A novel explicit finite element scheme for laminar incompressible Navier-Stokes equations using Bézier elements

Chennakesava Kadapa*

Swansea Academy of Advanced Computing, Swansea University, Singleton Campus, Swansea SA2 8PP, Wales, UK.

Abstract

We present a novel finite element scheme for computing fully-explicit solutions of incompressible Navier-Stokes equations. The proposed scheme is based on Bézier elements and employs inf-sup stable velocity-pressure combinations that are easily amenable for the explicit schemes using the standard lumped-mass matrices. We illustrate the inf-sup stability of the proposed velocity-pressure combinations by computing numerical inf-sup constants and demonstrate the spatial convergence of the proposed scheme by computing error norms for the benchmark example of Kovasznay flow. We demonstrate the accuracy of the proposed scheme and its suitability for computing steady and unsteady solutions of laminar flows by studying the benchmark examples of lid-driven cavity flow and flow past a circular cylinder. We show that the use of higher-order inf-sup stable elements and a single-step explicit time integration scheme yield significant computational benefits in computing accurate numerical solutions of laminar incompressible Navier-Stokes.

Keywords: Incompressible Navier-Stokes; Explicit CFD; Bézier elements; Generalised-alpha scheme; Artificial compressibility; Vortex shedding

1. Introduction

Navier-Stokes equations which model fluid flow are one of the most widely used partial differential equations in the fields of science and engineering. With the advent of computers, development of accurate, robust and computationally efficient schemes for obtaining numerical solutions of Navier-Stokes equations has been one of the principal areas of research in computational engineering and numerical mathematics. The amount of literature available on the numerical schemes for Navier-Stokes equations is quite exhaustive, see [1–11] and references therein for an in-depth discussion of computer numerical methods for fluid flow problems.

The vast majority of schemes available for numerical solutions of Navier-Stokes belong to either the finite difference method (FDM) or finite volume method (FVM) or finite element method (FEM). Finite volume methods have been the front runners of computational fluid dynamics (CFD) and still enjoy a significant share among the commercial software suites for CFD. Thanks to the research and developments during the past couple of decades, see [7, 11–14] and references therein, finite element methods are not only proving to be strong competitors to finite volume methods but also are surpassing them in simulating multiphysics problems, see [15, 16]; several recent open-source and commercial software tools, for example, COMSOL Multiphysics, Autodesk CFD, deal.ii, FEBio and FEniCS, are based on FEM. The increased interest in the finite element methods for applications in CFD is because of the advantages of finite element methods in modelling complex geometries and multiphysics phenomenon such as fluid-structure interaction.

It is now an established fact that accurate numerical results can be obtained by using coarse meshes with higher-order spatial discretisations instead of using very fine meshes with the standard linear finite elements [5, 17, 18]. Therefore, higher-order finite element methods have the potential to yield significant benefits for the explicit schemes

*Corresponding author
Email address: c.kadapa@swansea.ac.uk (Chennakesava Kadapa)

since the stable time step size for explicit schemes is directly proportional the characteristic length of the smallest element in the mesh. However, despite the tremendous amount of research work that has gone into the development of finite element techniques for obtaining accurate numerical solutions for compressible and incompressible Navier-Stokes equations, robust and efficient finite element techniques for performing explicit simulations are still lacking.

This lack of development is fundamentally due to the adherence of the finite element community to the Lagrange family of elements which do not offer any inf-sup stable velocity-pressure combinations that are directly amenable for explicit schemes for incompressible flow problems. Taylor-Hood elements [19], which are inf-sup stable combinations of Lagrange elements, are limited only to implicit schemes for incompressible Stokes and Navier-Stokes because of the inapplicability of higher-order Lagrange elements to explicit schemes. As only linear elements of the Lagrange family are useful for explicit schemes, explicit finite element schemes for incompressible Navier-Stokes equations are entirely limited to linear finite elements. In fact, the list of such explicit finite element schemes available in the literature is short. To the author's knowledge, the characteristic-based split scheme of Zienkiewicz and co-workers [11, 20, 21] is the only fully-explicit scheme for incompressible Navier-Stokes that has been extensively studied and extended to other flow problems [22–26].

For higher-order finite element spaces that are amenable to explicit dynamic simulations, the paradigm of isogeometric analysis (IGA) offers the best possible option, see [17] and references therein for the comprehensive discussion. IGA and its variants offer several inf-sup stable combinations that are suitable for explicit schemes: velocity-pressure combinations of B-Splines by Rüberg and Cirak [27, 28] and displacement-pressure NURBS combinations by Kadapa et al. [29] are two such inf-sup stable combinations explored so far. Although these spaces can be adapted to develop explicit schemes for incompressible fluid flow problems, they do pose certain limitations in developing an explicit finite element framework that is applicable to a generic class of problems, especially those composed of complex geometries. The disadvantage of B-Splines is that they are limited to problems with rectangular domains and, with IGA in its early stages of commercialisation, NURBS lack generic spatial discretisation tools for complex geometries. Moreover, generation of inf-sup stable B-Spline and NURBS meshes with local refinements is still an open problem.

