Mixed finite element projection methods for the unsteady Stokes equations

Costanza Aricò [a,*], Rainer Helmig [b], Ivan Yotov [c]

[a] *Department of Engineering, University of Palermo, viale delle Scienze, Palermo, 90128, Italy*
[b] *Institute for Modelling Hydraulic and Environmental Systems (IWS), Department of Hydromechanics and Modelling of Hydrosystems, University of Stuttgart, Pfaffenwaldring 61, Stuttgart, D-70569, Germany*
[c] *Department of Mathematics, University of Pittsburgh, 301 Thackeray Hall, Pittsburgh, PA 15260, USA*

ARTICLE INFO

ABSTRACT

We develop $H(\mathrm{div})$-conforming mixed finite element methods for the unsteady Stokes equations modeling single-phase incompressible fluid flow. A projection method in the framework of the incremental pressure correction methodology is applied, where a predictor problem and a corrector problem are sequentially solved, accounting for the viscous effects and incompressibility, respectively. The predictor problem is based on a stress–velocity mixed formulation, while the corrector projection problem uses a velocity–pressure mixed formulation. The scheme results in pointwise divergence-free velocity computed at the end of each time step. We establish unconditional stability and first order in time accuracy. In the implementation we focus on generally unstructured triangular grids. We employ a second order multipoint flux mixed finite element method based on the next-to-the-lowest order Raviart–Thomas space RT_1 and a suitable quadrature rule. In the predictor problem this approach allows for a local stress elimination, resulting in element-based systems for each velocity component with three degrees of freedom per element. Similarly, in the corrector problem, the velocity is locally eliminated and an element-based system for the pressure is solved. At the end of each time step we obtain a second order accurate $H(\mathrm{div})$-conforming piecewise linear velocity, which is pointwise divergence free. We present a series of numerical tests to illustrate the performance of the method.

1. Introduction

The unsteady Stokes equations, which approximate the Navier–Stokes equations for flows at small Reynolds numbers, are representative of physical processes where the time scale of the viscous effects is much smaller than the time scale of the inertial effects, in which case the transient inertial term dominates the advection inertial term [1]. Some examples of unsteady Stokes flows are: unsteady creeping flows [1,2], microswimmers flows [3], and pulsating cerebrospinal fluid flows [4–7].

One of the main issues in the numerical solution of the incompressible Stokes equations is that a saddle point problem needs to be solved, where the pressure plays the role of a Lagrange multiplier for imposing the divergence-free constraint. For the solution of unsteady incompressible flow one of the most successful approaches is provided by the "projection methods", which started in the late 1960s with the two independent works of Chorin [8] and Temam [9]. Instead of solving a coupled saddle point system for the velocity and pressure unknowns at each time step, an intermediate velocity field is computed first, using the momentum equation and neglecting the incompressibility constraint. This intermediate velocity is projected to the space of (weakly) divergence-free velocity

fields in the second step, to get the next update for the velocity and pressure. Detailed discussions of the projection methods can be found, for example, in the review papers [10,11].

The classical projection methods are based on the velocity–pressure formulation of the governing equations, which has long been the mainstream of incompressible flow computations. The associated finite element methods use inf-sup stable finite element pairs, such as the Taylor-Hood, the Crouzeix–Raviart, or the MINI elements [12]. The projection step involves the solution of a Poisson problem for the pressure with continuous finite elements. As a result, the computed velocity is not pointwise divergence-free. In this paper we develop $H(\text{div})$-conforming mixed finite element projection methods for the unsteady Stokes equations with pointwise divergence-free velocity. The predictor problem is based on a stress–velocity mixed formulation, while the corrector projection problem uses a velocity–pressure mixed formulation. We consider the Raviart–Thomas RT_k, $k \geq 0$ or Brezzi–Douglas–Marini BDM_k, $k \geq 1$ families of spaces for the viscous stress components in the predictor problem and the velocity in the projection problem. In either case, the computed velocity at the end of each step is $H(\text{div})$-conforming and divergence-free piecewise polynomial of degree $\leq k$. Our family of methods is motivated by the projection method developed in [13] for the Navier–Stokes equations, see also [14,15], which uses a finite volume method for the predicted velocity and RT_0-based two point flux approximation (TPFA) method for the corrected velocity. We note that there has been extensive work on developing numerical methods for Stokes flow with divergence-free velocity and/or pressure-robustness (independence of the velocity error of the viscosity and pressure), see e.g., [16–21]. These approaches are based on the velocity–pressure formulation and require solving a saddle point problem. Our projection method provides an efficient alternate to obtain a divergence-free velocity using a stable pair of mixed finite element spaces for the Poisson problem in the projection step. Furthermore, our proposed implementation through the use of multipoint flux mixed finite element methods requires solving only symmetric and positive definite systems.

The stress–velocity mixed finite element method for the Stokes and Navier–Stokes equations has received increased attention in recent years, see, e.g. [22–25]. Advantages of this approach include unified framework for Newtonian and non-Newtonian flows, local momentum balance, and direct $H(\text{div})$-conforming computation of the stress, the latter being paramount for modeling fluid flow around an obstacle. A disadvantage of this formulation is that it results in increased number of degrees of freedom. This issue is addressed in [26], where a multipoint stress mixed finite element method for Stokes flow is developed, based on the stress–velocity–vorticity formulation. There, a suitable choice of mixed finite element spaces and quadrature rules allows for a local elimination of the stress and vorticity, resulting in a positive definite system for the velocity. In all of the aforementioned works, the pressure is eliminated using the incompressibility constraint. As a result, mass conservation is only weakly imposed. In addition, the formulation involves the deviatoric part of the stress and the velocity approximation is discontinuous. In contrast, in our projection method, the pressure gradient in the momentum balance equation solved in the predictor problem is taken from the previous time step and does not need to be eliminated, hence the method involves the full stress. Moreover, while the predicted velocity is discontinuous, the corrected velocity computed in the projection step is $H(\text{div})$-conforming and exactly divergence-free.

In the first part of the paper we develop the family of mixed finite element projection methods for the unsteady Stokes equations of general polynomial degree. We perform stability analysis for the fully discrete method, establishing unconditional stability. The analysis is complicated by the fact that the predicted and the corrected velocity live in different finite element spaces, the former being discontinuous, and the latter $H(\text{div})$-conforming. Critical components of the analysis are the inclusion in the scheme of a variable approximating the pressure gradient, which lives in the space of the predicted velocity, and the fact that divergence-free vectors in either BDM_K or RT_k are piecewise polynomials of degree $\leq k$. We then proceed with the time-discretization error analysis, focusing on the semi-discrete continuous-in-space scheme. We establish first order convergence in time for the viscous stress and the corrected velocity.

In the second part of the paper we develop a specific method from the family introduced in the first part, based on the next-to-the-lowest order Raviart–Thomas velocity space RT_1 and discontinuous piecewise linear polynomials on generally unstructured triangular grids. This results in a method that is second order accurate in space. In order to avoid solving saddle point problems, we employ the methodology of multipoint flux mixed finite element (MFMFE) methods developed in [27] for the Poisson problem. The approach is based on choosing a suitable combination of a finite element space for the vector flux variable and a quadrature rule for the associated bilinear form, which leads to a block-diagonal mass matrix with small local blocks. The flux can therefore be locally eliminated, resulting in a symmetric and positive definite algebraic system for the scalar potential variable. In [27], the lowest order Brezzi–Douglas–Marini BDM_1 space on simplicial or quadrilateral grids is utilized, together with a vertex quadrature rule. The blocks in the flux mass matrix are associated with mesh vertices and the resulting system for the potential variable, after local flux elimination, is of finite volume type with one constant value per element. The method is first order accurate in the L^2-norm and it is related to the multipoint flux approximation (MPFA) method [28,29]. A similar mass-lumping approach is studied in [30] and an alternative formulation based on a broken RT_0 velocity space is developed in [31]. The MFMFE methodology is extended to 3D hexahedral grids in [32,33], to higher order spaces in [34,35], and to the Stokes equations using a vorticity–velocity–pressure formulation in [36].

In our method we utilize the second order MFMFE methodology on triangular grids developed in [35,37], which is based on the RT_1 flux space and a quadrature rule that samples the functions at the degrees of freedom, which are the normal components at each edge evaluated at the two edge endpoints and the vector evaluated at the element center of mass. We employ this approach in both the predictor and the projection problems. In the predictor problem, we split the momentum equation into two separate equations for the x- and y-components of the predicted velocity. In each sub-problem, we locally eliminate the viscous stress component and obtain a sparse symmetric and positive definite system for the predicted velocity component. Similarly, in the projection problem, we locally eliminate the corrected velocity and we solve a sparse symmetric and positive definite system for the pressure. All three systems involve three unknowns per element describing the linear variation within the element. The global stencil couples each element with

all elements that share a vertex with it. The viscous stress and corrected velocity are easily computed by local postprocessing. The resulting method is highly computationally efficient, involving the solution of three sparse symmetric and positive definite systems per time step, which is done with the preconditioned conjugate gradient method.

We remark that in physical problems, including transport processes, fluid–structure interaction, free fluid–porous medium interface transfer, or turbulence modeling, the second order accuracy of the velocity and pressure would allow for a higher physical fidelity in the approximation. In this sense, the present paper could also be regarded as the seed of further extensions where more terms are included in the governing momentum equations, e.g., convective inertia or Brinkman term or coupling with transport or thermal effects, as in [13–15].

The paper is organized as follows. In Section 2 we present the governing equations and the numerical method. Stability and time discretization error analysis is performed in Section 3. The second order multipoint flux mixed finite element method on triangular grids is developed in Section 4. In Section 5 we report the results of a series of numerical experiments that illustrate the performance of the method. These include verification of the second order convergence in space and first order convergence in time, two benchmark problems, where the results of the method are compared to numerical and analytical solutions provided in the literature, and a challenging application with a strongly irregular computational domain. We end with conclusions in Section 6.

2. Governing equations and numerical method

We assume a Newtonian, single-phase, incompressible fluid with density ρ. Let $\Omega \subset \mathbf{R}^d$, $d = 2, 3$, be the computational domain. The unsteady Stokes equations are

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \nabla \Psi = 0 \quad \text{in } \Omega \times (0, T], \tag{2.1a}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times (0, T], \tag{2.1b}$$

where T is the final time, $\mathbf{u}$ is the fluid velocity vector, $\Psi = \frac{p}{\rho}$ is the kinematic pressure, $\nu = \frac{\mu}{\rho}$ is the kinematic fluid viscosity and μ the dynamic fluid viscosity. Let Γ be the boundary of Ω, with a unit outward normal vector $\mathbf{n}$. Two types of boundary conditions are assigned over $\Gamma = \Gamma_d \cup \Gamma_n$, where Γ_d and Γ_n are the portions of Γ where we assign the velocity vector and the normal component of the stress, respectively. The problem is complemented with initial conditions for the velocity and pressure. The boundary and initial conditions are:

$$\mathbf{u} = \mathbf{u}_b \quad \text{on } \Gamma_d, \ t \in [0, T], \tag{2.2a}$$

$$(-\nu \nabla \mathbf{u} + \Psi \mathbf{I}) \mathbf{n} = \mathbf{\Sigma}_b \quad \text{on } \Gamma_n, \ t \in [0, T], \tag{2.2b}$$

$$\mathbf{u} = \mathbf{u}_0 \quad \text{with} \quad \nabla \cdot \mathbf{u}_0 = 0, \quad \Psi = \Psi_0 \quad \text{in } \Omega, \ t = 0, \tag{2.2c}$$

where $\mathbf{I}$ the identity matrix.

2.1. Time discretization

We discretize the time interval $[0, T]$ by a partition with N sub-intervals of length $\Delta t = T/N$ with vertices t_n, $n = 0, \dots, N$, $t_n = n\Delta t$. We denote by φ^n the value of variable φ computed at time t_n. System (2.1) is solved at each discrete time t_{n+1} by applying an incremental pressure correction scheme [11], where a predictor and a projection problem are solved sequentially, such that the time-discretization form of (2.1a) splits into

$$\frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\Delta t} - \nu \Delta \tilde{\mathbf{u}}^{n+1} + \nabla \Psi^n = 0, \tag{2.3a}$$

$$\frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1}}{\Delta t} + \nabla \left(\Psi^{n+1} - \Psi^n \right) = 0, \tag{2.3b}$$

where (2.3a) is the predictor problem and (2.3b) is the projection problem.

2.2. Space discretization

The space discretization is based on a mixed variational formulation of the system (2.3). For this purpose we introduce the variables

$$\boldsymbol{\sigma} = -\nu \nabla \mathbf{u}, \quad \mathbf{q} = \nabla \Psi.$$

Here $\boldsymbol{\sigma}$ is the viscous pseudostress, a $d \times d$ tensor with rows σ_i^T, $\sigma_i = \nabla u_i$, $i = 1, \dots, d$, where u_i is the ith component of the velocity $\mathbf{u}$. With this notation the predictor problem (2.3a) can be written as

$$\frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\Delta t} + \nabla \cdot \boldsymbol{\sigma}^{n+1} + \mathbf{q}^n = 0. \tag{2.4}$$

In the variational formulation of (2.3) we will use the space $H(\text{div}, \Omega)$, with

$$H(\text{div}, \Omega) = \left\{ \mathbf{v} \in (L_2(\Omega))^d : \ \nabla \cdot \mathbf{v} \in L_2(\Omega) \right\}, \tag{2.5}$$

and define

$$\mathbf{V} = H(\text{div}, \Omega), \quad \mathbf{V}_{0,\Gamma_s} = \left\{ \mathbf{v} \in \mathbf{V} : \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \Gamma_s \right\}, \quad s \in \{d, n\}, \quad W = L_2(\Omega). \tag{2.6}$$

For a domain $D \subset \mathbf{R}^d$, the symbol $(\cdot, \cdot)_D$ marks the $L^2(D)$-inner product and $\| \cdot \|_D$ denotes the $L^2(D)$-norm for scalar, vector, and tensor valued functions. We also use the standard Hilbert space notation $H^r(D)$ with a norm $\| \cdot \|_{r,D}$, where r is a positive integer. We will omit the subscript D if $D = \Omega$. For $S \in \mathbf{R}^{d-1}$ we denote by $\langle \cdot, \cdot \rangle_S$ the $L^2(S)$ inner product or duality pairing.

We discretize Ω by a geometrically conforming partition T_h (also called grid or mesh) made of N_T non-overlapping triangles E called (computational) elements or cells. Two neighboring elements share a common side (or interface) e. Let $\mathfrak{S}_T$ be the number of edges.

Let $\mathbf{V}_h \times W_h \subset \mathbf{V} \times W$ be either the Raviart–Thomas (RT) or the Brezzi–Douglas–Marini (BDM) pairs of spaces [12] on T_h. These are pairs of inf-sup stable mixed finite element spaces with the property

$$\nabla \cdot \mathbf{V}_h = W_h. \tag{2.7}$$

Let $(\mathbf{V}_h)^d$ denote the tensor-valued space where each row is an element of $\mathbf{V}_h$. Let $Q_h : L^2(\Omega) \to W_h$ and $\boldsymbol{Q}_h : (L^2(\Omega))^d \to \mathbf{V}_h$ be the L^2-orthogonal projection operator onto W_h, such that for any $w \in L^2(\Omega)$,

$$(Q_h w - w, w_h) = 0 \quad \forall w_h \in W_h. \tag{2.8}$$

Let $Q_h^\Gamma : L^2(\Gamma) \to \mathbf{V}_h \cdot \mathbf{n}$ and $\boldsymbol{Q}_h^\Gamma : (L^2(\Gamma))^d \to (\mathbf{V}_h)^d \mathbf{n}$ be the L^2-orthogonal projection operators onto $\mathbf{V}_h \cdot \mathbf{n}$ and $(\mathbf{V}_h)^d \mathbf{n}$, respectively, such that for any $\varphi \in L^2(\Gamma)$ and any $\mathbf{v} \in (L^2(\Gamma))^d$,

$$\langle \varphi - Q_h^\Gamma \varphi, \mathbf{v}_h \cdot \mathbf{n} \rangle_\Gamma = 0 \quad \forall \mathbf{v}_h \in \mathbf{V}_h,$$

$$\langle \mathbf{v} - \boldsymbol{Q}_h^\Gamma \mathbf{v}, \boldsymbol{\tau}_h \mathbf{n} \rangle_\Gamma = 0 \quad \forall \boldsymbol{\tau}_h \in (\mathbf{V}_h)^d.$$

We will also use the mixed interpolant $\boldsymbol{\Pi}_h : (H^1(\Omega))^d \to \mathbf{V}_h$ [12] satisfying for all $\mathbf{v} \in (H^1(\Omega))^d$,

$$(\nabla \cdot (\boldsymbol{\Pi}_h \mathbf{v} - \mathbf{v}), w_h) = 0 \quad \forall w_h \in W_h, \tag{2.9}$$

$$\langle (\boldsymbol{\Pi}_h \mathbf{v} - \mathbf{v}) \cdot \mathbf{n}, \mathbf{v}_h \cdot \mathbf{n} \rangle_\Gamma = 0 \quad \forall \mathbf{v}_h \in \mathbf{V}_h. \tag{2.10}$$

We will also utilize the vector version of Q_h, $Q_h^d : (L^2(\Omega))^d \to (W_h)^d$ and the tensor version of $\boldsymbol{\Pi}_h$, $\boldsymbol{\Pi}_h^d : (H^1(\Omega))^{d \times d} \to (\mathbf{V}_h)^d$.

2.3. The mixed finite element projection method

The numerical algorithm is as follows.

- Initialization: Let $\mathbf{u}_h^0 = \boldsymbol{\Pi}_h \mathbf{u}_0$, $\Psi_h^0 = Q_h \Psi_0$, $\mathbf{q}_h^0 = Q_h^d \nabla \Psi_0$, $\Psi_b^0 = \Psi^0|_\Gamma$.

