h}v\tilde{\tau}_h \right] \right) \\ &= v(\nabla \mathbf{u}_h, \nabla \mathbf{u}_h) - \frac{1}{n}v(\nabla \mathbf{v}_h, \nabla \mathbf{u}_h)_{\Omega_{\Gamma, \text{in}}} - \sum_K \tau_K (v\Delta \mathbf{u}_h, v\Delta \mathbf{u}_h)_K \\ &+ \sum_K \tau_K (\nabla p_h, \nabla p_h)_K + \frac{1}{nv}(\boldsymbol{\sigma}_h, \boldsymbol{\sigma}_h) \\ &+ \frac{\beta}{nh}(\tilde{\tau}_h, \boldsymbol{\sigma}_h) - \frac{\beta v}{nh}(\tilde{\tau}_h, \nabla \mathbf{u}_h) + \frac{\beta v}{h}\langle \tilde{\tau}_h \cdot \mathbf{n}, \mathbf{u}_h \rangle_{\Gamma} \\ &\geq \left(1 - \frac{1}{n} - \beta \left(\frac{\gamma}{2n} + \delta_0 \right) \right) v \|\nabla \mathbf{u}_h\|^2 - \sum_K \tau_K \|v\Delta \mathbf{u}_h\|_K^2 + \beta \left(\delta_1 - \frac{\delta_2}{\gamma n} \right) \frac{v}{h} \|\mathbf{u}_h\|_{L^2(\Gamma)}^2 \\ &+ \sum_K \tau_K \|\nabla p_h\|_K^2 + \frac{1}{n} \left(1 - \frac{\beta\gamma}{2} \right) \frac{1}{v} \|\boldsymbol{\sigma}_h\|^2 \\ &\geq \left(1 - \frac{1}{n} - \beta \left(\frac{\gamma}{2n} + \delta_0 \right) - C_1 \right) v \|\nabla \mathbf{u}_h\|^2 + \beta \left(\delta_1 - \frac{\delta_2}{\gamma n} \right) \frac{v}{h} \|\mathbf{u}_h\|_{L^2(\Gamma)}^2 \\ &+ C_2 \frac{h^2}{v} \|\nabla p_h\|^2 + \frac{1}{n} \left(1 - \frac{\beta\gamma}{2} \right) \frac{1}{v} \|\boldsymbol{\sigma}_h\|^2, \end{aligned}$$

where C_1 is such that, making use of a classical inverse estimate

$$\sum_K \tau_K \|v\Delta \mathbf{u}_h\|_K^2 \leq C_1 v \|\nabla \mathbf{u}_h\|^2,$$

and C_2 is defined as $\frac{1}{c_1}$ in (40). Imposing

$$n > 1, \quad \gamma > \frac{\delta_2}{\delta_1 n}, \quad \beta < \min \left(\frac{1 - \frac{1}{n}}{\left(\frac{\gamma}{2n} + \delta_0 \right)}, \frac{2}{\gamma} \right),$$

with C_1 sufficiently small (i.e., c_1 sufficiently large), we can now define the constant

$$\alpha_0 = \min \left(1 - \frac{1}{n} - \beta \left(\frac{\gamma}{2n} + \delta_0 \right) - C_1, C_2, \frac{1}{n} \left(1 - \frac{\beta\gamma}{2} \right), \beta \left(\delta_1 - \frac{\delta_2}{\gamma n} \right) \right),$$

for which there holds

$$B^s \left([\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h], \left[\mathbf{u}_h, -p_h, -\boldsymbol{\sigma}_h - \frac{\beta}{h}v\tilde{\tau}_h \right] \right) \geq \alpha_0 \|[\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h]\|^2.$$

We now take into account that

$$\begin{aligned} & \left\| \left[\mathbf{u}_h, -p_h, -\boldsymbol{\sigma}_h - \frac{\beta}{h}v\tilde{\tau}_h \right] \right\|^2 = v \|\nabla \mathbf{u}_h\|^2 + \frac{v}{h} \|\mathbf{u}_h\|_{L^2(\Gamma)}^2 + \frac{h^2}{v} \|\nabla p_h\|^2 + \frac{1}{v} \left\| -\boldsymbol{\sigma}_h - \frac{\beta}{h}v\tilde{\tau}_h \right\|^2 \\ &\leq v \|\nabla \mathbf{u}_h\|^2 + \frac{v}{h} \|\mathbf{u}_h\|_{L^2(\Gamma)}^2 + \frac{h^2}{v} \|\nabla p_h\|^2 + \frac{2}{v} \|\boldsymbol{\sigma}_h\|^2 + \frac{2v\beta^2\delta_2}{h} \|\mathbf{u}_h\|_{L^2(\Gamma)}^2 \\ &\leq v \|\nabla \mathbf{u}_h\|^2 + \frac{v}{h} (1 + 2\beta^2\delta_2) \|\mathbf{u}_h\|_{L^2(\Gamma)}^2 + \frac{h^2}{v} \|\nabla p_h\|^2 + \frac{2}{v} \|\boldsymbol{\sigma}_h\|^2 \\ &\leq \max(1 + 2\beta^2\delta_2, 2) \|[\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h]\|^2, \end{aligned}$$

which allows us to obtain the result we were looking for.