Another set of inf-sup stable combinations of finite element spaces that have not been explored so far are those based on Bézier elements. One of the reasons for the limited use of Bézier elements in computational mechanics is the lack of generic mesh generation tools with Bézier elements for problems with complex geometries. Only recently, such generic mesh tools are being explored, see [30–33]. Nevertheless, our experience [34] indicates that quadratic Bézier elements with the standard Bernstein polynomials can be generated by applying a simple linear mapping over the corresponding Lagrange elements that can be generated using the existing mesh generators, thus avoiding the need for new mesh generators. In this work, we extend the mapping technique of [34] to quadratic Bézier quadratic elements and propose a novel explicit finite element scheme for incompressible Navier-Stokes by adapting the inf-sup stable Bézier elements thus generated. Thus, our present work addresses a critical issue associated with performing efficient explicit finite element simulations of incompressible Navier-Stokes by adapting inf-sup stable velocity-pressure combinations that are directly amenable for explicit simulations. The proposed scheme is robust and computationally efficient and is applicable to problems with complex geometries.

The rest of the paper is organised as follows. The governing equations, the finite element formulation and the time integration scheme used in the present work are detailed in Section 2. In Section 3, Bernstein polynomials and Bézier elements are introduced, and the generation of finite element meshes with quadratic Bézier elements using the existing mesh generators is discussed. Section 4 is concerned with the spatial convergence and accuracy studies using the benchmark examples. The paper is concluded with Section 5 with a summary of the observations made and conclusions drawn.

2. Formulation

The finite element scheme proposed in this work is a single-step fully explicit scheme based on the velocity-pressure mixed finite element formulation and the explicit version of the generalised-alpha scheme proposed of Jansen et al. [35]. For the sake of simplicity, fluid flow is assumed to be laminar, viscous, isothermal and incompressible. The proposed scheme can be extended straightforwardly to general incompressible flows, including the turbulence models, as well as compressible flows.

2.1. Governing equations

The development of fully explicit schemes for the incompressible Navier-Stokes is hindered by the absence of the time derivative term in the continuity equation. However, this difficulty is usually overcome by employing the artificial compressibility method proposed by Chorin [36]. According to the artificial compressibility technique, the continuity equation is modified by including a temporal derivative of the pressure field so that fully explicit numerical schemes can be developed. The artificial compressibility method has been extensively used to compute explicit solutions of incompressible Navier-Stokes using finite difference, finite volume, finite element and spectral methods alike, see [24, 37–42] and references therein.

Using the artificial compressibility method, the initial-boundary value problem for a laminar, viscous and incompressible fluid is stated as:
Given $\boldsymbol{g} : \Omega \to \mathbb{R}^3$; $\bar{\boldsymbol{v}} : \Gamma_{\mathrm{D}} \to \mathbb{R}^3$; and $\bar{\boldsymbol{t}} : \Gamma_{\mathrm{N}} \to \mathbb{R}^3$, find velocity, $\boldsymbol{v} : \Omega \to \mathbb{R}^3$; and pressure, $p : \Omega \to \mathbb{R}$, such that:

$$\rho \frac{\partial \boldsymbol{v}}{\partial t} + \rho(\boldsymbol{v} \cdot \nabla)\boldsymbol{v} - \mu \nabla^2 \boldsymbol{v} + \nabla p = \boldsymbol{g} \qquad \text{in} \quad \Omega \tag{1a}$$

$$\frac{\partial p}{\partial t} + \beta^2 \, \nabla \cdot \boldsymbol{v} = 0 \qquad \text{in} \quad \Omega \tag{1b}$$

$$\boldsymbol{v} = \bar{\boldsymbol{v}} \qquad \text{on} \quad \Gamma_{\mathrm{D}} \tag{1c}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\boldsymbol{t}} \qquad \text{on} \quad \Gamma_{\mathrm{N}} \tag{1d}$$

$$\boldsymbol{v}(\cdot, 0) = \boldsymbol{v}_0 \qquad \text{in} \quad \Omega \tag{1e}$$

$$p(\cdot, 0) = p_0 \qquad \text{in} \quad \Omega \tag{1f}$$

where, ρ is the density of the fluid, μ is the dynamic viscosity of the fluid, $\boldsymbol{g}$ is the body force, $\mathbf{n}$ is the unit outward normal on the boundary, Γ, of Ω, β is an artificial compressibility parameter and $\boldsymbol{\sigma}$ is the (pseudo) viscous stress which is given as,

$$\boldsymbol{\sigma} = \mu \nabla \boldsymbol{v} - p \, \mathbf{I}, \tag{2}$$

Γ_{D} is the part of the boundary of the domain Ω where Dirichlet boundary condition $\bar{\boldsymbol{v}}$ is applied and Γ_{N} is the part of the boundary of the domain Ω where Neumann boundary condition $\bar{\boldsymbol{t}}$ is applied. Here, $\Gamma = \Gamma_{\mathrm{D}} \cup \Gamma_{\mathrm{N}}$ and $\Gamma_{\mathrm{D}} \cap \Gamma_{\mathrm{N}} = \emptyset$. $\boldsymbol{v}_0$ and p_0 are the initial velocity and pressure fields.