For $n = 0, \ldots, N - 1$:

- Predictor problem (MFE discretization of (2.4)):
  Find $\boldsymbol{\sigma}_h^{n+1} \in (\mathbf{V}_h)^d : \boldsymbol{\sigma}_h^{n+1} \mathbf{n} = \boldsymbol{Q}_h^\Gamma (\Sigma_b^{n+1} - \Psi_b^n \mathbf{n})$ on Γ_n and $\tilde{\mathbf{u}}_h^{n+1} \in (W_h)^d$, such that

$$(\nu^{-1} \boldsymbol{\sigma}_h^{n+1}, \boldsymbol{\tau}_h) - (\tilde{\mathbf{u}}_h^{n+1}, \nabla \cdot \boldsymbol{\tau}_h) = -\langle \mathbf{u}_b^{n+1}, \boldsymbol{\tau}_h \mathbf{n} \rangle_{\Gamma_d} \quad \forall \boldsymbol{\tau}_h \in (\mathbf{V}_{h,0,\Gamma_n})^d, \tag{2.11a}$$

$$\left(\frac{\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^n}{\Delta t}, \boldsymbol{\xi}_h \right) + (\nabla \cdot \boldsymbol{\sigma}_h^{n+1}, \boldsymbol{\xi}_h) + (\mathbf{q}_h^n, \boldsymbol{\xi}_h) = 0 \quad \forall \boldsymbol{\xi}_h \in (W_h)^d. \tag{2.11b}$$

- Compute Ψ_b^{n+1} on Γ_n:

$$\Psi_b^{n+1} = \left(\Sigma_b^{n+1} + \nu \nabla \tilde{\mathbf{u}}_h^{n+1} \mathbf{n} \right) \cdot \mathbf{n} \quad \text{on} \quad \Gamma_n.$$

- Projection problem (MFE discretization of (2.3b)):
  Find $\mathbf{u}_h^{n+1} \in \mathbf{V}_h : \mathbf{u}_h^{n+1} \cdot \mathbf{n} = Q_h^\Gamma (\mathbf{u}_b \cdot \mathbf{n})$ on Γ_d and $\Psi_h^{n+1} \in W_h$ such that

$$\left(\frac{\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}}{\Delta t}, \mathbf{v}_h \right) - (\Psi_h^{n+1} - \Psi_h^n, \nabla \cdot \mathbf{v}_h) = -\langle \Psi_b^{n+1} - \Psi_b^n, \mathbf{v}_h \cdot \mathbf{n} \rangle_{\Gamma_n} \quad \forall \mathbf{v}_h \in \mathbf{V}_{h,0,\Gamma_d}, \tag{2.12a}$$

$$(\nabla \cdot \mathbf{u}_h^{n+1}, w_h) = 0 \quad \forall w_h \in W_h. \tag{2.12b}$$

- Update the pressure gradient: Find $\mathbf{q}_h^{n+1} \in (W_h)^d$ such that

$$(\mathbf{q}_h^{n+1}, \boldsymbol{\xi}_h) = (\mathbf{q}_h^n, \boldsymbol{\xi}_h) - \left(\frac{\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}}{\Delta t}, \boldsymbol{\xi}_h \right) \quad \forall \boldsymbol{\xi}_h \in (W_h)^d. \tag{2.13}$$

- Go to the next time step.

Remark 2.1.   The predictor problem (2.11) accounts for the viscous effects. It uses the pressure gradient $\mathbf{q}^n$ computed at the previous time step. The intermediate velocity $\tilde{\mathbf{u}}_h^{n+1} \in (W_h)^d$ computed after the predictor problem is discontinuous. The problem also provides $H(\mathrm{div})$-conforming viscous pseudostress $\boldsymbol{\sigma}_h^{n+1} \in (\mathbf{V}_h)^d$.

The projection problem (2.12) computes the new pressure $\Psi_h^{n+1} \in W_h$ and corrects the velocity, resulting in $H(\mathrm{div})$-conforming and pointwise divergence free $\mathbf{u}_h^{n+1} \in \mathbf{V}_h$. In particular, due to (2.7), (2.12b) implies

$$\nabla \cdot \mathbf{u}_h^{n+1} = 0. \tag{2.14}$$

We note that for both the RT and BDM pairs, it holds that

$$\mathbf{u}_h^{n+1} \in (W_h)^d. \tag{2.15}$$

The above property is true for the BDM pairs, since $\mathbf{V}_h$ and W_h contain polynomials of the same degree. It also holds for the RT pairs, due to (2.14) and [12, Corollary 2.3.1].

The update on $\mathbf{q}_h^{n+1}$ (2.13) provides an approximation of $\nabla\Psi(t_{n+1})$, since, together with (2.15) and (2.12a), it implies

$$\mathbf{q}_h^{n+1} = \mathbf{q}_h^n - \frac{\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}}{\Delta t} \simeq \nabla\Psi(t_n) + \nabla\big(\Psi(t_{n+1}) - \Psi(t_n)\big) = \nabla\Psi(t_{n+1}).$$

3. Stability and time discretization error analysis

In this section we establish unconditional stability and first order accuracy in time for the projection method (2.11)–(2.13). In the analysis we focus on the case of homogeneous Dirichlet boundary conditions $\mathbf{u} = 0$ on Γ. We will use the notation

$$\mathbf{V}_0 = \{\mathbf{v} \in \mathbf{V} : \mathbf{v} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega\}.$$

In this case, in order to have uniqueness of the pressure Ψ, its space is restricted to

$$W_0 = \left\{ w \in W : \int_\Omega w = 0 \right\}.$$

The corresponding mixed finite element spaces are denoted by $\mathbf{V}_{h,0}$ and $W_{h,0}$. We note that for this choice of spaces, it holds that

$$\nabla \cdot \mathbf{V}_{h,0} = W_{h,0}. \tag{3.1}$$

With this choice of boundary conditions, the algorithm (2.11)–(2.13) takes the form

- Predictor problem: Find $\boldsymbol{\sigma}_h^{n+1} \in (\mathbf{V}_h)^d$ and $\tilde{\mathbf{u}}_h^{n+1} \in (W_h)^d$, such that

$$\left(\nu^{-1}\boldsymbol{\sigma}_h^{n+1}, \boldsymbol{\tau}_h\right) - \left(\tilde{\mathbf{u}}_h^{n+1}, \nabla \cdot \boldsymbol{\tau}_h\right) = 0 \quad \forall \boldsymbol{\tau}_h \in (\mathbf{V}_h)^d, \tag{3.2a}$$

$$\left(\frac{\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^n}{\Delta t}, \boldsymbol{\xi_h}\right) + \left(\nabla \cdot \boldsymbol{\sigma}_h^{n+1}, \boldsymbol{\xi_h}\right) + \left(\mathbf{q}_h^n, \boldsymbol{\xi_h}\right) = 0 \quad \forall \boldsymbol{\xi_h} \in (W_h)^d. \tag{3.2b}$$

- Projection problem: Find $\mathbf{u}_h^{n+1} \in \mathbf{V}_{h,0}$ and $\Psi_h^{n+1} \in W_{h,0}$ such that

$$\left(\frac{\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}}{\Delta t}, \mathbf{v}_h\right) - (\Psi_h^{n+1} - \Psi_h^n, \nabla \cdot \mathbf{v}_h) = 0 \quad \forall \mathbf{v}_h \in \mathbf{V}_{h,0}, \tag{3.3a}$$

$$\left(\nabla \cdot \mathbf{u}_h^{n+1}, w_h\right) = 0 \quad \forall w_h \in W_{h,0}. \tag{3.3b}$$

- Update the pressure gradient: Find $\mathbf{q}_h^{n+1} \in (W_h)^d$ such that

$$\left(\mathbf{q}_h^{n+1}, \boldsymbol{\xi}_h\right) = \left(\mathbf{q}_h^n, \boldsymbol{\xi}_h\right) - \left(\frac{\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}}{\Delta t}, \boldsymbol{\xi}_h\right) \quad \forall \boldsymbol{\xi}_h \in (W_h)^d. \tag{3.4}$$

Remark 3.1.   Due to (3.1), (3.3b) implies that (2.14) still holds.

3.1. Stability analysis

In the analysis, we will utilize the algebraic identity

$$a(a - b) = \frac{1}{2}\left(a^2 - b^2 + (a - b)^2\right). \tag{3.5}$$

Let $\mathbf{V}^0 = \{\mathbf{v} \in \mathbf{V} : \nabla \cdot \mathbf{v} = 0\}$ denote the divergence-free subspace of $\mathbf{V}$, with a similar notation for $\mathbf{V}_h^0$ and $\mathbf{V}_{h,0}^0$. We have the following useful orthogonality property.

Lemma 3.1.   *For $\mathbf{q}_h^{n+1}$ computed in (3.4), it holds that*

$$(\mathbf{q}_h^{n+1}, \mathbf{v}_h) = 0 \quad \forall \mathbf{v}_h \in \mathbf{V}_{h,0}^0. \tag{3.6}$$

Proof. Recalling the argument for (2.15), we have that $\mathbf{V}_h^0 \subset (W_h)^d$, so we can test (3.4) with $\xi_h = \mathbf{v}_h \in \mathbf{V}_{h,0}^0$, which, together with (3.3a), implies

$$(\mathbf{q}_h^{n+1}, \mathbf{v}_h) = (\mathbf{q}_h^n, \mathbf{v}_h) = \cdots = (\mathbf{q}_h^0, \mathbf{v}_h) = (Q_h^d \nabla \Psi_0, \mathbf{v}_h) = (\nabla \Psi_0, \mathbf{v}_h) = -(\Psi_0, \nabla \cdot \mathbf{v}_h) + \langle \Psi_0, \mathbf{v}_h \cdot \mathbf{n} \rangle_\Gamma = 0. \quad \square$$

We next establish a stability bound for the projection method.

Theorem 3.1. *For the method* (3.2)–(3.4), *there exists a constant C independent of Δt and h such that*

$$\Delta t \sum_{n=0}^{N-1} v^{-1} \|\boldsymbol{\sigma}_h^{n+1}\|^2 + \frac{1}{2}\|\mathbf{u}_h^N\|^2 + \frac{1}{2}\sum_{n=0}^{N-1} \|\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}\|^2 + \frac{\Delta t^2}{2}\|\mathbf{q}_h^N\|^2 \le \frac{1}{2}\|\mathbf{u}_h^0\|^2 + \frac{\Delta t^2}{2}\|\mathbf{q}_h^0\|^2. \tag{3.7}$$

Proof. We start by taking test functions $(\boldsymbol{\tau}_h, \xi_h) = (\boldsymbol{\sigma}_h^{n+1}, \tilde{\mathbf{u}}_h^{n+1})$ in (3.2), combining the equations, using (3.5), and multiplying by Δt, to obtain

$$\Delta t \, v^{-1}\|\boldsymbol{\sigma}_h^{n+1}\|^2 + \frac{1}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 - \frac{1}{2}\|\mathbf{u}_h^n\|^2 + \frac{1}{2}\|\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^n\|^2 + \Delta t \,(\mathbf{q}_h^n, \tilde{\mathbf{u}}_h^{n+1}) = 0. \tag{3.8}$$

Next, we take $\mathbf{v}_h = \mathbf{u}_h^{n+1}$ in (3.3a) and use (3.5) and (2.14) to obtain

$$\frac{1}{2}\|\mathbf{u}_h^{n+1}\|^2 - \frac{1}{2}\|\tilde{\mathbf{u}}_h^{n+1}\|^2 + \frac{1}{2}\|\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}\|^2 = 0. \tag{3.9}$$

Taking $\xi_h = \mathbf{q}_h^{n+1}$, using (3.5) and multiplying by Δt^2 results in

$$\frac{\Delta t^2}{2}\left(\|\mathbf{q}_h^{n+1}\|^2 - \|\mathbf{q}_h^n\|^2 + \|\mathbf{q}_h^{n+1} - \mathbf{q}_h^n\|^2\right) = -\Delta t (\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}, \mathbf{q}_h^{n+1}) = \Delta t (\tilde{\mathbf{u}}_h^{n+1}, \mathbf{q}_h^{n+1}), \tag{3.10}$$

where we used (3.6) in the last equality. Summing (3.8)–(3.10), we obtain

$$\Delta t \, v^{-1}\|\boldsymbol{\sigma}_h^{n+1}\|^2 + \frac{1}{2}\left(\|\mathbf{u}_h^{n+1}\|^2 - \|\mathbf{u}_h^n\|^2\right) + \frac{1}{2}\left(\|\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^n\|^2 + \|\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}\|^2\right)$$

$$+ \frac{\Delta t^2}{2}\left(\|\mathbf{q}_h^{n+1}\|^2 - \|\mathbf{q}_h^n\|^2 + \|\mathbf{q}_h^{n+1} - \mathbf{q}_h^n\|^2\right) = \Delta t \,(\mathbf{q}_h^n - \mathbf{q}_h^n, \tilde{\mathbf{u}}_h^{n+1}). \tag{3.11}$$

For the term on the right hand side above, using (3.6) and Young's inequality $ab \le \frac{\epsilon}{2}a^2 + \frac{1}{2\epsilon}b^2$ for any $\epsilon > 0$, we write

$$\Delta t \,(\mathbf{q}_h^{n+1} - \mathbf{q}_h^n, \tilde{\mathbf{u}}_h^{n+1}) = \Delta t \,(\mathbf{q}_h^{n+1} - \mathbf{q}_h^n, \tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^n) \le \frac{\Delta t^2}{2}\|\mathbf{q}_h^{n+1} - \mathbf{q}_h^n\|^2 + \frac{1}{2}\|\tilde{\mathbf{u}}_h^{n+1} - \mathbf{u}_h^n\|^2. \tag{3.12}$$

Bound (3.7) follows by combining (3.11)–(3.12) and summing over n from 0 to $N-1$. $\quad \square$

3.2. Time discretization error analysis

In this section we establish a bound on the time discretization error in the projection method. For this purpose, we consider the semi-discrete continuous-in-space formulation of the method (3.2)–(3.4) given below.

- Initialization: Let $\mathbf{u}^0 = \mathbf{u}_0$, $\Psi^0 = \Psi_0$, $\mathbf{q}^0 = \nabla \Psi_0$.
  For $n = 0, \ldots, N-1$:
- Predictor problem: Find $\boldsymbol{\sigma}^{n+1} \in (\mathbf{V})^d$ and $\tilde{\mathbf{u}}^{n+1} \in (W)^d$, such that

$$\left(v^{-1}\boldsymbol{\sigma}^{n+1}, \boldsymbol{\tau}\right) - \left(\tilde{\mathbf{u}}^{n+1}, \nabla \cdot \boldsymbol{\tau}\right) = 0 \quad \forall \boldsymbol{\tau} \in (\mathbf{V})^d, \tag{3.13a}$$

$$\left(\frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\Delta t}, \xi\right) + \left(\nabla \cdot \boldsymbol{\sigma}^{n+1}, \xi\right) + (\mathbf{q}^n, \xi) = 0 \quad \forall \xi \in (W)^d. \tag{3.13b}$$

- Projection problem: Find $\mathbf{u}^{n+1} \in \mathbf{V}_0$ and $\Psi^{n+1} \in W_0$ such that

$$\left(\frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1}}{\Delta t}, \mathbf{v}\right) - (\Psi^{n+1} - \Psi^n, \nabla \cdot \mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0, \tag{3.14a}$$

$$\left(\nabla \cdot \mathbf{u}^{n+1}, w\right) = 0 \quad \forall w \in W_0. \tag{3.14b}$$

- Update the pressure gradient: Find $\mathbf{q}^{n+1} \in (W)^d$ such that

$$\left(\mathbf{q}^{n+1}, \xi\right) = (\mathbf{q}^n, \xi) - \left(\frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1}}{\Delta t}, \xi\right) \quad \forall \xi \in (W)^d. \tag{3.15}$$

As noted in Remark 3.1, (3.14b) implies that

$$\nabla \cdot \mathbf{u}^{n+1} = 0. \tag{3.16}$$

We next note that the solution to the model problem (2.1) with boundary condition $\mathbf{u} = 0$ on Γ satisfies, for $n = 0, \ldots, N - 1$,

$$\left(v^{-1}\boldsymbol{\sigma}(t_{n+1}), \boldsymbol{\tau}\right) - \left(\mathbf{u}(t_{n+1}), \nabla \cdot \boldsymbol{\tau}\right) = 0 \quad \forall \boldsymbol{\tau} \in (\mathbf{V})^d, \tag{3.17}$$

$$\left(\frac{\mathbf{u}(t_{n+1}) - \mathbf{u}(t_n)}{\Delta t}, \boldsymbol{\xi}\right) + \left(\nabla \cdot \boldsymbol{\sigma}(t_{n+1}), \boldsymbol{\xi}\right) + \left(\nabla \Psi(t_{n+1}), \boldsymbol{\xi}\right) = (T_{n+1}(\mathbf{u}), \boldsymbol{\xi}) \quad \forall \boldsymbol{\xi} \in (W)^d, \tag{3.18}$$

$$\nabla \cdot \mathbf{u}(t_{n+1}) = 0, \tag{3.19}$$

where

$$T_{n+1}(\mathbf{u}) = \frac{\mathbf{u}(t_{n+1}) - \mathbf{u}(t_n)}{\Delta t} - \frac{\partial \mathbf{u}}{\partial t}(t_{n+1}).$$

Theorem 3.2. *For the semi-discrete method (3.13)–(3.15), assuming that the true solution is sufficiently smooth in time, there exists a constant C independent of Δt such that*

$$\Delta t \sum_{n=0}^{N-1} v^{-1} \|\boldsymbol{\sigma}(t_{n+1}) - \boldsymbol{\sigma}^{n+1}\|^2 + \|\mathbf{u}(t_N) - \mathbf{u}^N\|^2 + \Delta t^2 \|\nabla \Psi(t_N) - \mathbf{q}^N\|^2 \le C\Delta t^2. \tag{3.20}$$