Theorem 3

Suppose that the vector counterpart of conditions (12)–(13) is satisfied, that $n > 1$ and that c_1 in (40) is sufficiently large. Then, the bilinear form (42) satisfies that for all $[\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h]$, there exist $[\mathbf{v}_h, q_h, \boldsymbol{\tau}_h]$ and $\alpha > 0$ such that

$$B^s([\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h], [\mathbf{v}_h, q_h, \boldsymbol{\tau}_h]) \geq \alpha \|[\mathbf{u}_h, p_h, \boldsymbol{\sigma}_h]\| \|[\mathbf{v}_h, q_h, \boldsymbol{\tau}_h]\|.$$

5. TRANSIENT NAVIER–STOKES EQUATIONS

The proposed method for imposing boundary conditions in the transient Navier–Stokes equations consists simply in putting together the terms appearing in the convection–diffusion equation with the ones in the Stokes problem. As in the Stokes problem, a stabilized formulation is required so that equal interpolations for the velocity and pressure can be used. Moreover, additional stabilization terms are added so that we can deal with convection-dominated problems. For details on the stabilized formulation employed, see [36].

5.1. Problem statement

Let us consider a three-field formulation for the transient Navier–Stokes equations, which consists of finding $\mathbf{u} : \Omega \rightarrow \mathbb{R}^d$, $p : \Omega \rightarrow \mathbb{R}$ and $\boldsymbol{\sigma} : \Omega_{\Gamma, \text{in}} \rightarrow \mathbb{R}^{d \times d}$ such that

$$\partial_t \mathbf{u} - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega, \quad (43)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad (44)$$

$$\frac{1}{\nu} \boldsymbol{\sigma} = \nabla \mathbf{u} \quad \text{in } \Omega_{\Gamma, \text{in}}, \quad (45)$$

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma = \partial\Omega, \quad (46)$$

for $t > 0$, where $\partial_t \mathbf{u}$ is the local time derivative of the velocity field. Appropriate initial conditions have to be appended to this problem.

5.2. Discrete weak form

Let us consider the finite element spaces V_h , Q_h , and S_h as for the Stokes problem. Suppose that $\delta_t \mathbf{u}_h$ is an approximation of $\partial_t \mathbf{u}_h$ obtained from a partition of the time domain, using for example a backward difference approximation, it is still denote by \mathbf{u}_h , p_h , and $\boldsymbol{\sigma}_h$ the velocity, pressure, and pseudo-stress at the time step where the problem is solved.

The discrete variational form of (43)–(46) consists of finding $\mathbf{u}_h \in V_h$, $p_h \in Q_h$, and $\boldsymbol{\sigma}_h \in S_h$ such that

$$\begin{aligned} & (\mathbf{v}_h, \delta_t \mathbf{u}_h) + \nu (\nabla \mathbf{u}_h, \nabla \mathbf{v}_h) + (\mathbf{v}_h, \mathbf{u}_h \cdot \nabla \mathbf{u}_h) - (\nabla \cdot \mathbf{v}_h, p_h) - \langle \boldsymbol{\sigma}_h \cdot \mathbf{n}, \mathbf{v}_h \rangle_{\Gamma} \\ & \quad + \langle \mathbf{n} \cdot \mathbf{v}_h, p_h \rangle_{\Gamma} + \sum_K \tau_K (\nu \Delta \mathbf{v}_h + \mathbf{u}_h \cdot \nabla \mathbf{v}_h, \delta_t \mathbf{u}_h - \nu \Delta \mathbf{u}_h + \mathbf{u}_h \cdot \nabla \mathbf{u}_h + \nabla p_h)_K \\ & \quad + \frac{1}{n} (\nabla \mathbf{v}_h, \boldsymbol{\sigma}_h) - \frac{1}{n} \nu (\nabla \mathbf{v}_h, \nabla \mathbf{u}_h)_{\Omega_{\Gamma, \text{in}}} + \frac{1}{2} \langle a \mathbf{v}_h, \mathbf{u}_h \rangle_{\Gamma} \\ & = \langle \mathbf{f}, \mathbf{v}_h \rangle_{\Omega} + \sum_K \tau_K (\nu \Delta \mathbf{v}_h + \mathbf{u}_h \cdot \nabla \mathbf{v}_h, \mathbf{f})_K + \frac{1}{2} \langle a \mathbf{v}_h, \bar{\mathbf{u}} \rangle_{\Gamma} \quad \forall \mathbf{v}_h \in V_h, \\ & - (q_h, \nabla \cdot \mathbf{u}_h) - \sum_K \tau_K (\nabla q_h, \delta_t \mathbf{u}_h - \nu \Delta \mathbf{u}_h + \mathbf{u}_h \cdot \nabla \mathbf{u}_h + \nabla p_h)_K + \langle q_h, \mathbf{n} \cdot \mathbf{u}_h \rangle_{\Gamma} \\ & = - \sum_K \tau_K (\nabla q_h, \mathbf{f})_K + \langle q_h, \mathbf{n} \cdot \bar{\mathbf{u}} \rangle_{\Gamma}, \quad \forall q_h \in Q_h, \\ & - \frac{1}{n \nu} (\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) + \frac{1}{n} (\boldsymbol{\tau}_h, \nabla \mathbf{u}_h) - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \mathbf{u}_h \rangle_{\Gamma} = - \langle \boldsymbol{\tau}_h \cdot \mathbf{n}, \bar{\mathbf{u}} \rangle_{\Gamma}, \quad \forall \boldsymbol{\tau}_h \in S_h, \end{aligned}$$