2.2. Mixed formulation

With $\boldsymbol{w}$ and q as the test function for the velocity and the pressure fields, mixed formulation for the governing equations (1) can be written as,

$$\int_\Omega \boldsymbol{w} \cdot \rho \frac{\partial \boldsymbol{v}}{\partial t} \, \mathrm{d}\Omega + \int_\Omega \boldsymbol{w} \cdot \rho(\boldsymbol{v} \cdot \nabla)\boldsymbol{v} \, \mathrm{d}\Omega + \int_\Omega \mu \, \nabla \boldsymbol{w} : \nabla \boldsymbol{v} \, \mathrm{d}\Omega - \int_\Omega (\nabla \cdot \boldsymbol{w}) \, p \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{w} \cdot \boldsymbol{g} \, \mathrm{d}\Omega + \int_{\Gamma_{\mathrm{N}}} \boldsymbol{w} \cdot \bar{\boldsymbol{t}} \, \mathrm{d}\Gamma \tag{3a}$$

$$\int_\Omega q \, \frac{\partial p}{\partial t} \, \mathrm{d}\Omega + \beta^2 \int_\Omega q \, (\nabla \cdot \boldsymbol{v}) \, \mathrm{d}\Omega = 0 \tag{3b}$$

Now, considering the approximations for velocity and pressure as,

$$\boldsymbol{v} = \mathbf{N}_v \mathbf{v}; \quad \boldsymbol{w} = \mathbf{N}_v \mathbf{w} \tag{4}$$

$$p = \mathbf{N}_p \mathbf{p}; \quad q = \mathbf{N}_p \mathbf{q} \tag{5}$$

and using them in the mixed formulation (3), we get,

$$\mathbf{M}_v \, \dot{\mathbf{v}} = \mathbf{F}^{\mathrm{ext},v} - \mathbf{F}^{\mathrm{int},v} \tag{6a}$$

$$\mathbf{M}_p \, \dot{\mathbf{p}} = -\mathbf{F}^{\mathrm{int},p} \tag{6b}$$

where,

$$\mathbf{M}_v = \int_\Omega \rho\, \mathbf{N}_v^{\mathrm{T}}\, \mathbf{N}_v \, \mathrm{d}\Omega \tag{7}$$

$$\mathbf{M}_p = \int_\Omega \mathbf{N}_p^{\mathrm{T}}\, \mathbf{N}_p \, \mathrm{d}\Omega \tag{8}$$

$$\mathbf{F}^{\mathrm{ext},v} = \int_\Omega \mathbf{N}_v^{\mathrm{T}}\, \boldsymbol{g} \, \mathrm{d}\Omega + \int_{\Gamma_{\mathrm{N}}} \mathbf{N}_v^{\mathrm{T}} \bar{\boldsymbol{t}} \, \mathrm{d}\Gamma \tag{9}$$

$$\mathbf{F}^{\mathrm{int},v} = \int_\Omega \mathbf{N}_v^{\mathrm{T}}\, \rho(\boldsymbol{v}\cdot\nabla)\boldsymbol{v} \, \mathrm{d}\Omega + \int_\Omega \mathbf{B}^{\mathrm{T}}\, \boldsymbol{\sigma}(\boldsymbol{v}, p) \, \mathrm{d}\Omega \tag{10}$$

$$\mathbf{F}^{\mathrm{int},p} = \beta^2 \int_\Omega \mathbf{N}_p^{\mathrm{T}}\, (\nabla\cdot\boldsymbol{v}) \, \mathrm{d}\Omega. \tag{11}$$

The matrix $\mathbf{B}$ for the two-dimensional problem is given as,

$$\mathbf{B} = \begin{bmatrix} \frac{\partial N_v}{\partial x} & 0 \\ 0 & \frac{\partial N_v}{\partial y} \\ \frac{\partial N_v}{\partial y} & \frac{\partial N_v}{\partial x} \end{bmatrix} \tag{12}$$

The set of equations (6a) and (6b) can be combined into a single equation as,

$$\mathbf{M}\,\dot{\mathbf{V}} = \mathbf{F} \tag{13}$$

where,

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_v & 0 \\ 0 & \mathbf{M}_p \end{bmatrix}, \mathbf{F} = \begin{Bmatrix} \mathbf{F}^{\mathrm{ext},v} - \mathbf{F}^{\mathrm{int},v} \\ -\mathbf{F}^{\mathrm{int},p} \end{Bmatrix} \quad \text{and} \quad \mathbf{V} = \begin{Bmatrix} \mathbf{v} \\ \mathbf{p} \end{Bmatrix}. \tag{14}$$

2.3. Time integration scheme

Equation (13) is a first-order ordinary differential equation which can be solved using any suitable single-step or multi-step explicit scheme, for example, mid-point method or explicit Runge-Kutta methods or Adams-Bashforth methods. In this work, we use the explicit version of the generalised-alpha scheme proposed by Jansen et al. [35] because of simplicity and single-step nature.