Proof. Let $e_{\boldsymbol{\sigma}}^n = \boldsymbol{\sigma}(t_n) - \boldsymbol{\sigma}^n$, $e_{\tilde{\mathbf{u}}}^n = \mathbf{u}(t_n) - \tilde{\mathbf{u}}^n$, $e_{\mathbf{u}}^n = \mathbf{u}(t_n) - \mathbf{u}^n$, and $e_{\mathbf{q}}^n = \nabla \Psi(t_n) - \mathbf{q}^n$. Subtracting (3.13a)–(3.13b), (3.16) from (3.17)–(3.19) gives the error equations

$$\left(v^{-1}e_{\boldsymbol{\sigma}}^{n+1}, \boldsymbol{\tau}\right) - \left(e_{\tilde{\mathbf{u}}}^{n+1}, \nabla \cdot \boldsymbol{\tau}\right) = 0 \quad \forall \boldsymbol{\tau} \in (\mathbf{V})^d, \tag{3.21}$$

$$\left(\frac{e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^n}{\Delta t}, \boldsymbol{\xi}\right) + \left(\nabla \cdot e_{\boldsymbol{\sigma}}^{n+1}, \boldsymbol{\xi}\right) + \left(e_{\mathbf{q}}^n, \boldsymbol{\xi}\right) = (T_{n+1}(\mathbf{u}), \boldsymbol{\xi}) - (S_{n+1}(\nabla \Psi), \boldsymbol{\xi}) \quad \forall \boldsymbol{\xi} \in (W)^d, \tag{3.22}$$

$$\nabla \cdot e_{\mathbf{u}}^{n+1} = 0, \tag{3.23}$$

where

$$S_{n+1}(\nabla \Psi) = \nabla \Psi(t_{n+1}) - \nabla \Psi(t_n).$$

Taking $(\boldsymbol{\tau}, \boldsymbol{\xi}) = (e_{\boldsymbol{\sigma}}^{n+1}, e_{\tilde{\mathbf{u}}}^{n+1})$ in (3.21)–(3.22), combining the equations, using (3.5), and multiplying by Δt, we obtain

$$\Delta t\, v^{-1} \|e_{\boldsymbol{\sigma}}^{n+1}\|^2 + \frac{1}{2}\|e_{\tilde{\mathbf{u}}}^{n+1}\|^2 - \frac{1}{2}\|e_{\mathbf{u}}^n\|^2 + \frac{1}{2}\|e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^n\|^2 + \Delta t\,(e_{\mathbf{q}}^n, e_{\tilde{\mathbf{u}}}^{n+1}) = \Delta t(T_{n+1}(\mathbf{u}), e_{\tilde{\mathbf{u}}}^{n+1}) - \Delta t(S_{n+1}(\nabla \Psi), e_{\tilde{\mathbf{u}}}^{n+1}). \tag{3.24}$$

Next, we subtract and add $\mathbf{u}(t_{n+1})$ in (3.14a), take $\mathbf{v} = e_{\mathbf{u}}^{n+1}$, and use (3.23), to obtain

$$(e_{\mathbf{u}}^{n+1} - e_{\tilde{\mathbf{u}}}^{n+1}, e_{\mathbf{u}}^{n+1}) = \Delta t(\Psi^{n+1} - \Psi^n, \nabla \cdot e_{\mathbf{u}}^{n+1}) = 0,$$

which implies, using (3.5)

$$\frac{1}{2}\|e_{\mathbf{u}}^{n+1}\|^2 - \frac{1}{2}\|e_{\tilde{\mathbf{u}}}^{n+1}\|^2 + \frac{1}{2}\|e_{\mathbf{u}}^{n+1} - e_{\tilde{\mathbf{u}}}^{n+1}\|^2 = 0. \tag{3.25}$$

Next, subtracting and adding $\nabla \Psi(t_{n+1})$, $\nabla \Psi(t_n)$, and $\mathbf{u}(t_{n+1})$ in (3.15), multiplying by Δt^2, and taking $\boldsymbol{\xi} = e_{\mathbf{q}}^{n+1}$ results in

$$\Delta t^2(e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n, e_{\mathbf{q}}^{n+1}) + \Delta t(e_{\mathbf{u}}^{n+1} - e_{\tilde{\mathbf{u}}}^{n+1}, e_{\mathbf{q}}^{n+1}) = \Delta t^2(S_{n+1}(\nabla \Psi), e_{\mathbf{q}}^{n+1}). \tag{3.26}$$

We observe that the argument for (3.6) implies

$$(\mathbf{q}^n, \mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0^0,$$

which, combined with

$$(\nabla \Psi, \mathbf{v}) = -(\Psi, \nabla \cdot \mathbf{v}) + \langle \Psi, \mathbf{v} \cdot \mathbf{n} \rangle_\Gamma = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0^0, \tag{3.27}$$

implies

$$(e_{\mathbf{q}}^n, \mathbf{v}) = 0 \quad \forall \mathbf{v} \in \mathbf{V}_0^0. \tag{3.28}$$

Therefore, using that $e_{\mathbf{u}}^{n+1} \in \mathbf{V}_0^0$, cf. (3.23), (3.26) reduces to

$$\Delta t^2(e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n, e_{\mathbf{q}}^{n+1}) - \Delta t(e_{\tilde{\mathbf{u}}}^{n+1}, e_{\mathbf{q}}^{n+1}) = \Delta t^2(S_{n+1}(\nabla \Psi), e_{\mathbf{q}}^{n+1}).$$

which, combined with (3.5), implies

$$\frac{\Delta t^2}{2}\left(\|e_{\mathbf{q}}^{n+1}\|^2 - \|e_{\mathbf{q}}^n\|^2 + \|e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n\|^2\right) - \Delta t(e_{\tilde{\mathbf{u}}}^{n+1}, e_{\mathbf{q}}^{n+1}) = \Delta t^2(S_{n+1}(\nabla \Psi), e_{\mathbf{q}}^{n+1}). \tag{3.29}$$

The next step is to sum (3.24), (3.25), and (3.29). For the sum of the last terms on left hand sides of (3.24) and (3.29), using (3.28), we write

$$-(e_{\tilde{\mathbf{u}}}^{n+1}, e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n) = (e_{\mathbf{u}}^n - e_{\tilde{\mathbf{u}}}^{n+1}, e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n). \tag{3.30}$$

Thus, summing (3.24), (3.25), and (3.29) and using (3.30) we arrive at

$$\Delta t \, v^{-1} \|e_\sigma^{n+1}\|^2 + \frac{1}{2} \left(\|e_{\mathbf{u}}^{n+1}\|^2 - \|e_{\mathbf{u}}^n\|^2 + \|e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^n\|^2 + \|e_{\mathbf{u}}^{n+1} - e_{\tilde{\mathbf{u}}}^{n+1}\|^2 \right) + \frac{\Delta t^2}{2} \left(\|e_{\mathbf{q}}^{n+1}\|^2 - \|e_{\mathbf{q}}^n\|^2 + \|e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n\|^2 \right)$$

$$= -\Delta t(e_{\mathbf{u}}^n - e_{\tilde{\mathbf{u}}}^n, e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n) + \Delta t(T_{n+1}(\mathbf{u}), e_{\tilde{\mathbf{u}}}^{n+1}) - \Delta t(S_{n+1}(\nabla \Psi), e_{\tilde{\mathbf{u}}}^{n+1}) + \Delta t^2(S_{n+1}(\nabla \Psi), e_{\mathbf{q}}^{n+1}) =: I_1 + I_2 + I_3 + I_4. \tag{3.31}$$

We proceed with bounding the four terms on the right hand side. For I_1, using Young's inequality, we have

$$|I_1| \le \frac{1}{2} \|e_{\mathbf{u}}^n - e_{\tilde{\mathbf{u}}}^{n+1}\|^2 + \frac{\Delta t^2}{2} \|e_{\mathbf{q}}^{n+1} - e_{\mathbf{q}}^n\|^2. \tag{3.32}$$

For I_2, using Young's inequality, we write

$$|I_2| = |\Delta t(T_{n+1}(\mathbf{u}), e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^{n+1}) + \Delta t(T_{n+1}(\mathbf{u}), e_{\mathbf{u}}^{n+1})| \le \Delta t^2 \|T_{n+1}(\mathbf{u})\|^2 + \frac{1}{4} \|e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^{n+1}\|^2 + \Delta t \|T_{n+1}(\mathbf{u})\|^2 + \frac{\Delta t}{4} \|e_{\mathbf{u}}^{n+1}\|^2. \tag{3.33}$$

For I_3, using (3.27) and (3.23), we have that $(S_{n+1}(\nabla \Psi), e_{\mathbf{u}}^{n+1}) = 0$. Then, using Young's inequality, we obtain

$$|I_3| = |\Delta t(S_{n+1}(\nabla \Psi), e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^{n+1})| \le \Delta t^2 \|S_{n+1}(\nabla \Psi)\|^2 + \frac{1}{4} \|e_{\tilde{\mathbf{u}}}^{n+1} - e_{\mathbf{u}}^{n+1}\|^2. \tag{3.34}$$

For I_4, the use of Young's inequality gives

$$|I_4| \le \Delta t \|S_{n+1}(\nabla \Psi)\|^2 + \frac{\Delta t^3}{4} \|e_{\mathbf{q}}^{n+1}\|^2. \tag{3.35}$$

Combining (3.31)–(3.35) gives

$$\Delta t \, v^{-1} \|e_\sigma^{n+1}\|^2 + \frac{1}{2} \left(\|e_{\mathbf{u}}^{n+1}\|^2 - \|e_{\mathbf{u}}^n\|^2 \right) + \frac{\Delta t^2}{2} \left(\|e_{\mathbf{q}}^{n+1}\|^2 - \|e_{\mathbf{q}}^n\|^2 \right)$$

$$\le \left(\Delta t^2 + \Delta t \right) \|T_{n+1}(\mathbf{u})\|^2 + \left(\Delta t^2 + \Delta t \right) \|S_{n+1}(\nabla \Psi)\|^2 + \frac{\Delta t}{4} \|e_{\mathbf{u}}^{n+1}\|^2 + \frac{\Delta t^3}{4} \|e_{\mathbf{q}}^{n+1}\|^2. \tag{3.36}$$

It is straightforward to show for the time discretization and splitting errors that $\|T_{n+1}(\mathbf{u})\| \le C\Delta t$ and $\|S_{n+1}(\nabla \Psi)\| \le C\Delta t$, assuming that the solution is smooth enough in time. Then (3.20) follows by summing (3.36) over n from 0 to $N-1$, using that $e_{\mathbf{u}}^0 = 0$ and $e_{\mathbf{q}}^0 = 0$, and, assuming that $\Delta t \le 1$, applying the discrete Gronwall inequality [38, Lemma 1.4.2] for the last two terms in (3.36). □

4. A second order multipoint flux mixed finite element method

We next describe our specific implementation of the mixed finite element projection method introduced in Section 2. It is based on the next-to-the lowest order Raviart–Thomas spaces RT_1 on triangular grids [39], which results in a method with second order accuracy in space. Moreover, we employ the multipoint flux mixed finite element methodology developed in [27] as a first order method and extended in [35] to a second order method using the RT_1 spaces [39]. Applying a quadrature rule with nodes associated with the degrees of freedom of $\mathbf{V}_h$, we obtain mass lumping in both the predictor and projection problems. In the predictor problem, we locally eliminate the viscous stress and solve a symmetric and positive definite system for each component of the predicted velocity. In the projection problem, we locally eliminate the corrected velocity and we solve a symmetric and positive definite system for the pressure. The viscous stress and corrected velocity are then easily computed by local postprocessing. The resulting method is computationally very efficient. It provides a second order accurate $H(\text{div})$-conforming velocity, which is linear on each triangle and pointwise divergence free.

4.1. Mixed finite element spaces

We consider a partition T_h of the computational domain into non-overlapping triangles, where h is the maximal element diameter. Let $\hat{E}$ be the reference triangle with vertices $\hat{\mathbf{r}}_1 = (0,0)^T$, $\hat{\mathbf{r}}_2 = (1,0)^T$, and $\hat{\mathbf{r}}_3 = (0,1)^T$. For any triangle $E \in T_h$ with counterclockwise oriented vertices $\mathbf{r}_i$, $i = 1,2,3$, there exists a bijection linear mapping $F_E : \hat{E} \to E$ such that $F_E(\hat{\mathbf{r}}_i) = \mathbf{r}_i$, $i = 1,2,3$, given by

$$F_E(\hat{\mathbf{x}}) = \mathbf{r}_1 (1 - \hat{x} - \hat{y}) + \mathbf{r}_2 \hat{x} + \mathbf{r}_3 \hat{y}. \tag{4.1}$$

We take the pair $\{\mathbf{V}_h \times W_h\}$ on T_h introduced in Section 2.2 to be the RT_1 spaces [39]. In the reference triangle $\hat{E}$ these spaces are defined as

$$\mathbf{V}(\hat{E}) = \left(P_1(\hat{E}) \right)^2 + \mathbf{x} P_1(\hat{E}), \tag{4.2a}$$

$$W(\hat{E}) = P_1(\hat{E}), \tag{4.2b}$$

with $P_k(\hat{E})$ being the space of bivariate polynomials of degree $\le k$ on $\hat{E}$. The dimension of $\mathbf{V}_1(\hat{E})$ is 8. On any triangle, we can take two normal component degrees of freedom (DOFs) associated with each edge and two DOFs associated with the vector at the center of mass [12]. On the reference triangle, we denote the center of mass as $\hat{\mathbf{r}}_4 = (1/3, 1/3)^T$. We choose the two DOFs associated to each edge e be the values of $\mathbf{v} \cdot \mathbf{n}$ at the two vertices of the edge [27]. This will allow us, through the use of a quadrature rule, to localize the interaction of the edge DOFs around mesh vertices. For $k = 1,2,3$, let $\hat{\mathbf{n}}_{k,l}$, $l = 1,2$, denote the outward unit normal vector to the two edges that share vertex $\hat{\mathbf{r}}_k$. For the center of mass $\hat{\mathbf{r}}_4$, let $\hat{\mathbf{n}}_{4,1} = (1 \ 0)^T$ and $\hat{\mathbf{n}}_{4,2} = (0 \ 1)^T$, see Fig. 1(a). For each of the points $\hat{\mathbf{r}}_i$, $i = 1, \ldots, 4$, the two associated RT_1 basis functions $\hat{\mathbf{v}}_{i,j}$, $j = 1,2$, are defined by setting

$$\hat{\mathbf{v}}_{i,j}(\hat{\mathbf{r}}_k) \cdot \hat{\mathbf{n}}_{k,l} = \delta_{ik} \delta_{jl}, \qquad i,k = 1, \ldots, 4, \quad j,l = 1,2, \tag{4.3}$$
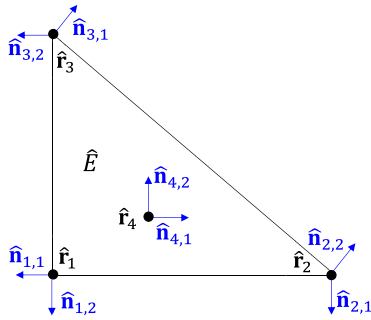
$\hat{\mathbf{v}}_{1,1}$	$-2\hat{x}^2 - \hat{x}\hat{y} + 3\hat{x} + \hat{y} - 1$	$-\hat{y}^2 - 2\hat{x}\hat{y} + \hat{y}$
$\hat{\mathbf{v}}_{1,2}$	$-\hat{x}^2 - 2\hat{x}\hat{y} + \hat{x}$	$-2\hat{y}^2 - \hat{x}\hat{y} + \hat{x} + 3\hat{y} - 1$
$\hat{\mathbf{v}}_{2,1}$	$\hat{x}^2 - \hat{x}\hat{y}$	$-\hat{y}^2 + \hat{x}\hat{y} - \hat{x} + \hat{y}$
$\hat{\mathbf{v}}_{2,2}$	$\sqrt{2}(2\hat{x}^2 + \hat{x}\hat{y} - \hat{x})$	$\sqrt{2}(\hat{y}^2 + 2\hat{x}\hat{y} - \hat{y})$
$\hat{\mathbf{v}}_{3,1}$	$\sqrt{2}(\hat{x}^2 + 2\hat{x}\hat{y} - \hat{x})$	$\sqrt{2}(2\hat{y}^2 + \hat{x}\hat{y} - \hat{y})$
$\hat{\mathbf{v}}_{3,2}$	$-\hat{x}^2 + \hat{x}\hat{y} + \hat{x} - \hat{y}$	$\hat{y}^2 - \hat{x}\hat{y}$
$\hat{\mathbf{v}}_{4,1}$	$-3(\hat{x}\hat{y}) - 6\ (\hat{x}^2 - \hat{x})$	$-3(\hat{y}^2 - \hat{y}) - 6\ (\hat{x}\hat{y})$
$\hat{\mathbf{v}}_{4,2}$	$-6(\hat{x}\hat{y}) - 3\ (\hat{x}^2 - \hat{x})$	$-6(\hat{y}^2 - \hat{y}) - 3\ (\hat{x}\hat{y})$

(a) Degrees of freedom of $\mathbf{V}(\hat{E})$.   (b) Basis functions of $\mathbf{V}(\hat{E})$.

Fig. 1. Degrees of freedom of $\mathbf{V}(\hat{E})$ and the associated basis functions.

where $\delta_{\alpha\beta} = 1$ if $\alpha = \beta$, $\delta_{\alpha\beta} = 0$ if $\alpha \neq \beta$. The basis functions are listed in Fig. 1(b), see also [35], where in the second and third column of the table we give their x and y components.

The RT_1 spaces on any element $E \in T_h$ are defined via the transformations

$$\mathbf{v} \longleftrightarrow \hat{\mathbf{v}} : \mathbf{v} = \frac{1}{|\mathbf{J}_E|} \mathbf{J}_E \hat{\mathbf{v}}, \tag{4.4a}$$

$$w \longleftrightarrow \hat{w} : w = \hat{w}, \tag{4.4b}$$

where (4.4a) is known as the Piola transformation. Here $\mathbf{J}_E = [\mathbf{r}_{21}, \mathbf{r}_{31}]$ is the Jacobian matrix associated with the mapping (4.1), with $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. The determinant of the Jacobian matrix is $|\mathbf{J}_E| = 2|E|$, and $|E|$ is the area of triangle E. The Piola transformation has the properties [12]

$$\nabla \cdot \mathbf{v} = \frac{1}{|\mathbf{J}_E|} \nabla_{\hat{\mathbf{x}}} \cdot \hat{\mathbf{v}}, \quad \mathbf{v} \cdot \mathbf{n}_e = \frac{1}{|e|} \hat{\mathbf{v}} \cdot \hat{\mathbf{n}}_e, \tag{4.5}$$

where $F_E : \hat{e} \to e$, $\mathbf{n}_e$ and $\hat{\mathbf{n}}_e$ are the unit normal vectors on the edges e and $\hat{e}$, respectively, and $|e|$ is the length of e. Since the Piola transformation preserves the normal components of the vector, it is suitable to enforce the continuity of $\mathbf{v} \cdot \mathbf{n}$ across any edge e [12].

The RT_1 spaces are defined on T_h as

$$\mathbf{V}_h = \left\{ \mathbf{v} \in \mathbf{V} : \mathbf{v}|_E \longleftrightarrow \hat{\mathbf{v}}, \ \hat{\mathbf{v}} \in \hat{\mathbf{V}}(\hat{E}) \ \forall E \in T_h \right\}, \tag{4.6a}$$

$$W_h = \left\{ w \in W : w|_E \longleftrightarrow \hat{w}, \ \hat{w} \in \hat{W}(\hat{E}) \ \forall E \in T_h \right\}, \tag{4.6b}$$

where the transformations $\mathbf{v} \longleftrightarrow \hat{\mathbf{v}}$ and $w \longleftrightarrow \hat{w}$ are defined in (4.4). In Fig. 2 we show the mapping $F_E : \hat{E} \to E$ with the DOFs of the RT_1 spaces, where the DOFs for W_h, which is discontinuous piecewise linear, are the values at any three points within an element E.