and now, the stabilization parameter is computed as [36]

$$\tau_K = \left(c_1 \frac{\nu}{h^2} + c_2 \frac{|\mathbf{u}_h|_K}{h} \right)^{-1},$$

where $|\mathbf{u}_h|_K$ is the mean velocity modulus in element K . We take the stability constants as $c_1 = 4$ and $c_2 = 2$. The parameter a in the momentum equation is taken as

$$a = \max(0, -\mathbf{n} \cdot \mathbf{u}_h).$$

6. NUMERICAL EXAMPLES

6.1. Diffusion and convection–diffusion equation

In this subsection, we illustrate the behavior of the proposed method for the scalar diffusion and convection–diffusion equations. The problems are posed in a domain Ω enclosed in a circle of radius $R < 1$. We choose the *hold-all domain* $B = (-1, 1) \times (-1, 1)$, where a system of Cartesian coordinates (x, y) with its origin at the center of the circle has been adopted. A structured mesh of right-angled linear triangular elements is constructed in B , h being the length of the edges corresponding to the cathetus.

6.1.1. The Poisson equation. Let us start solving the Poisson equation with $k = 1$, $\mathbf{a} = \mathbf{0}$, $f = 1$ to check the performance and convergence of the proposed method. The analytical solution for this case is known to be

$$u(x, y) = \frac{1}{4}(R^2 - x^2 - y^2).$$

Figure 3 shows the errors $\|u - u_h\|_{L^2(\Omega)}$ versus the element size h , both using the method proposed in this paper and Nitsche’s method for different values of the algorithmic parameters of both. The coarsest mesh is built of 1250 elements, whereas the finer one is built out of 320,000 elements. From the convergence curves, we can conclude that: (1) both methods display quadratic convergence when linear elements are used; (2) accuracy is in general better with the present approach; and (3) Nitsche’s results are more sensitive to the algorithmic constant α than the method proposed here to the value of n . The best approximation is obtained for $n = 2$, which strictly ensures stability according to the stability analysis ($n > 1$).

6.1.2. Convection–diffusion. Figure 4 shows the behavior of the method in the convection–diffusion problem, where $k = 10^{-2}$, $\mathbf{a} = (1, 0)$, $f = 1$. A stabilized formulation similar to the

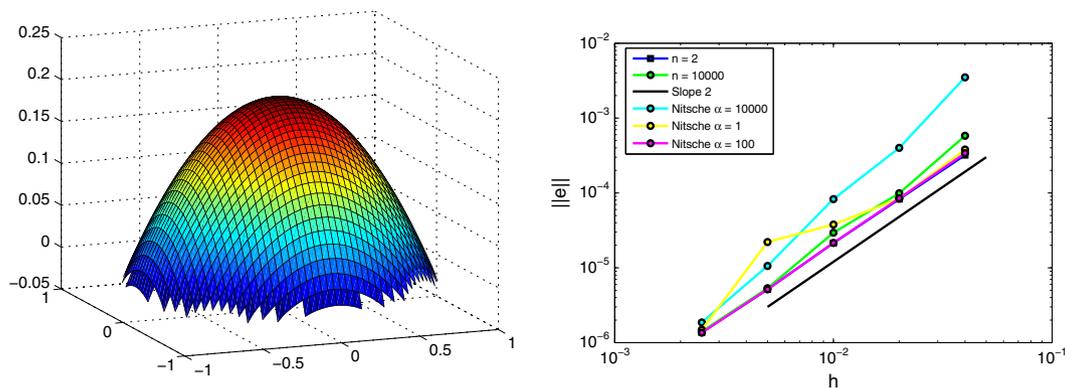


Figure 3. Results and error convergence for the solution of the diffusion equation.

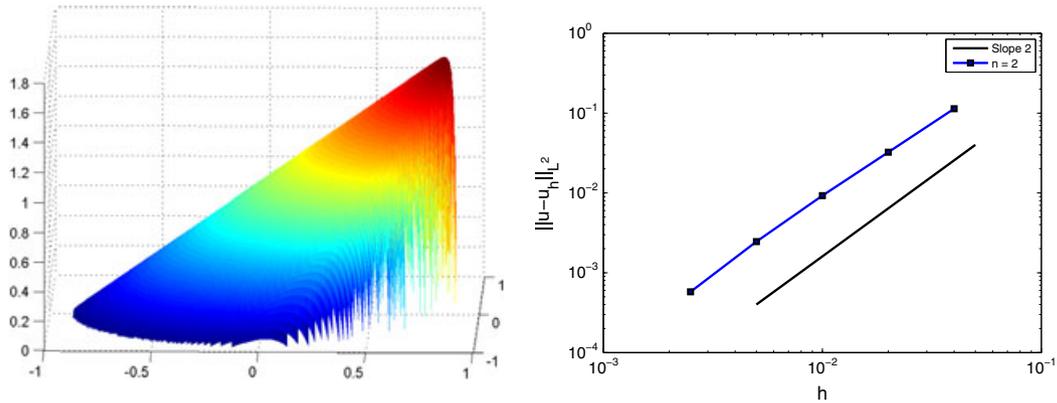


Figure 4. Results and error convergence for the solution of the convection–diffusion equation.