Now, using the explicit generalised-alpha scheme for the solutions of (13), we get,

$$\mathbf{M}\,\dot{\mathbf{V}}_{n+\alpha_m} = \mathbf{F}_n \tag{15}$$

with,

$$\dot{\mathbf{V}}_{n+\alpha_m} = \alpha_m\, \dot{\mathbf{V}}_{n+1} + (1-\alpha_m)\, \dot{\mathbf{V}}_n \tag{16}$$

$$\mathbf{V}_{n+1} = \mathbf{V}_n + \Delta t \left[\gamma\, \dot{\mathbf{V}}_{n+1} + (1-\gamma)\, \dot{\mathbf{V}}_n\right]. \tag{17}$$

It can be shown from the spectral analysis of the linear problem that this explicit scheme is second-order accurate for $\gamma = 0.5 + \alpha_m$ and that it is stable for $\alpha_m > 1/2$, for any stable time step. For simplicity, $\alpha_m = 1.0$ is assumed in this work.

Equations (15)-(17) may be written in terms of velocity and pressure DOFs as,

$$\mathbf{M}_v\,\dot{\mathbf{v}}_{n+1} = \mathbf{F}_n^{\text{ext},v} - \mathbf{F}_n^{\text{int},v} \tag{18a}$$

$$\mathbf{M}_p\,\dot{\mathbf{p}}_{n+1} = -\mathbf{F}_n^{\text{int},p} \tag{18b}$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \Delta t\,[\gamma\,\dot{\mathbf{v}}_{n+1} + (1-\gamma)\,\dot{\mathbf{v}}_n] \tag{18c}$$

$$\mathbf{p}_{n+1} = \mathbf{p}_n + \Delta t\,[\gamma\,\dot{\mathbf{p}}_{n+1} + (1-\gamma)\,\dot{\mathbf{p}}_n] \tag{18d}$$

Thus, once the equations (18a) and (18b) are solved for $\dot{\mathbf{v}}_{n+1}$ and $\dot{\mathbf{p}}_{n+1}$, respectively, $\mathbf{v}_{n+1}$ and $\mathbf{p}_{n+1}$ can be computed, respectively, from (18c) and (18d).

2.4. Inf-sup stability

It is now an established fact that the combinations of finite element spaces for the velocity and the pressure fields in the mixed formulation (3) must be chosen such that the combination is inf-sup stable, see [19] for further details. Failure to select an inf-sup stable velocity-pressure combination for the solution of (18) results in spurious oscillations in the pressure field. This issue is circumvented in the CBS scheme by using the Taylor-Galerkin formulation which adds a stabilisation term to the formulation. However, in this work, we use inf-sup stable combinations of Bézier elements, thus avoiding the need for additional stabilisations that involve ad-hoc stabilisation parameters. Quadratic Bézier elements are used for discretising the velocity field and the linear elements for the pressure field. Thus, the approximation functions $\mathbf{N}_v$ are the quadratic Bernstein polynomials and $\mathbf{N}_p$ are the corresponding linear Bernstein polynomials.

3. Bézier elements

The finite elements used in this work, known as Bézier elements, are based on the Bernstein polynomials over triangular and quadrilateral domains in 2D and tetrahedral and hexahedral domains in 3D, and they are generated using the existing mesh generators by exploiting the property of the Bézier curve that its end control points are interpolatory. The techniques used for mesh generation and the application of Dirichlet boundary conditions are already discussed in Kadapa [34] for triangular and tetrahedral elements. In this work, we extend these mapping techniques to quadrilateral elements.

Bernstein polynomials are the special class of polynomials that form the base of geometry modelling in computer graphics and computer-aided design. B-splines and NURBS are a generalisation of Bernstein polynomials over extended domains. A brief overview of the Bernstein polynomials, their important properties that form the basis for the present work and the steps involved in transforming the Lagrange elements into Bézier elements are presented in the following subsections. For comprehensive details on Bernstein polynomials and other related concepts discussed in this section, we refer the reader to Piegl and Tiller [43] and Kadapa [34].

Bernstein polynomials are the same as Lagrange polynomials for the zeroth and first order, and they are different for higher-order cases. As this work is focussed only on the quadratic elements, further discussion is limited to quadratic Bernstein polynomials and the corresponding Bézier elements only.

For the quadratic case, i.e., for $l, m, n = 2$, there are three univariate Bernstein polynomials, six bivariate polynomials over the triangular domain, ten trivariate polynomials over the tetrahedral domain, nine bivariate polynomials over the quadrilateral domain and twenty-seven trivariate polynomials over the hexahedral domain. For the triangular and quadrilateral elements shown in Fig. 1 in the local (or parametric) coordinates (ξ_1, ξ_2), the quadratic Bernstein polynomials over a triangular domain are given by,

$$N_1(\xi_1,\xi_2) = (1-\xi_1-\xi_2)^2 \tag{19a}$$

$$N_2(\xi_1,\xi_2) = \xi_1^2 \tag{19b}$$

$$N_3(\xi_1,\xi_2) = \xi_2^2 \tag{19c}$$

$$N_4(\xi_1,\xi_2) = 2\,\xi_1\,(1-\xi_1-\xi_2) \tag{19d}$$

$$N_5(\xi_1,\xi_2) = 2\,\xi_1\,\xi_2 \tag{19e}$$

$$N_6(\xi_1,\xi_2) = 2\,\xi_2\,(1-\xi_1-\xi_2) \tag{19f}$$

and the quadratic Bernstein polynomials over a quadrilateral domain are given by,