4.2. The quadrature rule

Given any pair of vector functions $\mathbf{s}, \mathbf{v} \in \mathbf{V}_h$, we define the quadrature rule

$$(\mathbf{s}, \mathbf{v})_Q = \sum_{E \in T_h} (\mathbf{s}, \mathbf{v})_{Q,E} = \sum_{E \in T_h} |E| \sum_{i=1}^{4} \omega_i \, \mathbf{s}\left(\mathbf{r}_i\right) \cdot \mathbf{v}\left(\mathbf{r}_i\right), \tag{4.7}$$

where the quadrature weights are $\omega_i = 1/12$, $i = 1, 2, 3$, and $\omega_4 = 3/4$. This quadrature rule is proposed in [35] and it is second order accurate. The vertex vector $\mathbf{v}(\mathbf{r}_i)$, $i = 1, 2, 3$, can be uniquely obtained from the degrees of freedom $\mathbf{v}(\mathbf{r}_i) \cdot \mathbf{n}_{i,1}$ and $\mathbf{v}(\mathbf{r}_i) \cdot \mathbf{n}_{i,2}$, i.e., its normal components to the two edges sharing vertex $\mathbf{r}_i$. Similarly, the vector at the center of mass $\mathbf{r}_4$ can be obtained from the degrees of freedom $\mathbf{v}(\mathbf{r}_4) \cdot \mathbf{n}_{4,1}$ and $\mathbf{v}(\mathbf{r}_4) \cdot \mathbf{n}_{4,2}$.

We apply the quadrature rule to the bilinear form $(\nu^{-1}\sigma_h^{n+1}, \tau_h)$ in (2.11a) in the prediction problem, and to the bilinear form $\left(\frac{\mathbf{u}_h^{n+1} - \tilde{\mathbf{u}}_h^{n+1}}{\Delta t}, \mathbf{v}_h\right)$ in (2.12a) in the projection problem. Since the quadrature rule (4.7) couples only the two basis functions associated with the quadrature point $\mathbf{r}_i$, see [27] for more details, the viscous stress $\sigma_{x(y),h}^{n+1}$ and the velocity $\mathbf{u}_h^{n+1}$ can be locally eliminated, resulting in symmetric and positive definite systems for $\tilde{\mathbf{u}}_h^{k+1}$ in (2.11) and Ψ_h^{n+1} in (2.12), respectively.

4.3. Predictor problem

We solve separately for the x and y components of (2.11). Let $\sigma_{x,h}^{n+1}$ and $\sigma_{y,h}^{n+1}$ denote the first and second rows of σ_h^{n+1}, respectively. Let $\tilde{u}_{x,h}^{n+1}$ and $\tilde{u}_{y,h}^{n+1}$ denote the x and y components of $\tilde{\mathbf{u}}_h^{n+1}$, with a similar notation for $q_{x,h}^n$ and $q_{y,h}^n$. For the boundary data, let $\Sigma_b = (\Sigma_{x,b}, \Sigma_{y,b})^T$ and $\mathbf{u}_b = (u_{x,b}, u_{y,b})^T$. For brevity, we present our method only for the x-component.
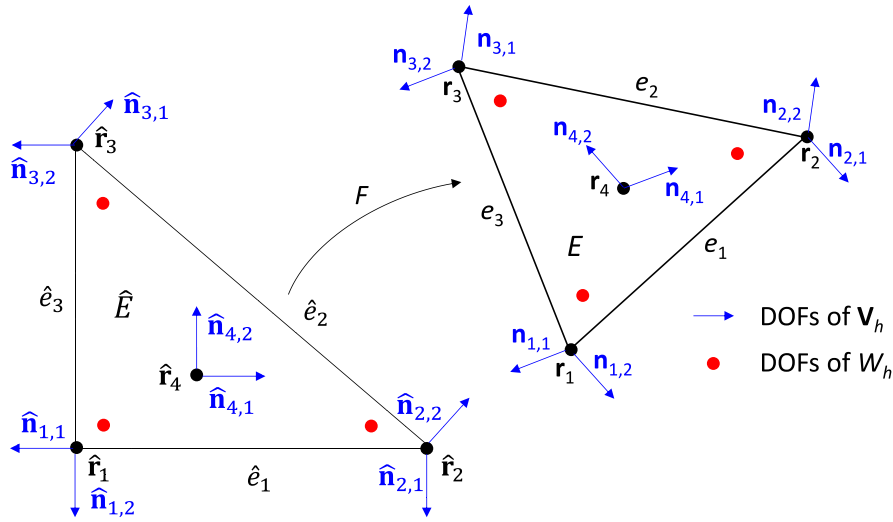
Fig. 2. Mapping from the reference triangle $\hat{E}$ to a computational triangle E with DOFs of RT_1 in $\hat{E}$ and E.
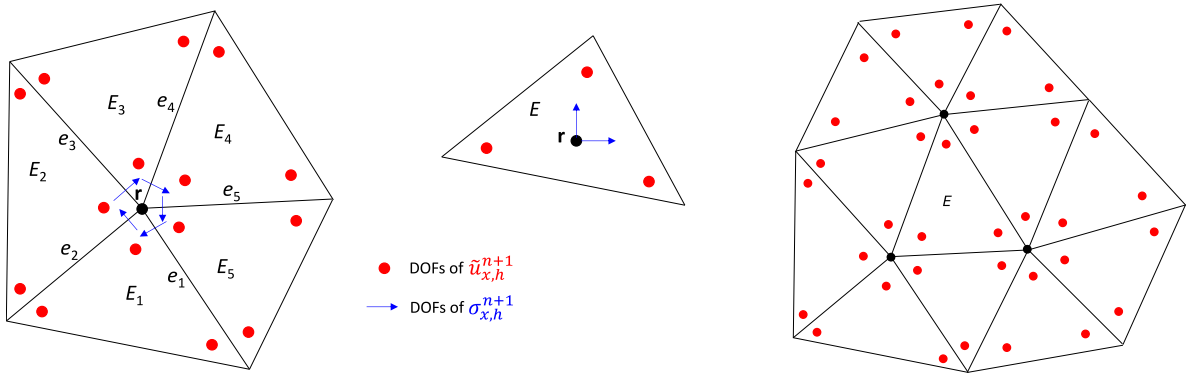

Fig. 3. Interaction of the DOFs of $\sigma_{x,h}^{n+1}$ for a vertex (left) and a center of mass (center). Stencil of the reduced system for $\tilde{\mathbf{u}}_h^{n+1}$ (right).

With the quadrature rule introduced in (4.7), the x-component of problem (2.11) becomes: find $\sigma_{x,h}^{n+1} \in \mathbf{V}_h$: $\sigma_{x,h}^{n+1} \cdot \mathbf{n} = Q_h^\Gamma(\Sigma_{x,b}^{n+1} - \Psi_b^n n_x)$ on Γ_n and $\tilde{u}_{x,h}^{n+1} \in W_h$, such that

$$\left(\nu^{-1}\sigma_{x,h}^{n+1}, \tau_h\right)_Q - \left(\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_h\right) = -\langle u_{x,b}^{n+1}, \tau_h \cdot \mathbf{n}\rangle_{\Gamma_d} \quad \forall \tau_h \in \mathbf{V}_{h,0,\Gamma_n}, \tag{4.8a}$$

$$\left(\frac{\tilde{u}_{x,h}^{n+1} - u_{x,h}^n}{\Delta t}, \xi_h\right) + \left(\nabla \cdot \sigma_{x,h}^{n+1}, \xi_h\right) + \left(q_{x,h}^n, \xi_h\right) = 0 \quad \forall \xi_h \in W_h. \tag{4.8b}$$

4.3.1. Local stress elimination in the predictor problem

In this subsection we describe the local elimination of the viscous stress $\sigma_{x,h}^{n+1}$ in the predictor problem (4.8) in terms of the x-velocity $\tilde{u}_x^{n+1}$.

4.3.1.1. The case of any vertex. Consider first an interior vertex $\mathbf{r}$ shared by $\mathcal{K}$ edges and $\mathcal{K}$ elements, see Fig. 3 (left), where $\mathcal{K} = 5$. Let E_i and e_i, $i = 1, \ldots, \mathcal{K}$, be, respectively, the elements and edges sharing $\mathbf{r}$. Let $\tau_i \in \mathbf{V}_h$ be the basis functions on the edges e_i associated with vertex $\mathbf{r}$, and let $\sigma_{x,i}^{n+1}$ be the associated DOFs of $\sigma_{x,h}^{n+1}$.

Thanks to the properties of the quadrature rule discussed in Section 4.2, the bilinear form $(\nu^{-1}\cdot, \cdot)_Q$ localizes the interactions of the DOFs of $\sigma_{x,h}^{n+1}$. For example, taking $\tau_h = \tau_1$ in (4.8a) couples $\sigma_{x,1}^{n+1}$ only with $\sigma_{x,2}^{n+1}$ and $\sigma_{x,5}^{n+1}$. In a similar way, $\sigma_{x,2}^{n+1}$ is coupled only with $\sigma_{x,1}^{n+1}$ and $\sigma_{x,3}^{n+1}$, and so on. Therefore, taking $\tau_h = \tau_1, \ldots, \tau_{\mathcal{K}}$ in (4.8a) results in a $\mathcal{K} \times \mathcal{K}$ local linear system for the $\sigma_{x,i}^{n+1}$ unknowns, $i = 1, \ldots, \mathcal{K}$:

$$\sum_{j=i-1}^{i+1} \sigma_{x,j}^{n+1}\left((\nu^{-1}\tau_j, \tau_i)_{Q,E_i} + (\nu^{-1}\tau_j, \tau_i)_{Q,E_{i+1}}\right) = (\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_i)_{E_i} + (\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_i)_{E_{i+1}}, \quad i = 1, \ldots, \mathcal{K}, \tag{4.9}$$

where we set $i - 1 := \mathcal{K}$ for $i = 1$ and $i + 1 := 1$ for $i = \mathcal{K}$.

We apply a 3-point Gaussian integration rule to compute $(\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_h)_E$. It is exact for quadratic functions, and therefore it is exact for $(\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_h)_E$, since $\tilde{u}_{x,h}^{n+1} \in P_1(E)$ and $\nabla \cdot \tau_h \in P_1(E)$.

We write the local linear system (4.9) in a matrix–vector form as

$$\mathbf{A}\Sigma_x = \mathbf{B}^T \mathbf{U}_x, \tag{4.10}$$

where

- $\mathbf{A}$ is a $(\mathcal{K} \times \mathcal{K})$ matrix with coefficients $A_{i,j} = \left(v^{-1}\tau_j, \tau_i\right)_Q$, $i, j = 1, \ldots, \mathcal{K}$,
- Σ_x is a $(\mathcal{K} \times 1)$ vector with coefficients $\sigma_{x,i}^{n+1}$, $i = 1, \ldots, \mathcal{K}$,
- $\mathbf{B}^T$ is a $(\mathcal{K} \times 3\mathcal{K})$ matrix, whose coefficients come from $(\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_i)_{E_i} + (\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_i)_{E_{i+1}}$, $i = 1, \ldots, \mathcal{K}$,
- $\mathbf{U}_x$ is a $(3\mathcal{K} \times 1)$ vector, whose coefficients are the three degrees of freedom of $\tilde{u}_{x,h}^{n+1}$ within each triangle E_i, $i = 1, \ldots, \mathcal{K}$.

The $(\mathcal{K} \times \mathcal{K})$ matrix $\mathbf{A}$ is symmetric and positive definite, see [27], and therefore the local system (4.10) is solvable, which allows us to express the $\mathcal{K}$ normal components of the viscous stress DOFs $\sigma_{x,i}^{n+1}$, $i = 1, \ldots, \mathcal{K}$ that share vertex $\mathbf{r}$ in terms of the $3\mathcal{K}$ DOFs of $\tilde{u}_{x,h}^{n+1}$ on the $\mathcal{K}$ triangles that share vertex $\mathbf{r}$.

Next, consider a boundary vertex $\mathbf{r}$. In this case the number of triangles that share it is $\mathcal{K} - 1$, so $\mathbf{B}^T$ is a $(\mathcal{K} \times 3(\mathcal{K} - 1))$ matrix and $\mathbf{U}_x$ is a $(3(\mathcal{K} - 1) \times 1)$ vector. If $\mathbf{r}$ is on $\overline{\Gamma}_d$, according to (4.8a), in the local system (4.10) we also have the contribution $-\langle u_{x,b}^{n+1}, \tau_h \cdot \mathbf{n}\rangle_{\Gamma_d}$. This is computed by numerical integration over the boundary edge(s) e_i on Γ_d that share $\mathbf{r}$ and system (4.8) changes accordingly to

$$\mathbf{A}\Sigma_x = \mathbf{B}^T \mathbf{U}_x + \mathbf{G}_{d,x}, \tag{4.11}$$

where $\mathbf{G}_{d,x}$ is a $(\mathcal{K} \times 1)$ vector with nonzero coefficients only for the rows corresponding to the boundary edge(s) on Γ_d sharing $\mathbf{r}$. If $\mathbf{r}$ is a boundary vertex on $\overline{\Gamma}_n$, then the system (4.8) is modified accordingly to incorporate the essential stress boundary condition, resulting in a system of type

$$\tilde{\mathbf{A}}\Sigma_x = \tilde{\mathbf{B}}^T \mathbf{U}_x + \mathbf{G}_{n,x}, \tag{4.12}$$

where $\mathbf{G}_{n,x}$ is a $(\mathcal{K} \times 1)$ vector.

4.3.1.2. The case of any center of mass. We now consider any center of mass $\mathbf{r}$ in triangle E, see Fig. 3 (center). Let $\tau_i \in \mathbf{V}_h$, $i = 1, 2$, be the basis functions of $\sigma_{x,h}^{n+1}$ associated with $\mathbf{r}$. Let $\sigma_{x,1}^{n+1}$ and $\sigma_{x,2}^{n+1}$ be the two DOFs of $\sigma_{x,h}^{n+1}$ associated with $\mathbf{r}$, see (4.3). Let $\tilde{u}_{x,l}^{n+1}$, $l = 1, \ldots, 3$, be the three DOFs of $\tilde{u}_{x,h}^n$ in E. We recall that, due to the localization property of the quadrature rule $\left(v^{-1}\cdot, \cdot\right)_Q$, $\sigma_{x,1}^{n+1}$ and $\sigma_{x,2}^{n+1}$ are coupled only with each other. Therefore, taking $\tau_h = \tau_1$ and $\tau = \tau_2$ in (4.8a), we obtain a 2×2 linear system for $\sigma_{x,1}^{n+1}$ and $\sigma_{x,2}^{n+1}$:

$$\sum_{j=1,2} \sigma_{x,j}^{n+1} \left(v^{-1}\tau_j, \tau_i\right)_{Q,E} = \left(\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_i\right)_E \qquad i = 1, 2. \tag{4.13}$$

We write the local linear system in a matrix–vector form as $\mathbf{A}\Sigma_x = \mathbf{B}^T \mathbf{U}_x$, where

- $\mathbf{A}$ is a (2×2) matrix with coefficients $A_{i,j} = \left(v^{-1}\tau_j, \tau_i\right)_Q$, $i, j = 1, 2$,
- Σ_x is a (2×1) vector with coefficients $\sigma_{x,i}^{n+1}$, $i = 1, 2$,
- $\mathbf{B}^T$ is a (2×3) matrix, whose coefficients come from $(\tilde{u}_{x,h}^{n+1}, \nabla \cdot \tau_i)_E$, $i = 1, 2$,
- $\mathbf{U}_x$ is a (3×1) vector, whose coefficients are the velocity components $\tilde{u}_{x,l}^{n+1}$, $l = 1, \ldots, 3$.

As in the case of a vertex, the matrix $\mathbf{A}$ is symmetric and positive definite. Solving the 2×2 system allows to express the two stress DOFs $\sigma_{x,1}^{n+1}$ and $\sigma_{x,2}^{n+1}$ that share $\mathbf{r}$ in terms of the three velocity DOFs on E, $\tilde{u}_{x,l}^{n+1}$, $l = 1, \ldots, 3$.

4.3.2. Reduction of the predictor problem to a system for $\tilde{u}_x$ and its solution

With the local stress elimination presented in Section 4.3.1, the MFMFE method in the predictor problem can be reduced to a system for $\tilde{u}_{x,h}^{n+1}$ with three DOFs per element. We briefly describe this procedure. The algebraic system arising from (4.8) is

$$\begin{pmatrix} \mathbb{A} & -\mathbb{B}^T \\ \mathbb{B} & \mathbb{D} \end{pmatrix} \begin{pmatrix} \Sigma_x \\ \mathbb{U}_x \end{pmatrix} = \begin{pmatrix} \mathbb{G}_x \\ \mathbb{Q}_x + \mathbb{F}_x \end{pmatrix}, \tag{4.14}$$

where

- $\mathbb{A}$ is a $(\mathfrak{K} \times \mathfrak{K})$ block diagonal matrix, with $\mathfrak{K} = 2 \times \mathfrak{S}_T + 2 \times N_T$, $\mathfrak{S}_T$ is the total number of the edges and N_T is the total number of triangles, as specified in Section 2, which is obtained by assembling the local block matrices $\mathbf{A}$, described in 4.3.1.1 and 4.3.1.2,
- Σ_x is a $(\mathfrak{K} \times 1)$ vector with coefficients the DOFs of $\sigma_{x,h}^{n+1}$, given by assembling the local vectors Σ_x, defined in 4.3.1.1 and 4.3.1.2,
- $\mathbb{B}^T$ is a $(\mathfrak{K} \times 3N_T)$ matrix, given by assembling the block local matrices $\mathbf{B}^T$, defined in 4.3.1.1 and 4.3.1.2,
- $\mathbb{D}$ is a $(3N_T \times 3N_T)$ block diagonal matrix, with blocks associated with $\frac{1}{\Delta t}\left(\tilde{u}_{x,h}^{n+1}, \xi_h\right)_E$,

- $\mathbb{U}_x$ is a $(3N_T \times 1)$ vector with coefficients the DOFs of $\tilde{u}^{n+1}_{x,h}$,
- $\mathbb{G}_x$ is a $(\mathfrak{K} \times 1)$ vector, given by assembling the local vectors $\mathbf{G}_{d,x}$ and $\mathbf{G}_{n,x}$, defined in 4.3.1.1,
- $\mathbb{Q}_x$ and $\mathbb{F}_x$ are $(3N_T \times 1)$ vectors corresponding to $-\left(q^n_{x,h}, \xi_h\right)$ and $\frac{1}{\Delta t}\left(u^n_{x,h}, \xi_h\right)$, respectively.