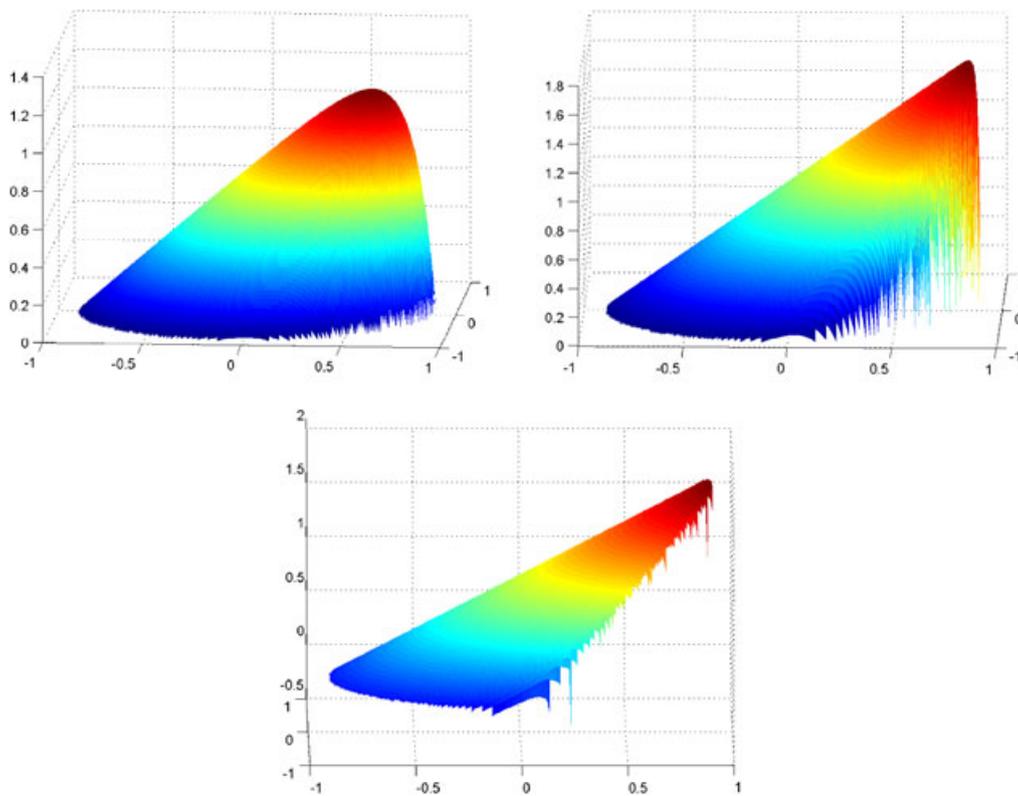


Figure 5. Solution comparison for the convection–diffusion equation with diffusion coefficients 10^{-1} , 10^{-2} , and 10^{-5} .

one described in Section 5 has been used. In order to obtain the error, we have computed the solution for a very fine mesh ($h = 2/800$, 1,200,000 elements) which we have used as the reference solution. Again, the method shows quadratic convergence.

It is also interesting to observe how the method behaves in strongly convection-dominated problems. In Figure 5, the solutions for the problem with different diffusion coefficients are compared. We can observe that when convection grows, the corresponding boundary layer becomes thinner, and the mesh is no longer capable of capturing the boundary layer geometry. Because of the fact that the weak formulation does not weight the boundary conditions against convection velocity in the outflow boundary, we can observe that there are no spurious oscillations in the outflow boundary

layer. As the diffusion decreases, the solution of the problem resembles the solution of the pure transport equation, where no boundary conditions are imposed on the outflow.

6.2. Stokes problem

In this section, we solve the Stokes problem, and we check the convergence properties of the proposed method. We study the stationary Stokes flow around a cylinder. We use linear interpolations both for the velocity and pressure and the stabilized formulation proposed in the previous sections. The setting of the problem is shown in Figure 6. A parabolic inflow profile with unitary mean horizontal velocity is set on $x = 0$. Velocity is prescribed to zero on $y = 0$ and $y = 1$ and on the cylindrical boundary. The proposed method for weakly imposing boundary conditions has been used both in the immersed cylindrical interface, and also in the external grid matching boundaries, even if these conditions could have been imposed exactly there.

In Figure 7, velocity and pressure fields for a fine mesh are shown. In Figure 8, we have plotted the error versus the mesh size, both for the velocity and pressure fields. The coarsest mesh is built of 625 elements, whereas the finer one is built out of 40,000 elements. Again, results for each mesh size have been compared against results in a much finer mesh (160,000 elements). We can see that quadratic convergence rates are obtained in both cases.