$$N_1(\xi_1,\xi_2) = (1-\xi_2)^2 \quad (1-\xi_1)^2 \tag{20a}$$
$$N_5(\xi_1,\xi_2) = (1-\xi_2)^2 \quad 2\,\xi_1(1-\xi_1) \tag{20b}$$
$$N_2(\xi_1,\xi_2) = (1-\xi_2)^2 \quad \xi_1^2 \tag{20c}$$
$$N_8(\xi_1,\xi_2) = 2\,\xi_2(1-\xi_2) \quad (1-\xi_1)^2 \tag{20d}$$
$$N_9(\xi_1,\xi_2) = 2\,\xi_2(1-\xi_2) \quad 2\,\xi_1(1-\xi_1) \tag{20e}$$
$$N_6(\xi_1,\xi_2) = 2\,\xi_2(1-\xi_2) \quad \xi_1^2 \tag{20f}$$
$$N_4(\xi_1,\xi_2) = \xi_2^2 \quad (1-\xi_1^2) \tag{20g}$$
$$N_7(\xi_1,\xi_2) = \xi_2^2 \quad 2\,\xi_1(1-\xi_1) \tag{20h}$$
$$N_3(\xi_1,\xi_2) = \xi_2^2 \quad \xi_1^2 \tag{20i}$$

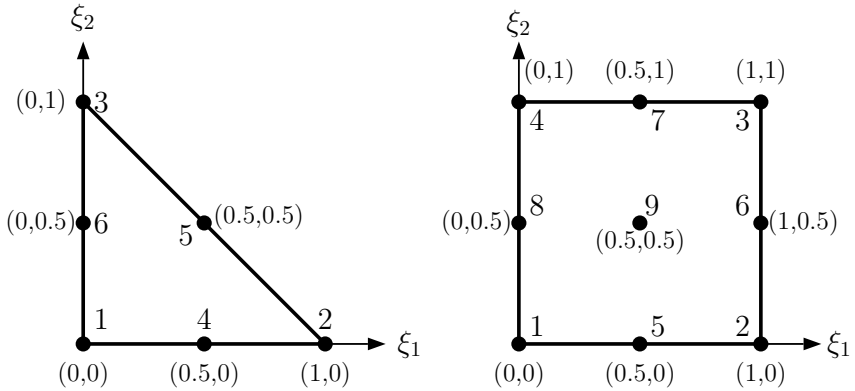


Figure 1: Quadratic triangular and quadrilateral elements in the parametric (or local) coordinates.

It can be observed from equations (19)-(20) that the Bernstein polynomials are non-negative within the parametric interval $[0, 1]$; therefore, the coefficients of consistent mass matrices computed with Bernstein polynomials as the basis functions are all non-negative, thus rendering row-sum-lumped mass matrices with positive values for all the diagonal entries, unlike the quadratic and higher-order Lagrange polynomials which result in zero diagonal entries for corner vertices when used with row-sum mass lumping. This non-negative property of the Bernstein polynomials is the crucial property that renders them applicable for explicit dynamic simulations in a straightforward manner by using row-sum-lumped mass matrices.

3.1. Mesh generation

Finite element discretisations with quadratic Bézier elements are generated from the corresponding Lagrange elements using a simple mapping technique, thus facilitating the mesh generation using existing mesh generators. This mapping technique is already discussed in Kadapa [34] for triangular and tetrahedral elements. Therefore, in this work, we provide only the final expressions for the mapping and the refer the reader to Kadapa [34] for further details.

Assuming that the edge-nodes for the Lagrange elements are exactly in the middle, the control points for the quadratic Bézier triangular element with the numbering sequence as shown in Fig. 2 can be computed using the

following relations:

$$\mathbf{P}_1 = \mathbf{X}_1 \tag{21a}$$
$$\mathbf{P}_2 = \mathbf{X}_2 \tag{21b}$$
$$\mathbf{P}_3 = \mathbf{X}_3 \tag{21c}$$
$$\mathbf{P}_4 = 2\,[\mathbf{X}_4 - 0.25\,\mathbf{X}_1 - 0.25\,\mathbf{X}_2] \tag{21d}$$
$$\mathbf{P}_5 = 2\,[\mathbf{X}_5 - 0.25\,\mathbf{X}_2 - 0.25\,\mathbf{X}_3] \tag{21e}$$
$$\mathbf{P}_6 = 2\,[\mathbf{X}_6 - 0.25\,\mathbf{X}_3 - 0.25\,\mathbf{X}_1] \tag{21f}$$