Remark 4.1. Similarly to the integrals in $\mathbb{B}$, we use a 3-point Gaussian quadrature rule for computing the integrals in $\mathbb{D}$, $\mathbb{Q}_x$, and $\mathbb{F}_x$. All components in the associated bilinear forms are linear, recall (2.15), thus the integrands are quadratic. Therefore the computation of all integrals is exact.

The stress unknown Σ_x in (4.14) can be eliminated by inverting the matrix $\mathbb{A}$, which involves solving small local systems associated with the diagonal blocks of $\mathbb{A}$, as described in 4.3.1.1 and 4.3.1.2. This leads to a $(3N_T \times 3N_T)$ system for $\mathbb{U}_x$:

$$\left(\mathbb{D} + \mathbb{B}\,\mathbb{A}^{-1}\,\mathbb{B}^T\right)\mathbb{U}_x = \mathbb{Q}_x + \mathbb{F}_x + \mathbb{B}\mathbb{A}^{-1}\mathbb{G}_x, \tag{4.15}$$

where the three DOFs of $\tilde{u}^{n+1}_{x,h}$ within each triangle E are coupled with the three DOFs of all triangles that share a vertex with E, see Fig. 3 (right).

The matrix in (4.15) is symmetric and positive definite [27]. The system is solved applying a preconditioned conjugate gradient method with incomplete Cholesky factorization [40], resulting in a very efficient and fast procedure. The coefficients of the system matrix depend only on geometrical quantities, as well as on the fluid viscosity ν and the time step size Δt. Therefore the matrix is factorized only once, before the beginning of the time loop, saving a lot of computational effort.

We proceed in the same way along the y direction, solving a system with the same matrix in (4.15).

At the end of the predictor problem we obtain the predicted velocity vector $\tilde{\mathbf{u}}^{n+1}_h|_E \in (P_1(E))^2$. The normal flux continuity at the element interfaces is lost.

4.4. Computation of Ψ^{n+1}_b on Γ_n

We compute the kinematic pressure Ψ^{n+1}_b on any edge $e \in \Gamma_n$ as

$$\Psi^{n+1}_b|_e = \left(\Sigma^{n+1}_b|_e + \nu\nabla\tilde{\mathbf{u}}^{n+1}_h|_e\,\mathbf{n}\right)\cdot\mathbf{n},$$

where the tensor $\nabla\tilde{\mathbf{u}}^{n+1}_h$ is computed on the element E with edge e. We note that, since $\tilde{\mathbf{u}}^{n+1}_h \in (P_1(E))^2$, $\nabla\tilde{\mathbf{u}}^{n+1}_h$ is constant on E.

4.5. Projection problem

Using the quadrature rule in (4.7), the discrete variational formulation of the projection problem in (2.12) is: find $\mathbf{u}^{n+1}_h \in \mathbf{V}_h$: $\mathbf{u}^{n+1}_h \cdot \mathbf{n} = Q^\Gamma_h\left(\mathbf{u}_b \cdot \mathbf{n}\right)$ on Γ_d and $\Psi^{n+1}_h \in W_h$ such that

$$\left(\frac{\mathbf{u}^{n+1}_h - \tilde{\mathbf{u}}^{n+1}_h}{\Delta t}, \mathbf{v}_h\right)_Q - (\Psi^{n+1}_h - \Psi^n_h, \nabla\cdot\mathbf{v}_h) = -\langle\Psi^{n+1}_b - \Psi^n_b, \mathbf{v}_h\cdot\mathbf{n}\rangle_{\Gamma_n} \quad \forall\mathbf{v}_h \in \mathbf{V}_{h,0,\Gamma_d},$$

$$\left(\nabla\cdot\mathbf{u}^{n+1}_h, w_h\right) = 0 \quad \forall w_h \in W_h.$$

The method is the same as the predictor problem method for $\sigma^{n+1}_{x,h} \in \mathbf{V}_h$ and $\tilde{u}^{n+1}_{x,h} \in W_h$ in (4.8). We apply the same local elimination procedure as described in Sections 4.3.1 and 4.3.2, this time for the velocity $\mathbf{u}^{n+1}_h$, which results in a symmetric and positive system for Ψ^{n+1}_h of type (4.15), where the three DOFs of Ψ^{n+1}_h within each triangle E are coupled with the three DOFs of all triangles that share a vertex with E. After solving the system, the velocity $\mathbf{u}^{n+1}_h$ is easily recovered by local postprocessing.

4.6. Pressure gradient upgrade

The last step is the pressure gradient update (2.13). Since $\mathbf{q}^{n+1}_h$ and the test function ξ_h are in the same discrete space $(W_h)^d$, this involves a simple update of the DOFs of $\mathbf{q}^{n+1}_h$. In particular, for any degree of freedom point $\mathbf{r}$,

$$\mathbf{q}^{n+1}_h(\mathbf{r}) = \mathbf{q}^n_h(\mathbf{r}) - \frac{\mathbf{u}^{n+1}_h(\mathbf{r}) - \tilde{\mathbf{u}}^{n+1}_h(\mathbf{r})}{\Delta t}.$$

Remark 4.2. The method presented above requires solving at each time step three symmetric positive definite systems, two for the components of the predicted velocity in the predictor step, and one for the pressure in the projection step. Due to the local elimination of the viscous stress and the corrected velocity, the systems have only three unknowns per element, which is a significant reduction from the original saddle point systems. The resulting algorithm is computationally very efficient and obtains exactly divergence free velocity.
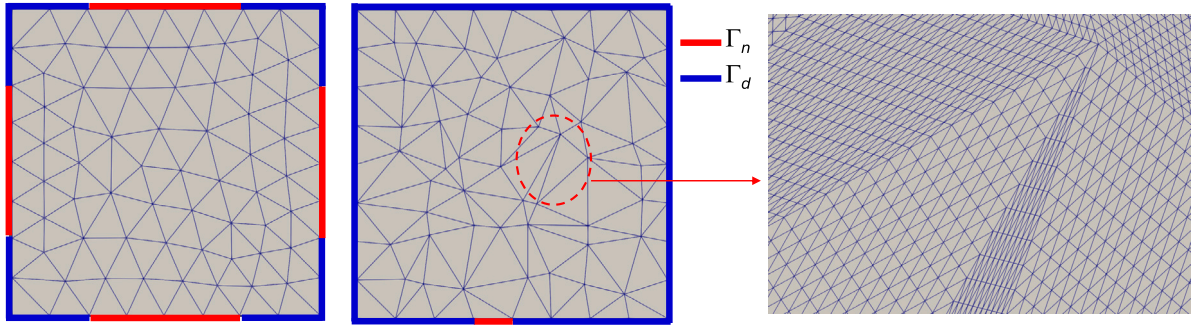
Fig. 4. Test 1. Coarse computational grids and BCs: MDG and BCs scenario 1 (left), BDG and BCs scenario 2 (center). Right: Zoom of the BDG at the 5th refinement level.

5. Numerical tests

We present four tests to illustrate the behavior of the presented method. In the first test we study the convergence order in both space and time, under different boundary conditions (BCs) settings, considering both mildly and strongly distorted computational grids. The second test deals with the lid-driven cavity problem for several scenarios of the assigned BCs, where we compare our results with a solution provided in the recent literature. In the third test, we simulate Stokes flows in a confined channel with a variable geometry and compare the results with reference literature solutions obtained by applying the lubrication theory. In the last test, we investigate the capabilities of the proposed method in a "real-world" application with a strongly irregular computational domain.

The computational grids of the presented tests have been created by the open-source software Netgen [41]. An in-house code has been used to implement the method, and the open-source software Paraview [42] has been used to visualize the results.

The problems presented in tests 2–4 are made dimensionless using the scales for length and velocity equal to the characteristic size of the domain and velocity, L_0 and U_0, respectively, and adopting the scales for kinematic pressure Ψ_0 and time T_0 as

$$\Psi_0 = \frac{\nu U_0}{L_0}, \qquad T_0 = \frac{L_0^2}{\nu}. \tag{5.1}$$

These scales are specified for each test in its corresponding section.

5.1. Test 1: convergence order in space and time

We consider the (dimensionless) square domain $[-0.5, 0.5] \times [-0.5, 0.5]$ and the analytical (dimensionless) solution

$$u_x = -\cos(x)\sin(y)\sin(2\pi t)^2, \tag{5.2a}$$

$$u_y = -\sin(x)\cos(y)\sin(2\pi t)^2, \tag{5.2b}$$

$$\Psi = \frac{\cos(2x) + \cos(2y)}{4}\sin(2\pi t)^2. \tag{5.2c}$$

The kinematic fluid viscosity is assumed to be 1. We consider the coarse mildly and badly distorted grids shown in Fig. 4, denoted by MDG and BDG, respectively. MDG has 85 vertices and 136 triangles, while BDG has 81 vertices and 128 triangles. For any triangle E of the grids, we consider the dimensionless quantity A_r,

$$A_r = \frac{h_{min}}{L_{max}} \frac{2}{\sqrt{3}},$$

where h_{min} and L_{max} are the minimum value of the heights and the maximum value of the lengths of E, respectively, $\frac{h_{min}}{L_{max}}$ is the aspect ratio of E, and $\frac{\sqrt{3}}{2}$ is the aspect ratio of the equilateral triangle. The number A_r, which is in the range $[0, 1]$, represents the deviation of the aspect ratio of E from the ideal value of an equilateral triangle. Dividing the interval $[0, 1]$ into 19 sub-intervals with equal length, Fig. 5 shows the number of triangles of MDG and BDG falling within each of the sub-intervals. As expected, for the MDG, the number of triangles is higher for the sub-ranges with higher values of A_r, compared to the BDG. The minimum values of A_r are 0.36 and 0.21 for MDG and BDG, respectively.

Starting from the coarse MDG and BDG, we operate five refinements by halving each edge. At each grid refinement, the number of triangles within each aspect ratio sub-interval in Fig. 5 is multiplied by four. Below, the coarsest MDG and BDG are denoted as "refinement level 0".

We consider two sets of BCs as depicted in Fig. 4. In the first case, there are alternating sections of Γ_d and Γ_n, while in the second case only one boundary edge is on Γ_n. Below, the two cases are denoted as "BCs scenario 1" and "BCs scenario 2", respectively. The
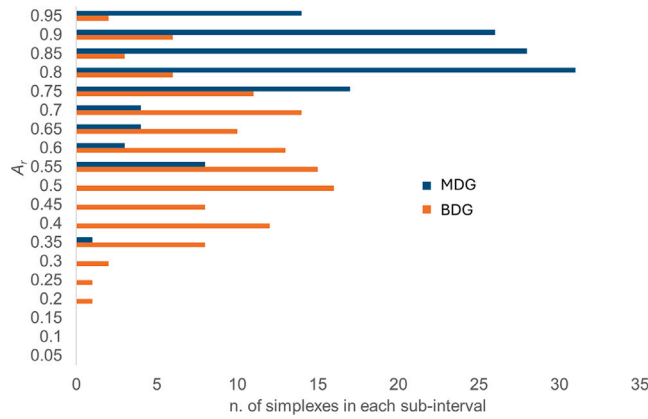
Fig. 5. Test 1. Number of triangles of the coarse MDG and BDG falling in each sub-range of the interval $[0, 1]$.

Table 1
Test 1. L_2-norms of errors and convergence orders in space: MDG, BCs scenario 1.

Ref. level l	L_2^{Ψ}	$L_2^{u_x}$	$L_2^{u_y}$	r_h^{Ψ}	$r_h^{u_x}$	$r_h^{u_y}$
0	1.142E−02	2.996E−04	2.972E−04			
1	2.865E−03	7.606E−05	7.582E−05	1.995E+00	1.978E+00	1.971E+00
2	7.187E−04	1.923E−05	1.909E−05	1.995E+00	1.984E+00	1.990E+00
3	1.796E−04	4.831E−06	4.802E−06	2.000E+00	1.993E+00	1.991E+00
4	4.505E−05	1.212E−06	1.204E−06	1.995E+00	1.995E+00	1.995E+00
5	1.148E−05	3.106E−07	3.087E−07	1.972E+00	1.964E+00	1.964E+00

second case is a way to implement full Dirichlet boundary condition, where the pressure is fixed at one edge to ensure uniqueness of the solution.

To analyze the convergence order in space, we choose a small (dimensionless) time step size, $\Delta t = 1 \cdot 10^{-4}$, and for each grid refinement level we compute the L_2-norms of the errors of the velocity components and kinematic pressure as

$$L_2^q = \max_{1 \leq n \leq N} \| q(t_n) - q_h^n \|, \quad q = u_x, u_y, \Psi. \tag{5.3}$$

Let h_l be the grid size of the lth refinement level. If we assume that the error of variable q associated with the lth grid refinement level is $err_l^q \sim h_l^{r_h^q}$, the associated rate of convergence r_h^q is obtained by comparing the errors of the two consecutive refinement level grids with sizes h_{l-1} and h_l:

$$r_h^q = \frac{\log\left(\frac{err_{l-1}^q}{err_l^q}\right)}{\log\left(\frac{h_{l-1}}{h_l}\right)}.$$

In Tables 1–4 we list the L_2 norms of the errors and the associated convergence order. The values of the errors obtained for the MDG are similar for the two scenarios of BCs. As expected, $L_2^{u_x}$ and $L_2^{u_y}$ are slightly smaller for BCs scenario 2, where all but one of the boundary edges are on Γ_d, while L_2^{Ψ} is slightly higher in BCs scenario 2 than in BCs scenario 1. The convergence order is 2 both all three variables, L_2^{Ψ} and $L_2^{u_x}$ and $L_2^{u_y}$, see Tables 1–2. The errors on the BDG, reported in Tables 3–4, are generally 2 orders of magnitude higher than on the MDG. The order of convergence for the pressure is 2 and it is just below 2 for the velocity components on the finest grid levels.

To study the convergence order in time, we use the finest grid (5-th refinement level) and a (dimensionless) time step size Δt in the range $\left[1 \cdot 10^{-1}, 1 \cdot 10^{-4}\right]$. Again, we compute the L_2-norms of the errors as in (5.3), and the convergence rates as

$$r_{\Delta t}^q = \frac{\log\left(\frac{err_{l-1}^q}{err_l^q}\right)}{\log\left(\frac{\Delta t_{l-1}}{\Delta t_l}\right)},$$

where Δt_{l-1} and Δt_l are two consecutive values of the time step size in the selected range. Tables 5–8 show the computed errors and convergence rates. The convergence rate is 1 for all variables, with slightly higher errors on the MDG. These results are consistent with the theoretical first order convergence for the time discretization error established in Theorem 3.2. In addition, the results from all tables illustrate the unconditional in time and space stability of the method established in Theorem 3.1, as well as its robustness for highly distorted grids.

Table 2

Test 1. L_2-norms of errors and convergence orders in space: MDG, BCs scenario 2.

Ref. level l	L_2^{ψ}	$L_2^{u_x}$	$L_2^{u_y}$	r_h^{ψ}	$r_h^{u_x}$	$r_h^{u_y}$
0	1.188E−02	2.402E−04	2.546E−04			
1	2.981E−03	6.137E−05	6.518E−05	1.995E+00	1.970E+00	1.966E+00
2	7.457E−04	1.547E−05	1.631E−05	1.999E+00	1.992E+00	1.998E+00
3	1.861E−04	3.867E−06	4.092E−06	1.999E+00	1.994E+00	1.994E+00
4	4.659E−05	9.662E−07	1.027E−06	2.000E+00	2.000E+00	1.993E+00
5	1.163E−05	2.476E−07	2.634E−07	1.999E+00	1.964E+00	1.963E+00

Table 3

Test 1. L_2-norms of errors and convergence orders in space: BDG, BCs scenario 1.

Ref. level l	L_2^{ψ}	$L_2^{u_x}$	$L_2^{u_y}$	r_h^{ψ}	$r_h^{u_x}$	$r_h^{u_y}$
0	1.758E+01	1.955E−01	1.939E−01			
1	4.423E+00	2.126E−02	2.138E−02	1.990E+00	3.201E+00	3.181E+00
2	1.107E+00	3.123E−03	3.122E−03	1.999E+00	2.767E+00	2.775E+00
3	2.770E−01	8.013E−04	8.017E−04	1.998E+00	1.963E+00	1.962E+00
4	6.544E−02	2.089E−04	2.098E−04	2.082E+00	1.939E+00	1.934E+00
5	1.640E−02	5.586E−05	5.617E−05	1.997E+00	1.903E+00	1.901E+00

Table 4

Test 1. L_2-norms of errors and convergence orders in space: BDG, BCs scenario 2.

Ref. level l	L_2^{ψ}	$L_2^{u_x}$	$L_2^{u_y}$	r_h^{ψ}	$r_h^{u_x}$	$r_h^{u_y}$
0	2.389E+01	2.021E−01	1.976E−01			
1	6.709E+00	2.657E−02	2.609E−02	1.832E+00	2.927E+00	2.921E+00
2	1.489E+00	3.604E−03	3.477E−03	2.171E+00	2.882E+00	2.908E+00
3	3.765E−01	9.022E−04	8.801E−04	1.984E+00	1.998E+00	1.982E+00
4	9.428E−02	2.350E−04	2.291E−04	1.998E+00	1.941E+00	1.941E+00
5	2.363E−02	6.251E−05	6.099E−05	1.996E+00	1.910E+00	1.910E+00

Table 5

Test 1. L_2-norms of errors and convergence order in time: MDG, BCs scenario 1.

Δt [−]	L_2^{ψ}	$L_2^{u_x}$	$L_2^{u_y}$	$r_{\Delta t}^{\psi}$	$r_{\Delta t}^{u_x}$	$r_{\Delta t}^{u_y}$
1.E-01	1.117E−02	3.081E−04	3.017E−04			
1.E-02	1.130E−03	3.088E−05	3.053E−05	9.95E−01	9.99E−01	9.95E−01
1.E-03	1.140E−04	3.095E−06	3.053E−06	9.96E−01	9.99E−01	1.00E+00
1.E-04	1.148E−05	3.106E−07	3.087E−07	9.97E−01	9.98E−01	9.95E−01

Table 6

Test 1. L_2-norms of errors and convergence orders in time: MDG, BCs scenario 2.