6.3. Transient Navier–Stokes equations

In this section, we deal with the transient incompressible Navier–Stokes equations. As in the previous subsection, we will solve the flow around a cylinder, although the overall domain is larger in this case in order to allow the development of the vortices which arise behind the cylinder. The setting of the problem is depicted in In Figure 9. A parabolic inflow profile with mean horizontal velocity equal to 1 is set on $x = 0$. Velocity is prescribed to zero at $y = 0$, $y = 8$, and the cylindrical boundary. The proposed method for weakly imposing boundary conditions has been used both in the immersed cylindrical interface and in the external grid matching boundaries. Viscosity has been set to $\nu = 10^{-2}$, which yields a Reynolds number $Re = 100$ based on the diameter of the cylinder and the mean inflow velocity. A backward Euler scheme has been used for the time integration with time step $\delta t = 0.2$. A 12,566 linear element mesh has been used to solve the problem. The mesh has been refined in the area around the cylinder, but it is still a rather coarse mesh in which the length of the cylinder is only 12 times the element length.

In Figure 10, velocity and pressure fields at the end of the simulation ($t = 100$) are shown. Fully developed vortices behind the cylinder and a smooth solution around the immersed boundary can be appreciated. Figure 11 shows the time history of the vertical velocity at a point behind the cylinder (10, 4). After the initial transitory stage, an oscillatory pattern of amplitude 0.6 and period 4.9 is established.

6.4. Weak imposition of boundary conditions in the pure transport equation

In this subsection, we study the pure transport equation in which only boundary conditions on the inflow are needed. We solve the problem described in the convection–diffusion subsection, but we

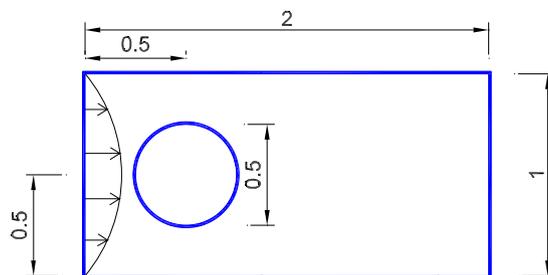


Figure 6. Geometry and boundary conditions for the Stokes flow around a cylinder.

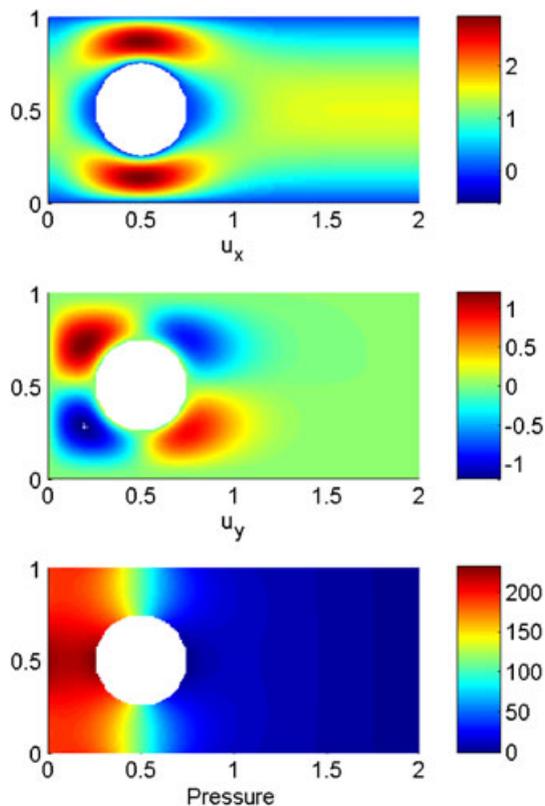


Figure 7. Velocity and pressure fields for the Stokes flow around a cylinder.

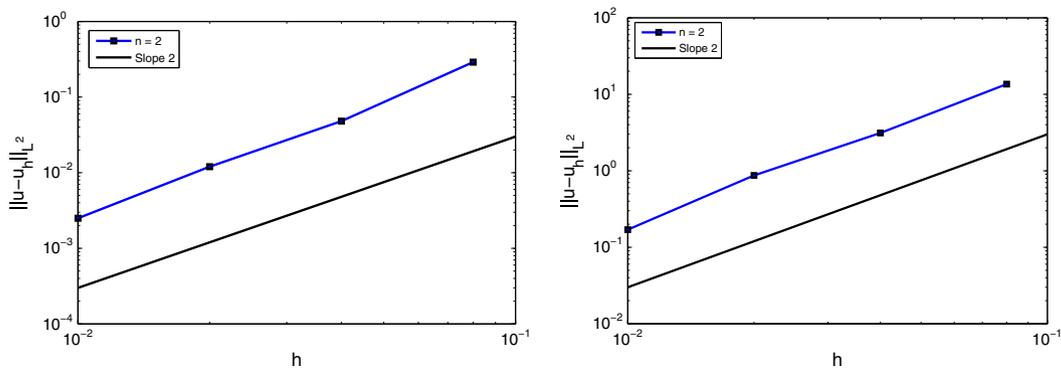


Figure 8. Convergence plots for the velocity (left) and pressure (right) fields in the Stokes flow around a cylinder.