Similarly, the control points for the quadratic Bézier quadrilateral element shown in Fig. 3 can be computed using,

$$\mathbf{P}_1 = \mathbf{X}_1 \tag{22a}$$
$$\mathbf{P}_2 = \mathbf{X}_2 \tag{22b}$$
$$\mathbf{P}_3 = \mathbf{X}_3 \tag{22c}$$
$$\mathbf{P}_4 = \mathbf{X}_4 \tag{22d}$$
$$\mathbf{P}_5 = 2\,[\mathbf{X}_5 - 0.25\,\mathbf{X}_1 - 0.25\,\mathbf{X}_2] \tag{22e}$$
$$\mathbf{P}_6 = 2\,[\mathbf{X}_6 - 0.25\,\mathbf{X}_2 - 0.25\,\mathbf{X}_3] \tag{22f}$$
$$\mathbf{P}_7 = 2\,[\mathbf{X}_7 - 0.25\,\mathbf{X}_3 - 0.25\,\mathbf{X}_4] \tag{22g}$$
$$\mathbf{P}_8 = 2\,[\mathbf{X}_8 - 0.25\,\mathbf{X}_4 - 0.25\,\mathbf{X}_1] \tag{22h}$$

For the center control point of the quadrilateral elements i.e., $\mathbf{P}_9$, either of the following two mappings can be used.

$$\mathbf{P}_9 = \mathbf{X}_9 \tag{23a}$$
$$\mathbf{P}_9 = 4\,[\mathbf{X}_9 - 0.0625\,(\mathbf{X}_1 + \mathbf{X}_2 + \mathbf{X}_3 + \mathbf{X}_4) - 0.125\,(\mathbf{X}_5 + \mathbf{X}_6 + \mathbf{X}_7 + \mathbf{X}_8)] \tag{23b}$$

Similarly, the control points for the quadratic Bézier tetrahedron and hexahedron elements can be computing using the nodes of corresponding quadratic Lagrange elements. Thus, coordinates of all the control points for the quadratic Bézier element(s) can be computed from the nodal coordinates of the quadratic Lagrange element(s) by a simple mapping technique over the respective edges and faces. In this way, finite element meshes with quadratic Bézier elements can be generated, even for complex geometries, using existing mesh generators. The computational cost of this mapping is negligible when compared with the overall theoretical and computational advantages gained.

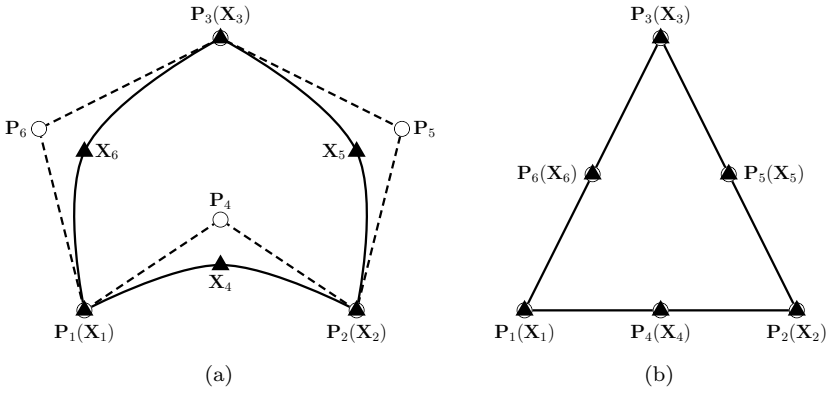


Figure 2: Quadratic Bézier triangular element with (a) curved edges and (b) straight edges. Here ◯ denotes a control point and ▲ denotes a node of the Lagrange element; the dashed line represented the control polygon and the solid line represents the actual geometry of the curve.

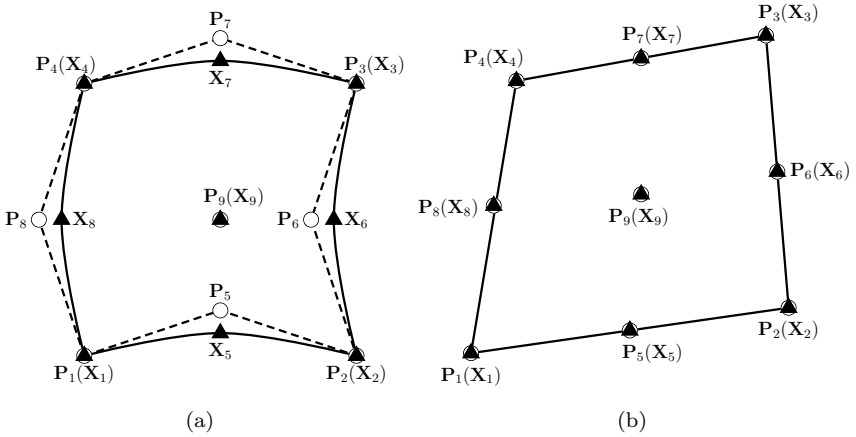


Figure 3: Quadratic Bézier quadrilateral element with (a) curved edges and (b) straight edges. Notations are same as those in Fig. 2.