Δt [−]	L_2^{ψ}	$L_2^{u_x}$	$L_2^{u_y}$	$r_{\Delta t}^{\psi}$	$r_{\Delta t}^{u_x}$	$r_{\Delta t}^{u_y}$
1.E-01	1.156E−02	2.436E−04	2.603E−04			
1.E-02	1.158E−03	2.436E−05	2.609E−05	9.991E−01	1.000E+00	9.990E−01
1.E-03	1.162E−04	2.460E−06	2.603E−06	9.986E−01	9.956E−01	1.001E+00
1.E-04	1.165E−05	2.477E−07	2.634E−07	9.988E−01	9.971E−01	9.948E−01

Table 7

Test 1. L_2-norms of errors and convergence orders in time: BDG, BCs scenario 1.

Δt [−]	L_2^{ψ}	$L_2^{u_x}$	$L_2^{u_y}$	$r_{\Delta t}^{\psi}$	$r_{\Delta t}^{u_x}$	$r_{\Delta t}^{u_y}$
1.E-01	1.588E+01	5.444E−02	5.461E−02			
1.E-02	1.591E+00	5.456E−03	5.471E−03	9.991E−01	9.990E−01	9.992E−01
1.E-03	1.612E−01	5.524E−04	5.544E−04	9.943E−01	9.947E−01	9.942E−01
1.E-04	1.640E−02	5.586E−05	5.617E−05	9.926E−01	9.952E−01	9.943E−01

5.2. Test 2: lid-driven cavity

Stokes flow in closed rectangular cavities is a useful approximation for many applications, such as ceramics casting, polymer processing, roll coating, etc., see, e.g., [43,44]. We consider the classical lid-driven cavity test case in a square cavity $[0, L] \times [0, L]$,

Table 8

Test 1. L_2-norms of errors and convergence orders in time: BDG, BCs scenario 2.

Δt [−]	L_2^{Ψ}	$L_2^{u_x}$	$L_2^{u_y}$	$r_{\Delta t}^{\Psi}$	$r_{\Delta t}^{u_x}$	$r_{\Delta t}^{u_y}$
1.E−01	2.288E+01	6.106E−02	5.938E−02			
1.E−02	2.294E+00	6.119E−03	5.947E−03	9.989E−01	9.991E−01	9.993E−01
1.E−03	2.323E−01	6.187E−04	6.025E−04	9.944E−01	9.951E−01	9.943E−01
1.E−04	2.363E−02	6.251E−05	6.099E−05	9.926E−01	9.956E−01	9.947E−01
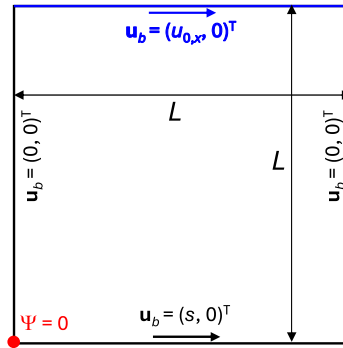
Fig. 6. Test 2. Sketch of the domain and assigned BCs.
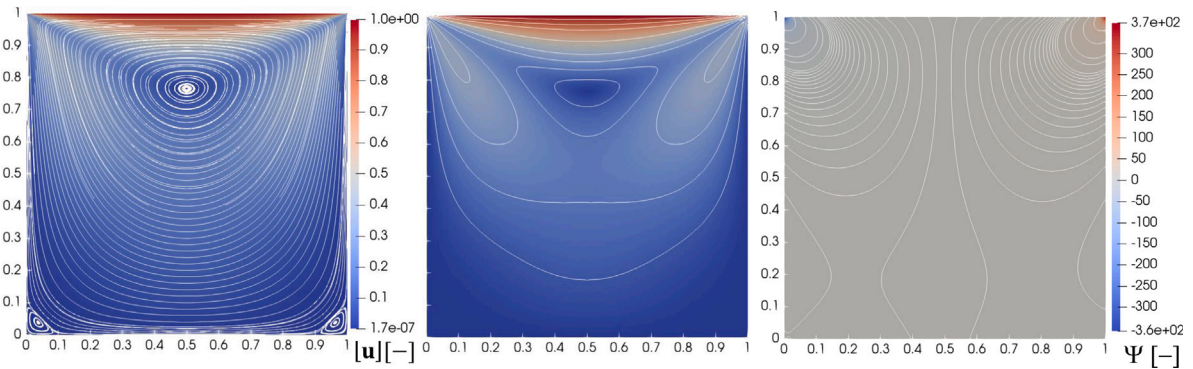
Fig. 7. Test 2. Computed solution on the 2nd refinement level of the BDG, case $s = 0$. Left: velocity field. Center: streamline contours. Right: pressure field.

as sketched in Fig. 6. The initial conditions (ICs) are zero velocity and pressure, while the BCs are assigned as in Fig. 6, where the upper wall moves along the x−direction with assigned velocity $u_{0,x}$ and s is the horizontal velocity component assigned at the bottom wall. We consider the cases $s = u_{0,x}$, $s = 0$, and $s = -u_{0,x}$. To make the results dimensionless, the scales for length and velocity are set $L_0 = L$ and $U_0 = u_{0,x}$, and we assume $\nu = 1$ m^2/s. The computational grids are the same mildly and badly distorted ones (MDG and BDG) used in Test 1, presented in Section 5.1 and the (dimensionless) time step size is $1 \cdot 10^{-1}$. A pressure BC $\Psi = 0$ is assumed at the bottom-left corner, to fix the pressure values.

In Fig. 7 we plot the solution computed on the 2nd refinement grid level of the BDG assuming $s = 0$ (fixed bottom wall). Very similar results have been obtained with the MDG and for brevity are not shown. The results are qualitatively in very good agreement with the literature, see [45, Figure 10 (left)] and [43, Figure 3.8 a,b]. In [45], the authors use a multiscale hybrid-mixed method, while in [43], the author presents a regularized solution of the Stokes equations, where a cut-off blob function replaces the classical delta-Dirac one. In Fig. 8 we plot the computed profiles with both MDG and BDG of the velocity components and pressure along the vertical and horizontal mid-lines of the domain, in the case of $s = 0$. The grid distortion does not seem to affect the results. The u_x profiles along the vertical mid-line match very well the solution provided in [43].

The velocity streamlines, computed over the 2nd refinement of the BDG, are in very good agreement with the solution provided in [44], for the cases of $s = u_{0,x}$ and $s = -u_{0,x}$, as shown in Fig. 9. In [44], the authors solve the steady part of the Stokes flows problem using boundary element methods, while the time derivative is approximated by the finite difference method. The streamlines on the MDG and finer grids are very similar, and for brevity are not shown. Both velocity components computed along the vertical and horizontal mid-lines in the cases of $s = u_{0,x}$ and $s = -u_{0,x}$ fit very well the results provided in [44], see Figs. 10 and 11.

Overall, we conclude that the results from the presented method for this benchmark problem match very well literature solutions. Moreover the results are robust with respect to grid distortion.
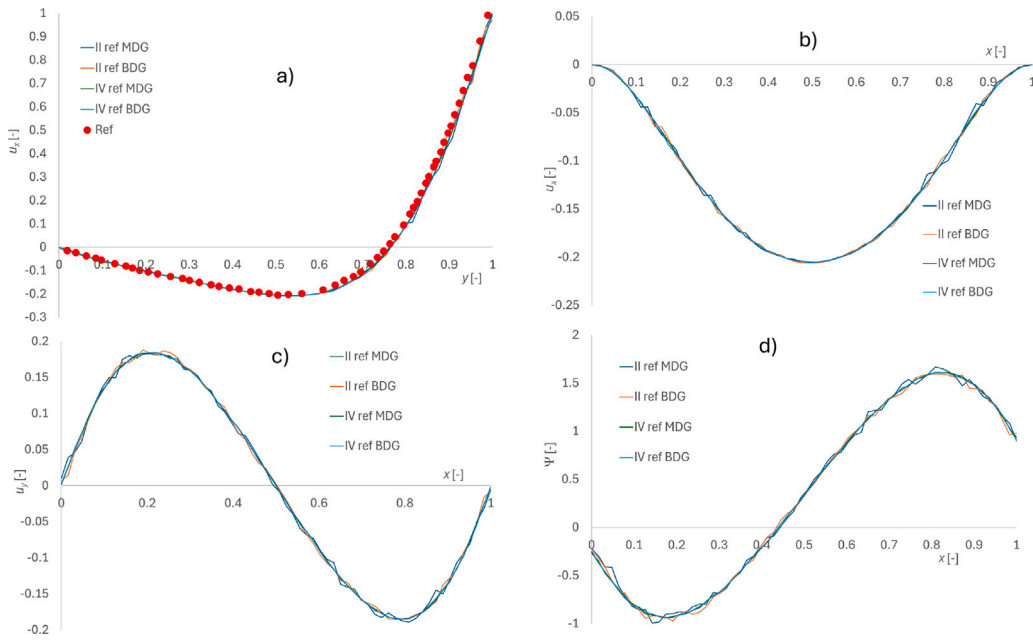
Fig. 8. Test 2. Computed solution on the 2nd and 4th refinements of the MDG and BDG case $s = 0$: (a) profile of u_x along the vertical mid-line, (b) profile of u_x along the horizontal mid-line, (c) profile of u_y along the horizontal mid-line, (d) profile of Ψ along the horizontal mid-line. "Ref." marks the solution provided in [43].
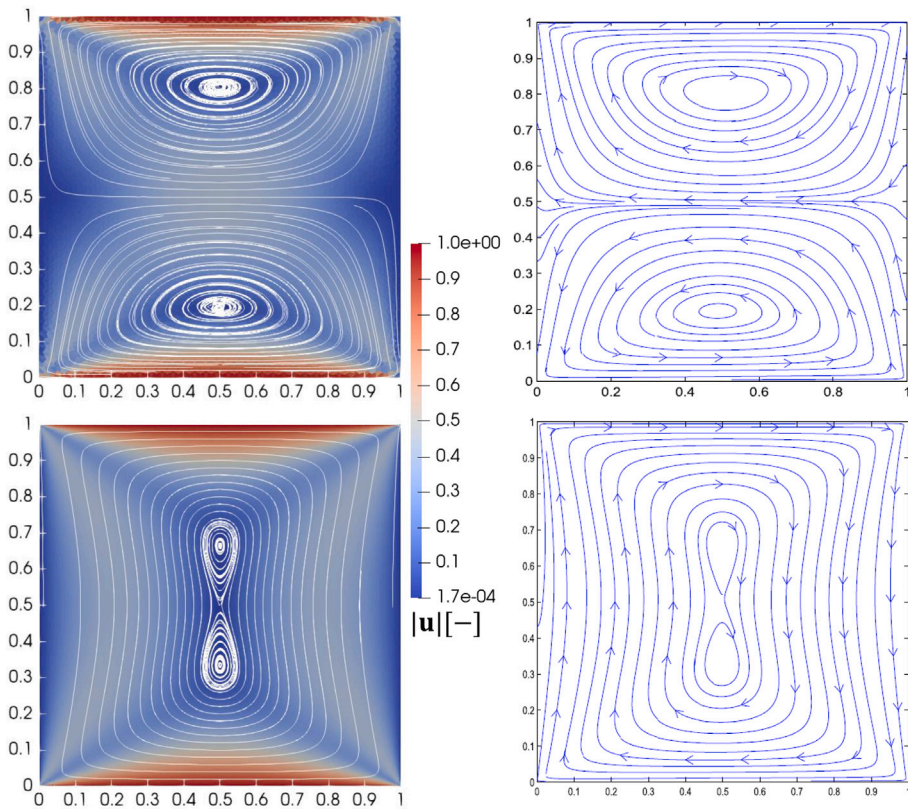

Fig. 9. Test 2. Computed velocity streamlines on the 2nd refinement of BDG. Top row: case $s = u_{0,x}$; left: our solution, right: literature results in [44]. Bottom row: case $s = -u_{0,x}$; left: our solution, right: literature results in [44].
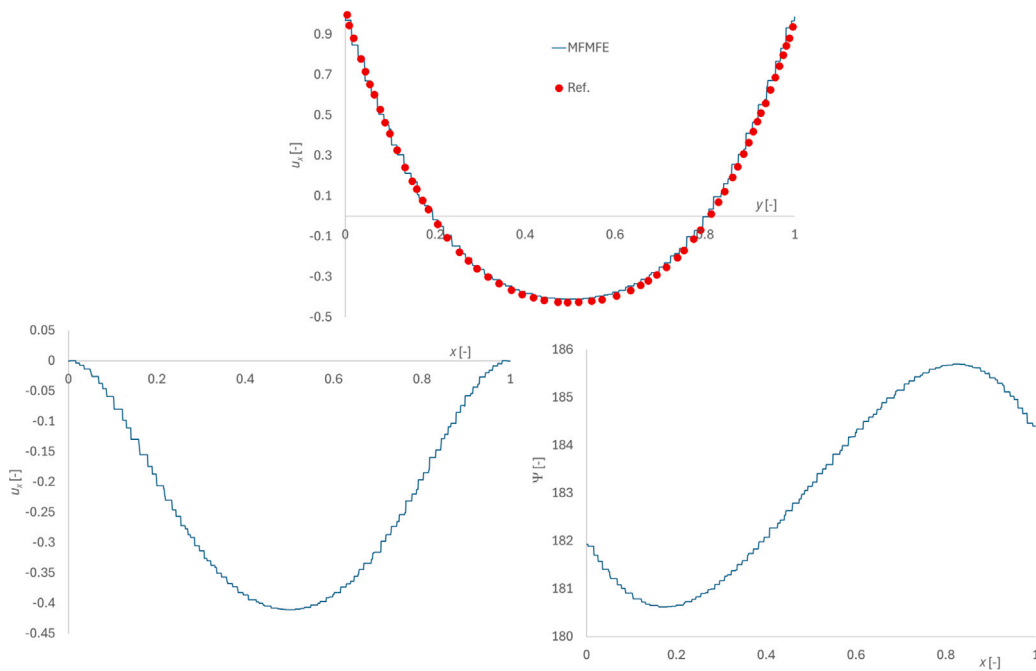
Fig. 10. Test 2. Computed profiles of the velocity components and pressure on the 2nd refinement of BDG, case $s = u_{0,x}$. Top: u_x along the vertical mid-line with "Ref." showing the results in [44]. Bottom row: data along the horizontal mid-line; left: u_x, right: Ψ.
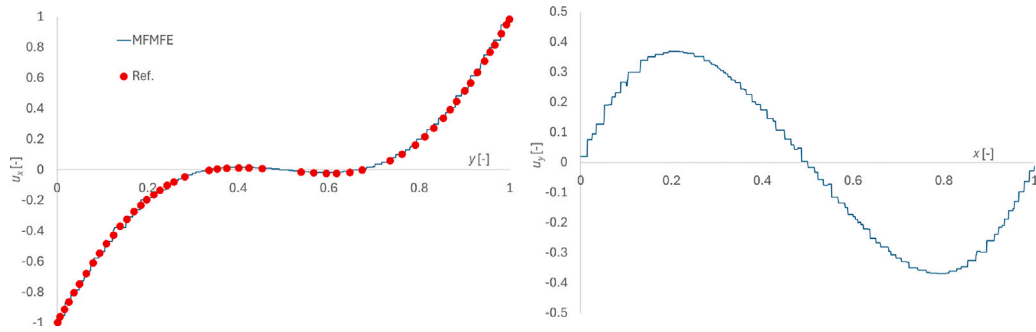
Fig. 11. Test 2. Computed profiles of the velocity components and pressure over the 2nd refinement BDG, case $s = -u_{0,x}$. Left: u_x along the vertical mid-line with "Ref." showing the results in [44]. Right: u_y along the horizontal mid-line.

5.3. Test 3: Stokes flow in a channel with smoothly changing width

Stokes flows in confined channel with variable geometries is encountered in many industrial, biomedical, and biological systems, such as microfluidics, flow driven by contracting channel walls, lab-chip technologies, drug carrying micro-devices, etc., see [46]. We consider the channel in Fig. 12, with characteristic length and depth L_0 and h_0, respectively, and set $\delta = \frac{h_0}{L_0} = 1$. The ICs are zero velocity and pressure, and the BCs are assigned as in Fig. 12, with parabolic Poiseuille inflow velocity profile, no-slip BCs on the upper and lower channel walls, and zero pressure at the downstream channel end. To make the results dimensionless, the length scale is set equal to $L_0 = h_0$ and the velocity scale is set equal to the mean inflow velocity, $U_0 = \bar{U}$. Following the experiments in [46], we set $L_0 = h_0 = 0.005$ m, $\bar{U} = 5.76 \cdot 10^{-4}$ m/s and $\nu = 2.89 \cdot 10^{-3}$ m^2/s, resulting in Reynolds number $Re = \frac{\bar{U} L_0}{\nu} \simeq 1 \cdot 10^{-3}$. We also introduce the following (dimensionless) variables,

$$X = \frac{x}{L_0}, \quad Y = \frac{y}{L_0}, \quad H = \frac{h}{L_0},$$

where h is the distance of the top channel wall from the bottom (see Fig. 12). The equation for the (dimensionless) restriction of the channel width between $X = -1$ and $X = 1$ is [46,47]

$$H(X) = 1 - \frac{\lambda}{2} \left(1 + \cos(\pi X) \right), \quad 0 \leq \lambda < 1, \tag{5.4}$$

Table 9

Test 3. Number of triangles and vertices for the coarse and fine grids.