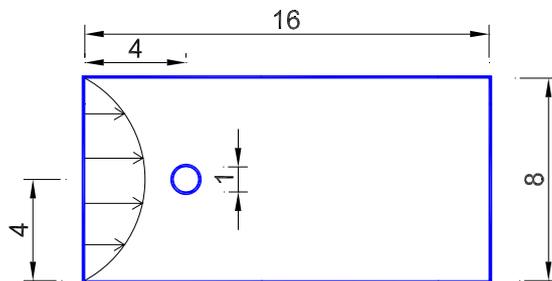


Figure 9. Geometry and boundary conditions for the transient Navier–Stokes flow around a cylinder.

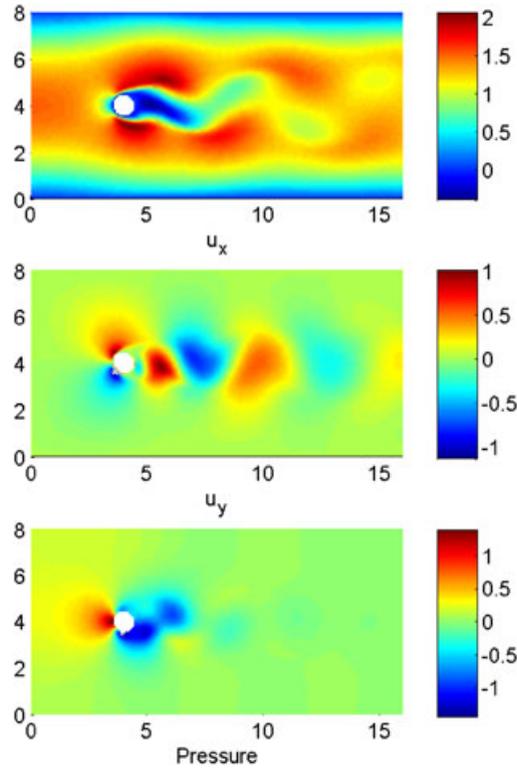


Figure 10. Velocity and pressure fields for the transient Navier–Stokes flow around a cylinder. Results at $t = 100$.

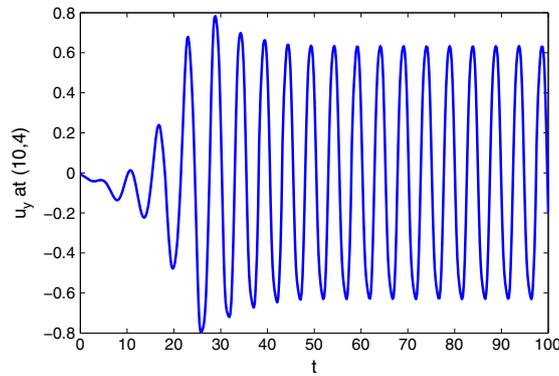


Figure 11. Vertical velocity evolution at (10, 4).

only impose boundary conditions on the inflow. Linear convergence is obtained for linear elements. The problem does not seem to be in the outflow but in the inflow, where the method does not impose boundary conditions strongly enough.

In Figure 12, we can observe the error for different meshes in the pure transport equation. It can be seen that the error diminishes linearly with the mesh size, and also, most importantly, that the computed solution is displaced, that is, the error does not oscillate around 0 but around 0.08 ($h = 1/50$), 0.04 ($h = 1/100$), and 0.02 ($h = 1/200$). This suggests that the boundary conditions are not imposed strongly enough. A specific definition of the boundary condition terms should be devised for the pure transport equation in order to obtain quadratic convergence also for this equation.

Figure 13 shows the convergence rates for the solution of the pure transport for the stabilized and the Galerkin transport equation. We can see that the convergence is linear if a stabilized formulation