3.2. Application of boundary conditions

For the present formulation, Neumann boundary conditions can be applied by evaluating the corresponding boundary integrals in the standard manner. However, since all the quadratic Bernstein polynomials are not interpolatory, application of Dirichlet boundary conditions for the quadratic Bézier elements does not seem straightforward. However, we show that Dirichlet boundary conditions for the proposed elements can be applied using the classical elimination approach by extending the mapping technique for the mesh generation proposed in Section. 3.1. The procedure for the imposition of Dirichlet boundary conditions is explained using the quadratic Bézier triangular and quadrilateral elements for the solution of a Laplace equation in two-dimensions. This procedure is straightforward to extend to tetrahedral and hexahedral elements in three dimensions as well as for problems with multiple degrees of freedom (DOFs).

Let us assume $u_1^B, u_2^B, u_3^B, u_4^B, u_5^B, u_6^B, u_7^B, u_8^B$ and u_9^B as the values of DOFs at the respective control points of the Bézier elements shown in Fig. 2; and let us assume that $u_1^L, u_2^L, u_3^L, u_4^L, u_5^L, u_6^L, u_7^L, u_8^L$ and u_9^L are the corresponding DOFs values for the Lagrange elements. Now, following the similar analogy used in deriving Eqs. (21)-(23), we can get the relations for computing the degree of freedom values at the control points for the Bézier triangular element

as,

$$u_1^B = u_1^L \tag{24a}$$
$$u_2^B = u_2^L \tag{24b}$$
$$u_3^B = u_3^L \tag{24c}$$
$$u_4^B = 2\,\left[\,u_4^L - 0.25\,u_1^L - 0.25\,u_2^L\,\right] \tag{24d}$$
$$u_5^B = 2\,\left[\,u_5^L - 0.25\,u_2^L - 0.25\,u_3^L\,\right] \tag{24e}$$
$$u_6^B = 2\,\left[\,u_6^L - 0.25\,u_3^L - 0.25\,u_1^L\,\right] \tag{24f}$$

and the relations for computing the degree of freedom values at the control points for the Bézier quadrilateral element as,

$$u_1^B = u_1^L \tag{25a}$$
$$u_2^B = u_2^L \tag{25b}$$
$$u_3^B = u_3^L \tag{25c}$$
$$u_4^B = u_4^L \tag{25d}$$
$$u_5^B = 2\,\left[\,u_5^L - 0.25\,u_1^L - 0.25\,u_2^L\,\right] \tag{25e}$$
$$u_6^B = 2\,\left[\,u_6^L - 0.25\,u_2^L - 0.25\,u_3^L\,\right] \tag{25f}$$
$$u_7^B = 2\,\left[\,u_7^L - 0.25\,u_3^L - 0.25\,u_4^L\,\right] \tag{25g}$$
$$u_8^B = 2\,\left[\,u_8^L - 0.25\,u_4^L - 0.25\,u_1^L\,\right] \tag{25h}$$

We now assess the correctness of the mappings for control points and DOFs given by Eqs. (21)-(23) and (24)-(25), by computing the solutions of the Laplace equation,

$$\nabla^2 u = 0 \tag{26}$$

over a circular domain $[1 \leq r \leq 2] \times [0 \leq \theta \leq \pi/2]$, and with the analytical solution,

$$u(r,\theta) = \frac{2}{3}\,\left(r - \frac{1}{r}\right)\sin\theta. \tag{27}$$

The domain of the problem and the analytical solution is chosen such that the correctness of the mapping for control points can be tested for geometries with curved edges and the mapping for DOFs can be tested for non-homogeneous Dirichlet boundary conditions. Convergence studies are performed over a set of successfully refined triangular and quadrilateral meshes shown, respectively, in Figs. 4 and 5, and the error norms are plotted in Fig. 6. These graphs indicate that the mappings proposed for control points and DOFs result in optimal convergence rates in the solution field.

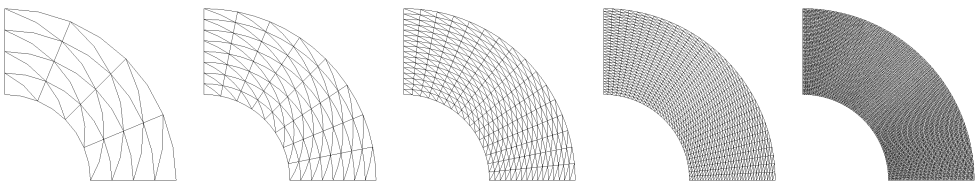


Figure 4: Laplace equation over circular annulus: finite element meshes with triangular elements.