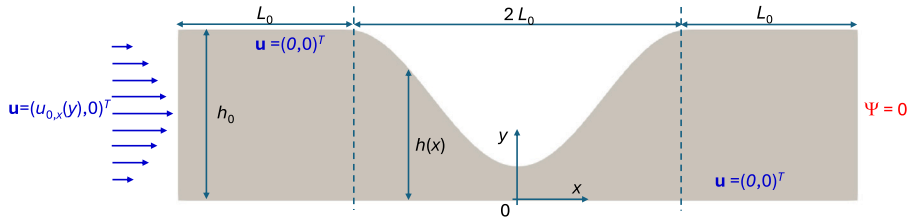| λ	Coarse grids		Fine grids	
	N_T	N	N_T	N
0.2	17 306	9149	64 079	32 705
0.5	16 257	8624	61 429	31 382
0.8	15 648	8320	57 090	29 212

Fig. 12. Test 3. Channel geometry and assigned BCs.
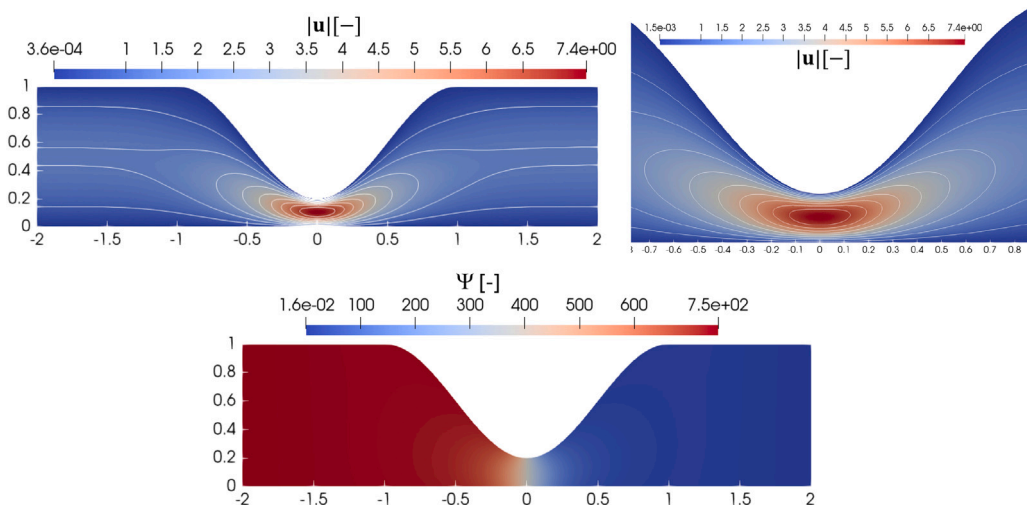
Fig. 13. Test 3. Computed (dimensionless) solution for $\lambda = 0.8$. Top row: velocity field and velocity magnitude contours (left) and zoom in the restriction area (right). Bottom row: computed pressure field.

where λ is the maximum (dimensionless) width contraction. The (dimensionless) inflow Poiseuille velocity profile is

$$\mathbf{u}(Y) = \left(6\left(Y - Y^2 \right), 0 \right)^T . \tag{5.5}$$

In the presented simulations we set $\lambda = 0.2,\ 0.5,\ 0.8$. We run our simulations over a set of coarse and fine grids, whose number of simplices N_T and vertices N are listed in Table 9. The dimensionless time step size is $1 \cdot 10^{-2}$. We also ran other simulations using smaller and larger time step sizes, but the results did not change significantly.

In Fig. 13 we show the (dimensionless) computed velocity and pressure fields with $\lambda = 0.8$. In [46], the same problem is solved by the lubrication theory, considering the higher-order terms coming from the perturbation theory, accounting for values of $\delta \simeq 1$, see also [47]. The authors of [46] provide an analytical expression of the (dimensionless) pressure drop $\Delta\Psi$ from $X = -1$ to $X = 1$ as function of the restriction parameter λ, shown in Fig. 14, and confirmed by the experimental results in [46]. The scatter between the analytical data given by the lubrication theory and the experiments can be justified by the 3D flow effects not accounted in the 2D lubrication theory. These 3D effects generally tend to disappear as the value of λ increases and the channel becomes narrower. In the same Fig. 14 we also plot the dimensionless pressure drops computed by the present method on the coarse and fine grids. Our results fit very well the reference data for the three investigated values of λ, and the mesh size does not affect significantly the pressure drop.
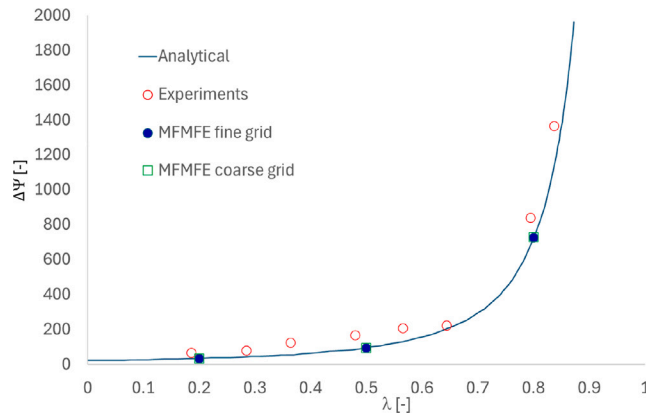
Fig. 14. Test 3. Computed (dimensionless) pressure drop from $X = -1$ to $X = 1$ for different values of λ. Nomenclature. "Analytical": data from the lubrication theory including the higher order terms [46]. "Experiments": experimental data provided in [46]. "MFMFE fine grids" and "MFMFE coarse grids": results from the present method.

Table 10
Test 4. Values of the significant geometrical, kinematic and rheological variables.

R [m]	0.0004
ϕ_0 [rad]	$\pi/12$
L_a [m]	$R\,\pi$
L_B [m]	$2\,R\,\tan\phi_0$
H [m]	$\frac{R}{\sin\phi_0}$
R_i [m]	$\frac{R}{2.5\ D^2\ \sin\phi_0}$
a [–]	0.7
n [–]	11
U_{max} [m/s]	0.0005
ν [m²/s]	$6.7 \cdot 10^{-6}$

5.4. Test 4: pulsatile Stokes flow within a highly irregular domain

The purpose of the test in this section is to investigate the capability of the proposed method to simulate pulsatile Stokes flows within very irregular domains, which find several industrial and biomedical applications, e.g., hydrodynamics in micro and mini coiling systems, vascular hemodynamics in small vessels, and peristaltic flows. We consider flow in a duct with a central elliptic expansion and a wavy-shaped inner solid obstacle with mean radius R_i, see the geometry in Fig. 15, where the equation of the elliptic-shape expansion of the duct is also shown. We assign a pulsatile inflow Poiseuille velocity profile at the upstream end, pressure condition $\Psi = 0$ at the downstream end, and zero velocity on the remaining boundaries of the external duct and on the surface of the inner obstacle. We make the problem dimensionless considering the scales for lengths $L_0 = D$ and velocity $U_0 = U_{max}$, where U_{max} is the maximum value of the Poiseuille assigned inflow velocity profile and $D = 2\,R$ is the diameter of the duct at the upstream end. The significant geometrical, kinematic and rheological variables are listed in Table 10. The Reynolds number is $Re = \frac{U_{max}D}{\nu} \simeq 0.059$. The wavy profile of the inner obstacle is similar to the one used in test 4 in [14],

$$x = r_i \sin\phi, \quad y = r_i \cos\phi, \quad r_i = R_i + a \cos n\phi, \quad 0 \le \phi \le 2\pi, \tag{5.6}$$

with a and n listed in Table 10. The inflow (dimensionless) pulsatile Poiseuille velocity is

$$u_x(r,t) = \hat{U}_{max}(t)\,\frac{(R^2 - d_r^2)}{R^2}, \quad \hat{U}_{max}(t) = \frac{U_{max}}{U_0}\cos\left(\frac{2\pi}{t_0}t\right), \quad u_y = 0, \quad \forall t \ge 0, \tag{5.7}$$

where d_r is the distance of any point r of the inflow section from its center, t_0 is the period of the pulsation, and the other symbols are specified in Fig. 15.

We set $t_0 = 1$ [s] and the associated Womersley number [48] $\alpha = R\sqrt{\frac{1}{t_0\nu}} \simeq 0.155$. The computational grid has 59 949 triangles and 31 831 vertices, and in Fig. 16 we plot a zoom of the grid discretizing the inner portions of the obstacle. The (dimensionless) grid size ranges between 0.05 and 0.0015, and the (dimensionless) time step size is set to $\frac{t_0}{T_0 \cdot 160}$, with T_0 the time scale defined in Eq. (5.1).

In Fig. 17 we plot the radial profiles of the (dimensionless) horizontal velocity component close to the inflow section every quarter of period. The vertical velocity component is approximately 3–4 magnitude orders smaller than the absolute value of u_x. Fig. 18 shows the velocity and pressure fields every quarter of period, and Fig. 19 shows the zooms of the velocity field close to the inner invested obstacle at the same simulation times. Complex vortical structures, composed of four or more vortices with alternating flow direction, can be observed between the waves of obstacle perimeter.
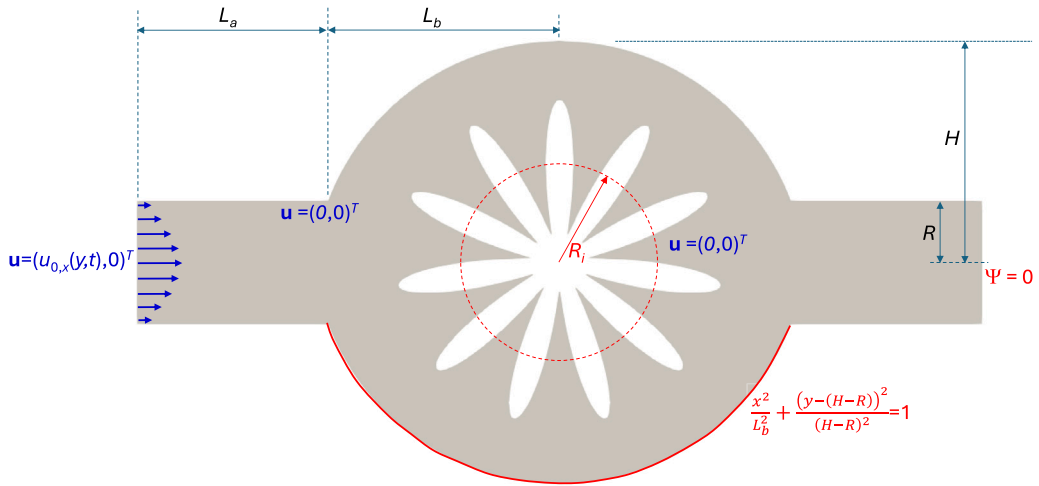
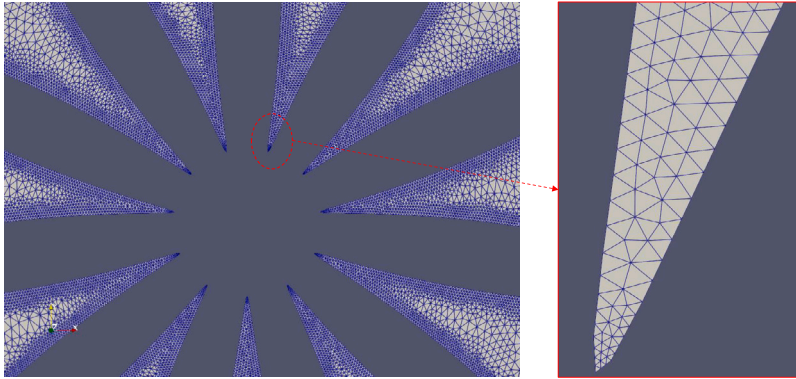**Fig. 15.** Test 4. Channel geometry and assigned BCs. Values of the variables are listed in Table 10.

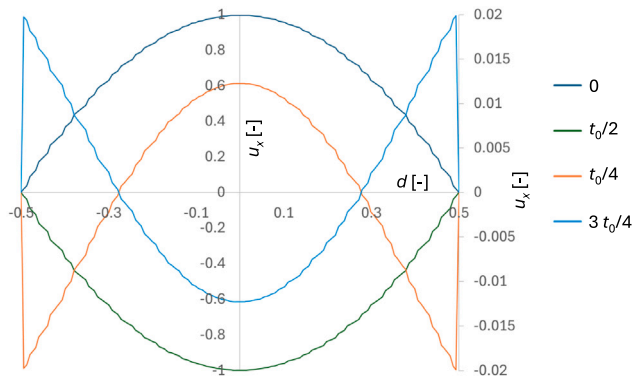**Fig. 16.** Test 4. Zoom of the computational grid.

Fig. 17. Test 4. Radial profiles of the (dimensionless) u_x close to the inflow section. Nomenclature: d is the (dimensionless) radial distance from the mid-line of the duct section. Values of u_x for $t = 0$ and $t = t_0/2$ on the primary axes. Values of u_x for $t = t_0/4$ and $t = 3t_0/4$ on the secondary axes.

We compute the forces of the fluid on the perimeter of the obstacle. The total force $\mathbf{F}$ is the sum of the kinematic pressure force, $\mathbf{F}_\Psi$, and the viscous force, $\mathbf{F}_v$. We divide the perimeter in N_p parts (edges) and set

$$\mathbf{F} = \mathbf{F}_\Psi + \mathbf{F}_v = \sum_{i=1}^{N_p} \left(\mathbf{F}_{\Psi,i} + \mathbf{F}_{v,i} \right). \tag{5.8}$$
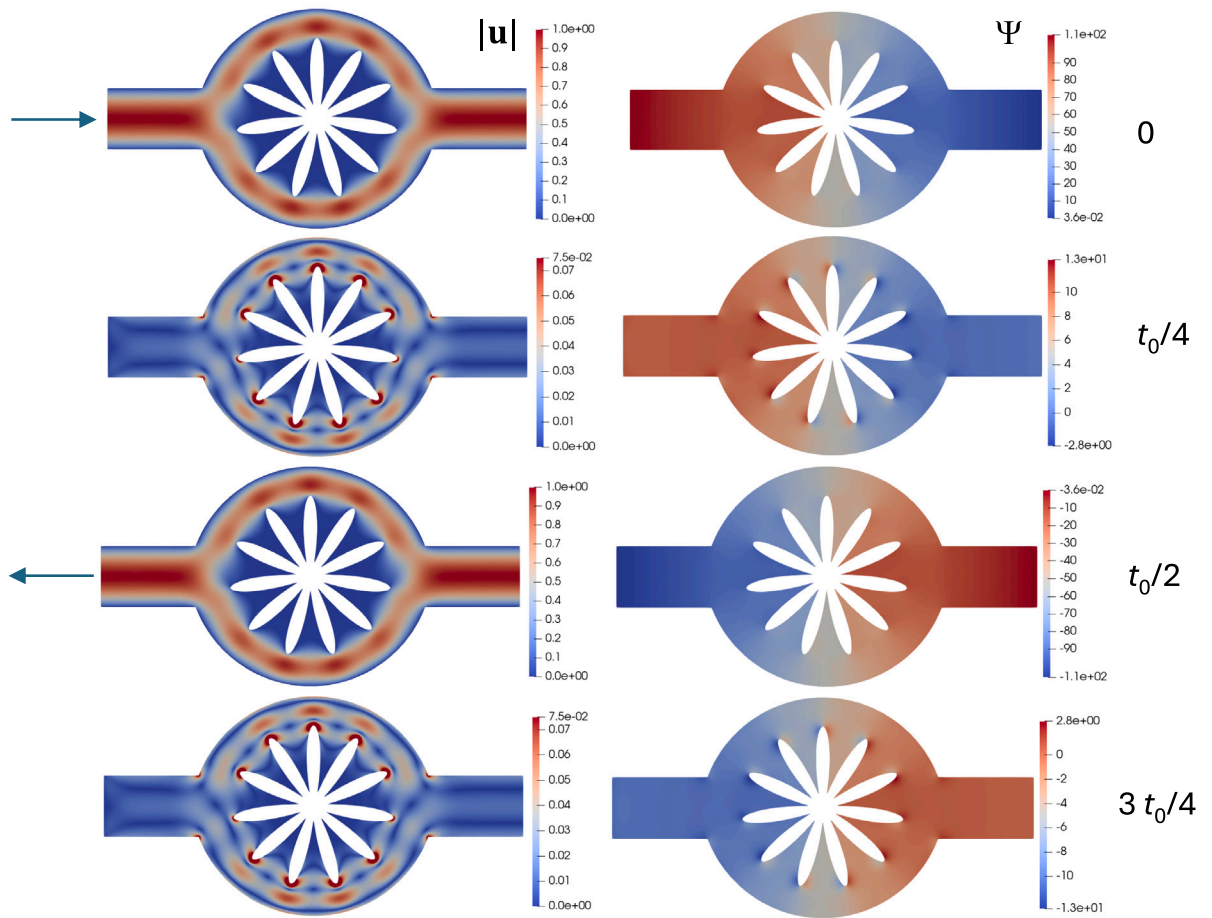
Fig. 18. Test 4. Dimensionless velocity and pressure fields every quarter-period. The arrow shows the mean flow direction.
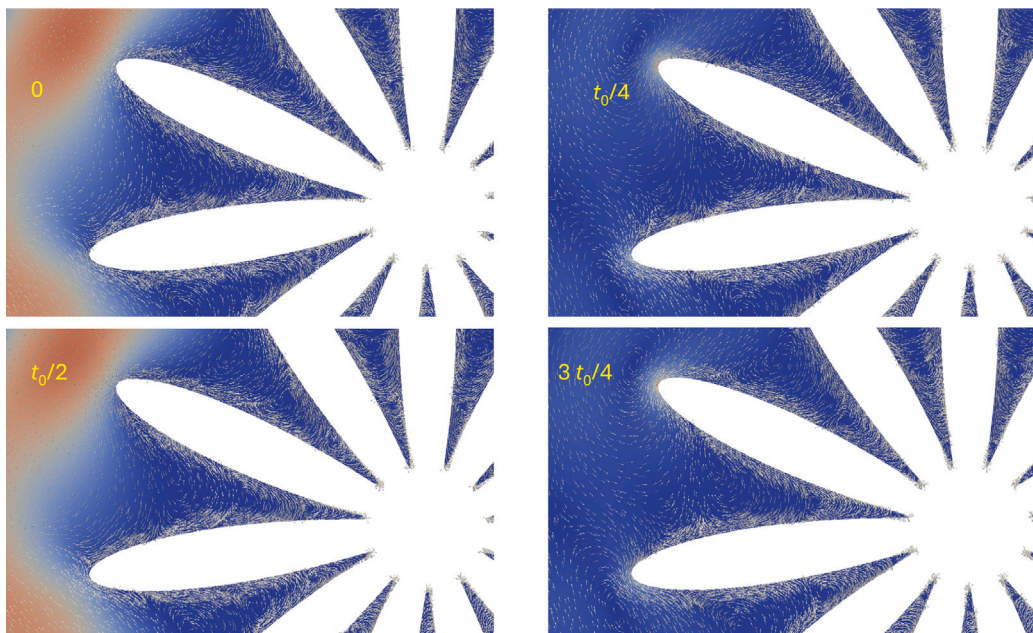

Fig. 19. Test 4. Zoom of the velocity vectors close to the invested obstacle every quarter-period.
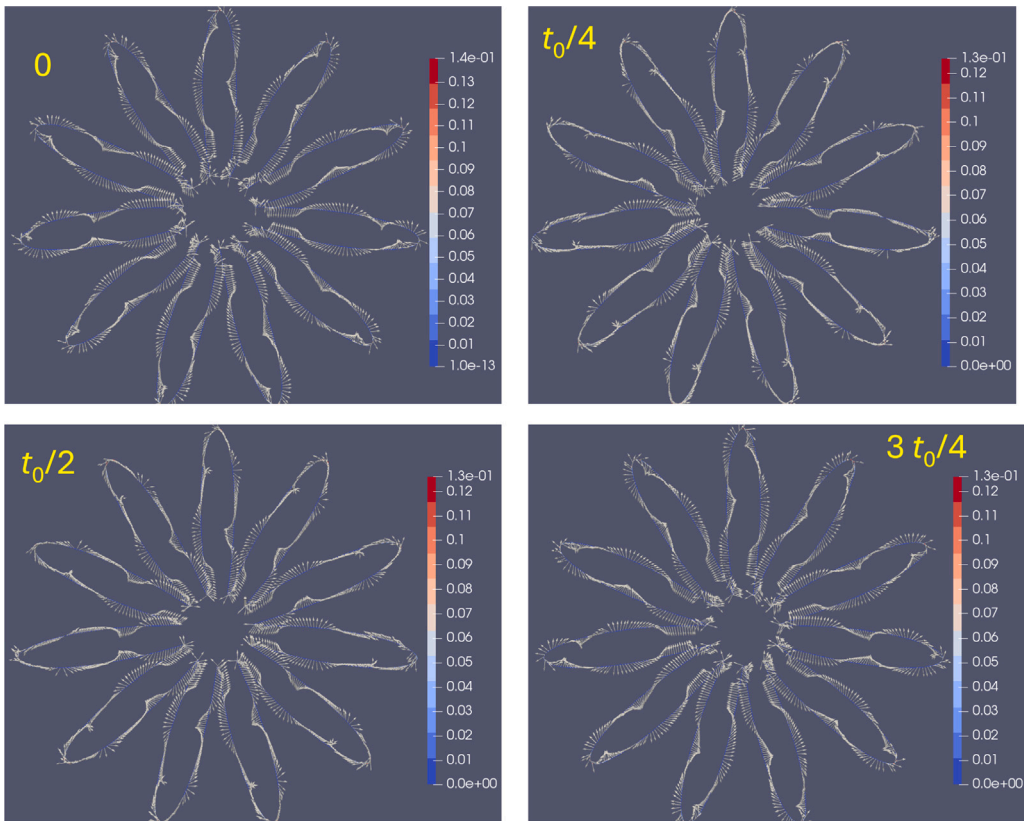
Fig. 20. Test 4. Viscous forces acting on the perimeter of the obstacle every quarter-period.