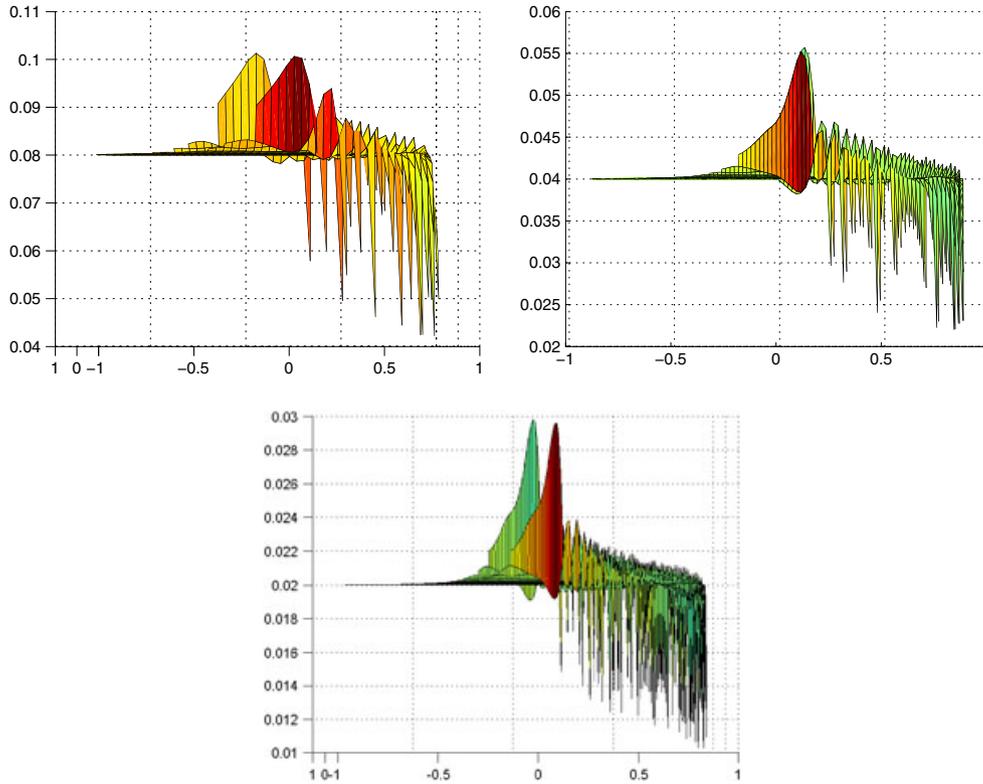


Figure 12. Error comparison ($u_h - \text{analytical}$) for the pure transport equation for $h = 1/50, 1/100, 1/200$. We can observe that the error is diminishing linearly with h . Moreover, 0.08, 0.04, and 0.02 is displaced from the 0 position. A stronger imposition of boundary conditions would improve the solution.

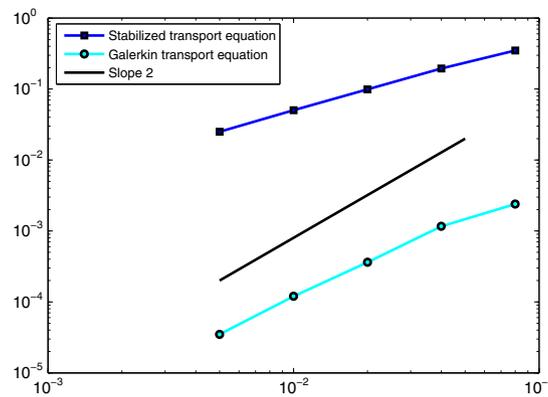


Figure 13. Convergence plots for the pure transport equation.

is used. If no stabilization is used, and the mesh nodes are aligned with the advection direction, the convergence is closer to quadratic. This suggests the use of weighting terms proportional to the stability parameters for the imposition of boundary conditions, a possibility which we will not explore here.

7. CONCLUSIONS

In this paper, we have proposed a way to weakly prescribe Dirichlet boundary conditions in embedded grids. The key feature of the proposed method is that we can ensure stability without the

need of a large, mesh-dependent penalty parameter, and that it is symmetric for symmetric problems. In the Poisson problem, this is achieved by introducing an additional element-discontinuous stress variable. Additional terms are required in order to guarantee stability in the convection–diffusion equation, in which we weight the boundary conditions with a particular norm of the convection velocity and the Stokes problem where we test the imposition of boundary conditions against the pressure test functions. The proposed strategy is then easily extended to the transient Navier–Stokes equations.

From the conceptual point of view, the method we propose can be related both to the Lagrange multiplier technique and to Nitsche’s method. The relationship with the former is that in our case, we prescribe the Lagrange multiplier to be the normal trace of the flux of the unknown, the flux condition being approximated in a least squares sense. The main benefit of our approach with respect to the use of Lagrange multipliers is that the additional variable we introduce (the flux) can be condensed at the element level, yielding a problem posed in terms of the original unknowns alone. Concerning the relationship with Nitsche’s method, it can be introduced at the purely algebraic level, simply as a different way to evaluate the penalty terms that enforce the boundary conditions. The advantage of the approach we have presented is that the algorithmic constant is independent of the geometry and approximation. Moreover, from numerical experiments, we have observed that the results are less sensitive to this parameter than in the case of Nitsche’s formulation.

The method proposed turns out to be accurate (second order for linear elements) and robust for all the problems tested except for the pure transport equation, in which we are not able to recover quadratic convergence. From the implementation point of view, the method satisfies the main design condition of using only the DOF of the mesh of Ω_h in the final system of equations to be solved. Moreover, the final resulting method is very easy to implement, because it only requires some additional boundary integrals to be added to the original variational form.

REFERENCES

1. Codina R, Houzeaux G, Coppola-Owen H, Baiges J. The fixed-mesh ALE approach for the numerical approximation of flows in moving domains. *Journal of Computational Physics* 2009; **228**:1591–1611.
2. Baiges J, Codina R. The fixed-mesh ALE approach applied to solid mechanics and fluid-structure interaction problems. *International Journal for Numerical Methods in Engineering* 2009; **81**:1529–1557.
3. Baiges J, Codina R, Coppola-Owen H. The fixed-mesh ALE approach for the numerical simulation of floating solids. *International Journal for Numerical Methods in Fluids* 2011; **67**(8):1004–1023.
4. Gerstenberger A, Wall WA. An extended finite element method/Lagrange multiplier based approach for fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(19–20):1699–1714.
5. Gerstenberger A, Wall WA. Enhancement of fixed-grid methods towards complex fluid-structure interaction applications. *International Journal for Numerical Methods in Fluids* 2008; **57**:1227–1248.
6. Wall WA, Garmnitzer P, Gerstenberger A. Fluid-structure interaction approaches on fixed grids based on two different domain decomposition ideas. *International Journal of Computational Fluid Dynamics* 2008; **22**:411–427.
7. Shahmiri S, Gerstenberger A, Wall WA. An XFEM based embedding mesh technique for incompressible viscous flows. *International Journal for Numerical Methods in Fluids* 2011; **65**:166–190.
8. Peskin CS. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics* 1972; **10**:252–271.
9. Lai M-C, Peskin CS. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *Journal of Computational Physics* 2000; **160**:705–719.
10. Leveque RJ, Li Z. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Journal on Numerical Analysis* 1994; **31**(4):1019–1044.
11. Lee L, Leveque RJ. An immersed interface method for incompressible Navier–Stokes equations. *SIAM Journal on Scientific and Statistical Computing* 2003; **25**(3):832–856.
12. Xu S, Wang ZJ. An immersed interface method for simulating the interaction of a fluid with moving boundaries. *Journal of Computational Physics* 2006; **216**:454–493.
13. Babuška I. Error bounds for finite element method. *Numerische Mathematik* 1971; **16**:322–333.
14. Dolbow J, Mourad HM, Harari I. A bubble-stabilized finite element method for Dirichlet constraints on embedded interfaces. *International Journal for Numerical Methods in Engineering* 2007; **69**:772–793.
15. Barbosa HJC, Hughes TJR. The finite element method with Lagrangian multipliers on the boundary: circumventing the Babuška-Brezzi condition. *Computer Methods in Applied Mechanics and Engineering* 1991; **85**:109–128.
16. Ji H, Dolbow JE. On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method. *International Journal for Numerical Methods in Engineering* 2004; **61**:2508–2535.

17. Glowinski R, Pan T-W, Périaux J. A fictitious domain method for Dirichlet problems and applications. *Computer Methods in Applied Mechanics and Engineering* 1994; **111**:283–303.
18. Glowinski R, Pan T-W, Hesla TI, Joseph DD, Périaux J. A distributed Lagrange multiplier/fictitious domain method for flows around moving rigid bodies: application to particulate flow. *International Journal for Numerical Methods in Fluids* 1999; **30**:1043–1066.
19. Glowinski R. Finite element methods for incompressible viscous flows. In *Numerical Methods for Fluids (Part 3), Handbook of Numerical Analysis*, Vol. 9. Elsevier: North-Holland, 2003.
20. Haslinger J, Renard Y. A new fictitious domain approach inspired by the extended finite element method. *SIAM Journal on Numerical Analysis* 2009; **47**(2):1474–1499.
21. Gilmanov A, Sotiropoulos F. A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies. *Journal of Computational Physics* 2005; **207**:457–492.
22. Ferziger JH, Tseng YH. A ghost-cell immersed boundary method for flow in complex geometry. *Journal of Computational Physics* 2003; **192**:593–623.
23. Mohd-Yusof J. Combined immersed boundaries/B-splines methods for simulations of flows in complex geometries. *CTR Annual Research Briefs*, Stanford University, NASA Ames, 1997.
24. Codina R, Baiges J. Approximate imposition of boundary conditions in immersed boundary methods. *International Journal for Numerical Methods in Engineering* 2009; **80**:1379–1405.
25. Lew AJ, Buscaglia GC. A discontinuous Galerkin-based immersed boundary method. *International Journal for Numerical Methods in Engineering* 2008; **76**:427–454.
26. Hansbo A, Hansbo P. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**:5537–5552.
27. Fernández-Méndez S, Huerta A. Imposing essential boundary conditions in mesh-free methods. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(12–14):1257–1275.
28. Nitsche J. Über ein Variationsprinzip zu Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematisches Seminar der Universität* 1971; **36**:9–15.
29. Dolbow J, Harari I. An efficient finite element method for embedded interface problems. *International Journal for Numerical Methods in Engineering* 2009; **78**(2):229–252.
30. Gerstenberger A, Wall WA. An embedded Dirichlet formulation for 3D continua. *International Journal for Numerical Methods in Engineering* 2010; **82**:537–563.
31. Mayer UM, Gerstenberger A, Wall WA. Interface handling for three-dimensional higher-order XFEM-computations in fluid-structure interaction. *International Journal for Numerical Methods in Engineering* 2009; **79**:846–869.
32. Stenberg R. Analysis of mixed finite element methods for the Stokes problem: a unified approach. *Mathematics of Computation* 1984; **42**:9–23.
33. Stenberg R. On some techniques for approximating boundary conditions in the finite element method. *Journal of Computational and Applied Mathematics* 1995; **63**:139–148.
34. Bazilevs Y, Hughes TJR. Weak imposition of Dirichlet boundary conditions in fluid mechanics. *Computers and Fluids* 2007; **36**:12–26.
35. Hughes TJR. Multiscale phenomena: Green's function, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized formulations. *Computer Methods in Applied Mechanics and Engineering* 1995; **127**:387–401.
36. Codina R. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**:2681–2706.