The force $\mathbf{F}_\Psi$ on the ith edge of the obstacle perimeter is computed as

$$\mathbf{F}_\Psi = |e_i| \frac{\Psi_1 + \Psi_2}{2} \mathbf{n}_i, \tag{5.9}$$

where $|e_i|$ is the length of the ith edge, Ψ_1 and Ψ_2 are the kinematic pressure values computed at the two vertices of the edge e_i, and $\mathbf{n}_i$ is the unit normal vector to e_i, pointing outward from the fluid region. The two values Ψ_1 and Ψ_2 are computed according to the P_1 approximation of Ψ within the triangle E sharing edge e_i with the perimeter of the solid obstacle. The x- and y-components of the viscous force acting on e_i are computed as

$$F_{x,v,i} = \nu \nabla u_x \cdot \mathbf{n}_i |e_i|, \qquad F_{y,v,i} = \nu \nabla u_y \cdot \mathbf{n}_i |e_i|, \tag{5.10}$$

where ∇u_x and ∇u_y are the gradient vectors of the velocity components computed within the triangle E sharing edge e_i with the obstacle. These gradient vectors are computed according to the linear variation of the velocity components within E.

Figs. 20 and 21 show the vectors $\mathbf{F}_v$ and $\mathbf{F}$, respectively, computed every quarter-period. The values of $\mathbf{F}_v$ and $\mathbf{F}$ in Figs. 20 and 21 are multiplied by the non-dimensionalizing coefficient $\frac{1}{\nu U_{max}}$. Observe the alternating direction of the vectors at time 0 and $t_0/2$, as well as at times $t_0/4$ and $3 t_0/4$, respectively, due to the periodicity of the process. The changes of the direction of $\mathbf{F}_v$ on the boundary of the obstacle at any investigated time are due to the vortical structures within the waves of the perimeter, as discussed before.

Overall the presented results illustrate the ability of our method to simulate flows in highly complex geometries in a numerically stable and accurate manner.

6. Conclusions

We have developed $H(\mathrm{div})$-conforming mixed finite element methods for the unsteady incompressible Stokes equations. We applied a projection method in the framework of the incremental pressure correction methodology, where a predictor and a projection problems are sequentially solved, accounting for the viscous effects and incompressibility, respectively. The predictor problem is based on a stress–velocity mixed formulation, while the projection problem uses a velocity–pressure mixed formulation. We established unconditional stability and first order in time accuracy. We then developed a specific method of the family on generally unstructured triangular grids, using the multipoint flux mixed finite element methodology. We used the RT_1 mixed finite
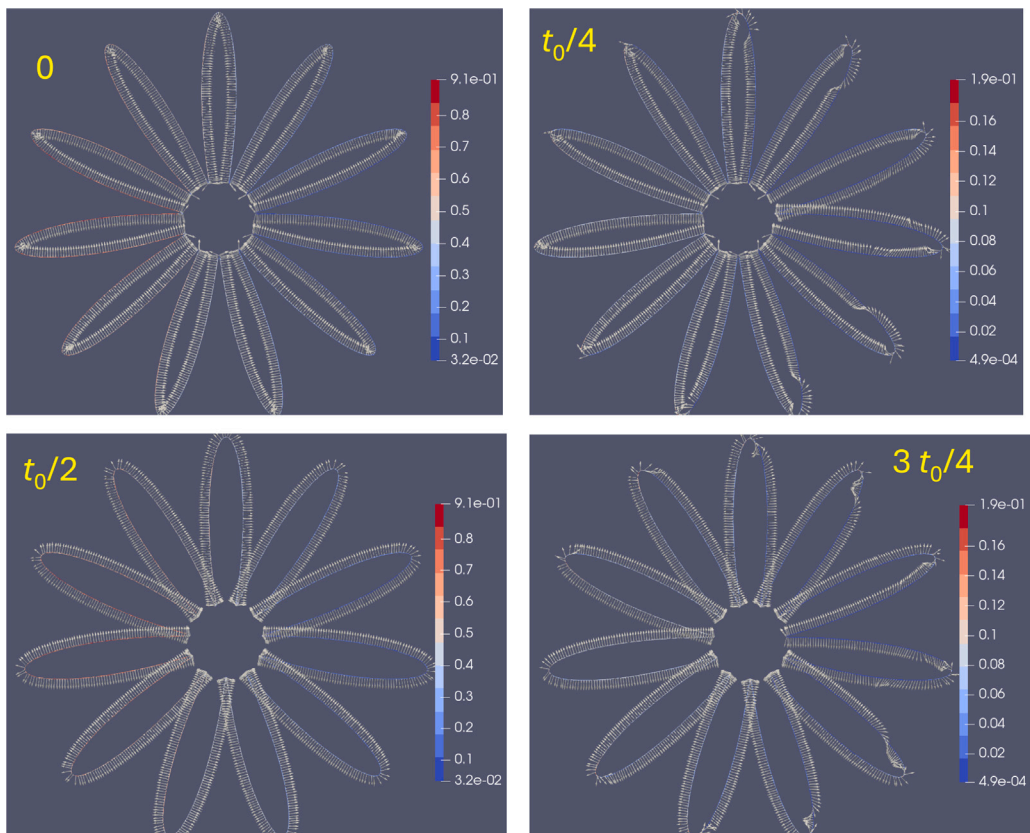
Fig. 21. Test 4. Total forces acting on the perimeter of the obstacle every quarter of period.

element pair and applied an inexact numerical integration to obtain mass lumping and local stress or flux elimination in both the prediction and projection problems. The resulting algebraic systems, which are sparse symmetric and positive definite with only three unknowns per element, are solved efficiently by the preconditioned conjugate gradient method. The scheme results in pointwise divergence-free velocity computed at the end of each time step. The computed velocity and pressure are second order accurate in space. The presented numerical experiments illustrate the accuracy and efficiency of the method for several benchmark problems and a challenging problem with highly complex geometry.

The proposed methodology could be straightforwardly extended to 3D problems. In this sense, the good performance obtained over the badly distorted grids, see Section 5.1, is encouraging for further applications to 3D tetrahedral grids, where an optimal aspect ratio is difficult to achieve. Furthermore, as mentioned in the introduction, the proposed methodology could be regarded as the seed of further extensions, e.g., coupled flow and transport problems and fluid–structure interaction. In particular, due to the good performance and robustness shown by the multipoint flux mixed finite element method proposed in [27] for Darcy flow with heterogeneous and discontinuous full tensor permeability, a natural extension of the presented algorithm would be to include the Brinkman term in the momentum equations to simulate the interaction between a free fluid and a porous medium, see [15].

CRediT authorship contribution statement

Costanza Aricò: Writing – original draft, Writing – review & editing, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Rainer Helmig:** Supervision, Resources, Project administration, Funding acquisition, Formal analysis. **Ivan Yotov:** Writing – original draft, Writing – review & editing, Visualization, Validation, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Data availability

Data will be made available on request.

References

[1] M. Deville, An Introduction to the Mechanics of Incompressible Fluids, Springer International Publishing, 2022.

[2] J.J. Feng, D. Joseph, The unsteady motion of solid bodies in creeping flows, J. Fluid Mech. 303 (1995) 83–102.

[3] D. Wei, P.G. Dehnavi, M.-E. Aubin-Tam, D. Tam, Measurements of the unsteady flow field around beating cilia, J. Fluid Mech. 915 (2021) A70.

[4] H. Mestre, J. Tithof, T. Du, W. Song, W. Peng, A.M. Sweeney, G. Olveda, J.H. Thomas, M. Nedergaard, D.H. Kelley, Flow of cerebrospinal fluid is driven by arterial pulsations and is reduced in hypertension, Nat. Commun. 9 (1) (2018) 4878.

[5] J.H. Thomas, Fluid dynamics of cerebrospinal fluid flow in perivascular spaces, J. R. Soc. Interface 16 (2019).

[6] C. Daversin-Catty, I.G. Gjerde, M.E. Rognes, Geometrically reduced modelling of pulsatile flow in perivascular networks, Front. Phys. 10 (2022).

[7] D.H. Kelley, J.H. Thomas, Cerebrospinal fluid flow, Annu. Rev. Fluid Mech. 55 (Volume 55, 2023) (2023) 237–264.

[8] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comp. 22 (104) (1968) 745–762.

[9] R. Temam, Sur l'approximation de la solution des équations de Navier–Stokes par la méthode des pas fractionnaires (II), Arch. Ration. Mech. Anal. 33 (1969) 377–385.

[10] D.L. Brown, R. Cortez, M.L. Minion, Accurate projection methods for the incompressible Navier–Stokes equations, J. Comput. Phys. 168 (2) (2001) 464–499.

[11] J. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Engrg. 195 (44) (2006) 6011–6045.

[12] D. Boffi, F. Brezzi, M. Fortin, Mixed Finite Element Methods and Applications, vol. 44, Springer, 2013.

[13] C. Aricò, M. Sinagra, C. Picone, T. Tucciarelli, MAST-RT0 solution of the incompressible Navier–Stokes equations in 3D complex domains, Eng. Appl. Comput. Fluid Mech. 15 (1) (2021) 53–93.

[14] C. Aricò, M. Sinagra, Z. Driss, T. Tucciarelli, A new solver for incompressible non-isothermal flows in natural and mixed convection over unstructured grids, Appl. Math. Model. 103 (2022) 445–474.

[15] C. Aricò, R. Helmig, D. Puleo, M. Schneider, A new numerical mesoscopic scale one-domain approach solver for free fluid/porous medium interaction, Comput. Methods Appl. Mech. Engrg. 419 (2024) Paper No. 116655, 28.

[16] B. Cockburn, G. Kanschat, D. Schötzau, A locally conservative LDG method for the incompressible Navier–Stokes equations, Math. Comp. 74 (2004) 1067–1095.

[17] J. Wang, X. Ye, New finite element methods in computational fluid dynamics by $H(\mathrm{div})$ elements, SIAM J. Numer. Anal. 45 (3) (2007) 1269–1286.

[18] J. Guzmán, M. Neilan, Conforming and divergence-free Stokes elements on general triangular meshes, Math. Comp. 83 (2013) 15–36.

[19] J. Volker, A. Linke, C. Merdon, M. Neilan, L.G. Rebholz, On the divergence constraint in mixed finite element methods for incompressible flows, SIAM Rev. 59 (3) (2017) 492–544.

[20] Y. Li, L.T. Zikatanov, New stabilized $P_1 \times P_0$ finite element methods for nearly inviscid and incompressible flows, Comput. Methods Appl. Mech. Engrg. 393 (2022) Paper No. 114815, 17.

[21] X. Hu, S. Lee, L. Mu, S.-Y. Yi, Pressure-robust enriched Galerkin methods for the Stokes equations, J. Comput. Appl. Math. 436 (2024) Paper No. 115449, 19.

[22] Z. Cai, C. Wang, S. Zhang, Mixed finite element methods for incompressible flow: Stationary Navier–Stokes equations, SIAM J. Numer. Anal. 48 (1) (2010) 79–94.

[23] G.N. Gatica, R. Oyarzúa, F.J. Sayas, Analysis of fully-mixed finite element methods for the Stokes–Darcy coupled problem, Math. Comp. 80 (276) (2011) 1911–1948.

[24] G.N. Gatica, A. Márquez, R. Oyarzúa, R. Rebolledo, Analysis of an augmented fully-mixed approach for the coupling of quasi-Newtonian fluids and porous media, Comput. Methods Appl. Mech. Engrg. 270 (2014) 76–112.

[25] S. Caucao, G.N. Gatica, R. Oyarzúa, I. Šebestová, A fully-mixed finite element method for the Navier–Stokes/Darcy coupled problem with nonlinear viscosity, J. Numer. Math. 25 (2) (2017) 55–88.

[26] S. Caucao, T. Li, I. Yotov, A multipoint stress-flux mixed finite element method for the Stokes–Biot model, Numer. Math. 152 (2) (2022) 411–473.

[27] M.F. Wheeler, I. Yotov, A multipoint flux mixed finite element method, SIAM J. Numer. Anal. 44 (5) (2006) 2082–2106.

[28] M.G. Edwards, C.F. Rogers, Finite volume discretization with imposed flux continuity for the general tensor pressure equation, Comput. Geosci. 2 (4) (1998) 259–290 (1999).

[29] I. Aavatsmark, An introduction to multipoint flux approximations for quadrilateral grids, Comput. Geosci. 6 (3–4) (2002) 405–432.

[30] F. Brezzi, M. Fortin, L.D. Marini, Error analysis of piecewise constant pressure approximations of Darcy's law, Comput. Methods Appl. Mech. Engrg. 195 (2006) 1547–1559.

[31] R.A. Klausen, R. Winther, Robust convergence of multi point flux approximation on rough grids, Numer. Math. 104 (3) (2006) 317–337.

[32] R. Ingram, M.F. Wheeler, I. Yotov, A multipoint flux mixed finite element method on hexahedra, SIAM J. Math. Anal. 48 (4) (2010) 1281–1312.

[33] M.F. Wheeler, G. Xue, I. Yotov, A multipoint flux mixed finite element method on distorted quadrilaterals and hexahedra, Numer. Math. 121 (1) (2012) 165–204.

[34] I. Ambartsumyan, E. Khattatov, J.J. Lee, I. Yotov, Higher order multipoint flux mixed finite element methods on quadrilaterals and hexahedra, Math. Models Methods Appl. Sci. 29 (06) (2019) 1037–1077.

[35] H. Egger, B. Radu, On a second-order multipoint flux mixed finite element methods on hybrid meshes, SIAM J. Numer. Anal. 58 (3) (2020) 1822–1844.

[36] W.M. Boon, A. Fumagalli, A multipoint vorticity mixed finite element method for incompressible Stokes flow, Appl. Math. Lett. 137 (2023) Paper No. 108498, 8.

[37] B. Radu, Finite Element Mass Lumping for H(div) and H(curl) (Ph.D. thesis), Technische Universität Darmstadt, Darmstadt, 2022.

[38] A. Quarteroni, A. Valli, Numerical Approximation of Partial Differential Equations, in: Springer Series in Computational Mathematics, vol. 23, Springer-Verlag, Berlin, 1994.

[39] P.A. Raviart, J.M. Thomas, A mixed finite element method for 2-nd order elliptic problems, in: I. Galligani, E. Magenes (Eds.), Mathematical Aspects of Finite Element Methods, Springer Berlin Heidelberg, Berlin, Heidelberg, 1977, pp. 292–315.

[40] J.J. Dongarra., I.S. Duff., D.C. Sorensen, H.V.D. Vorst, Solving Linear Systems on Vector and Shared Memory Computers, Society for Industrial and Applied Mathematics, 1990.

[41] J. Schöberl, NETGEN an advancing front 2D/3D-mesh generator based on abstract rules, Comput. Vis. Sci. 1 (1997) 41–52.

[42] J. Ahrens, B. Geveci, C. Law, ParaView: An end-user tool for large data visualization, in: Visualization Handbook, Elesvier, 2005.

[43] M.A. Mikhaylenko, An improved regularized fundamental solution to the 2-D steady-state Stokes equation, Eng. Anal. Bound. Elem. 91 (2018) 103–109.

[44] S. Gümgüm, L.C. Wrobel, DRBEM formulation for transient Stokes flow with slip boundary condition, Eng. Anal. Bound. Elem. 75 (2017) 65–78.

[45] R. Araya, C. Harder, A.H. Poza, F. Valentin, Multiscale hybrid-mixed method for the Stokes and Brinkman equations—the method, Comput. Methods Appl. Mech. Engrg. 324 (2017) 29–53.

[46] B. Tavakol, G. Froehlicher, D.P. Holmes, H.A. Stone, Extended lubrication theory: Improved estimates of flow in channels with variable geometry, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 473 (2206) (2017) 20170234.

[47] Y. Xue, S.L. Waters, L.N. Trefethen, Computation of two-dimensional Stokes flows via lightning and AAA rational approximation, SIAM J. Sci. Comput. 46 (2) (2024) A1214–A1234.

[48] J.R. Womersley, Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known, J. Physiol. 127 (3) (1955) 553–563.