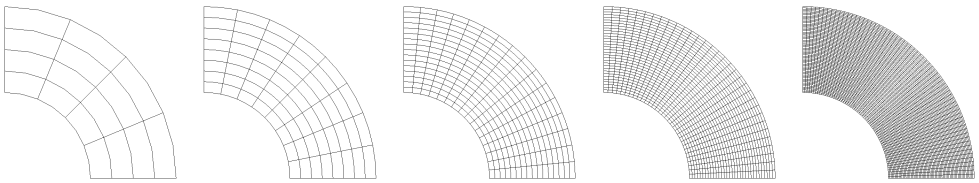
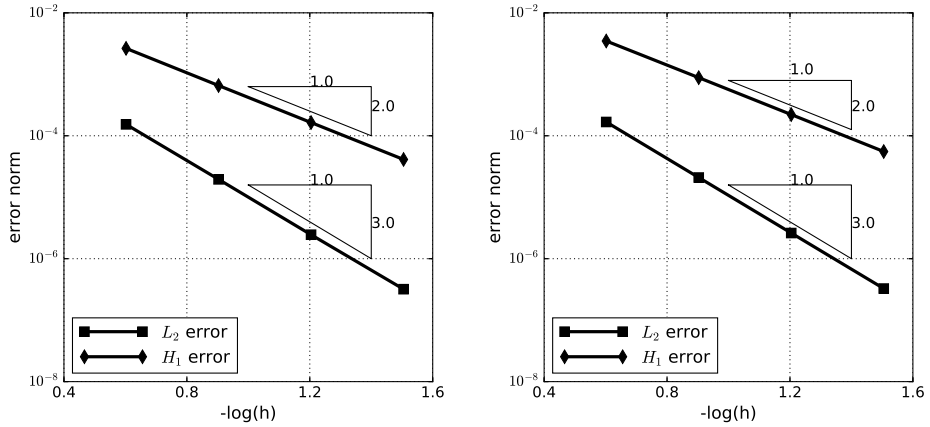


Figure 5: Laplace equation over circular annulus: finite element meshes with quadrilateral elements.



(a) triangular elements (b) quadrilateral elements

Figure 6: Laplace equation over circular annulus: convergence of L_2 and H_1 errors in the solution.

3.3. Inf-sup stability

In this section, we establish the inf-sup stability of the proposed velocity-pressure combinations by computing the numerical inf-sup constants, see [44], for the benchmark example of Kovasznay flow problem [45]. The values of inf-sup constants (β_h) computed for five different meshes are shown in Fig. 7. Inf-sup constants for equal-order approximations for the velocity and pressure fields (TRIA6/6 and QUAD9/9) are also presented for the sake of completeness. These graphs show that both the combinations, TRIA6/3 and QUAD9/4, with quadratic Bézier elements for the velocity field and linear elements for the pressure field are inf-sup stable.

3.4. Lumped mass matrices

Now, the lumped mass matrices for the inf-sup stable velocity-pressure combinations of Bézier elements can be used to obtain computationally efficient solutions of (18a) and (18b). Using row-sum-lumping method, the lumped mass matrices for quadratic Bézier triangular and quadrilateral elements become,

$$\mathbf{M}_v^{e,\text{tria}} = \frac{\rho\, A^e}{6} \quad \text{diag}[\mathbf{1}_6 \quad \mathbf{1}_6] \tag{28}$$

$$\mathbf{M}_v^{e,\text{quad}} = \frac{\rho\, A^e}{9} \quad \text{diag}[\mathbf{1}_9 \quad \mathbf{1}_9] \tag{29}$$

where, A^e is the area of the element, $\mathbf{1}_6 = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$ and $\mathbf{1}_9 = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$. The lumped mass matrices for the pressure DOFs are the same as those for the linear triangle and bilinear quadrilateral elements.

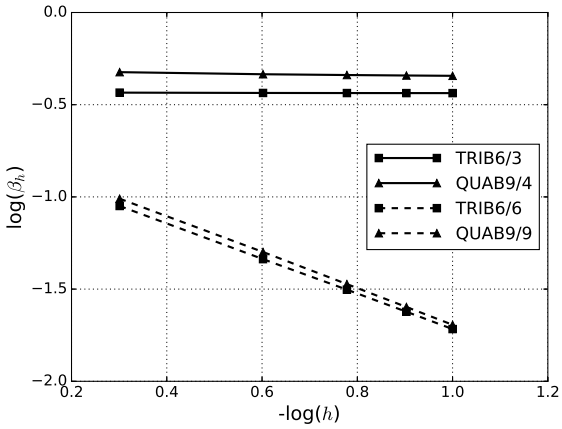


Figure 7: Kovasznay flow: convergence of numerical inf-sup constant with respect to mesh refinement.

4. Numerical examples

Having established the inf-sup stability of the proposed elements, we now assess the accuracy of the numerical solutions obtained using the proposed explicit scheme by studying the benchmark examples of Kovasznay flow, flow in a lid-driven cavity and flow past a cylinder.

Following [46], the stable time step for the proposed explicit scheme is computed as,

$$\Delta t = \frac{C}{\gamma}\,\min(\Delta t_{\text{conv}}, \Delta t_{\text{diff}}) \tag{30}$$

where, C is the CFL number,

$$\Delta t_{\text{conv}} = \frac{h}{|\boldsymbol{v}| + \sqrt{|\boldsymbol{v}|^2 + \beta^2}}, \quad \text{and} \quad \Delta t_{\text{diff}} = \frac{\rho\, h^2}{2\,\mu}. \tag{31}$$

The characteristic length of the element, h, is evaluated as,

$$h = \begin{cases} 0.9\,\text{minimum of length of edges 4-5, 5-6 and 6-4,} & \text{for triangular element} \\ 0.45\,\text{minimum of length of edges 1-2, 2-3, 3-4 and 4-1,} & \text{for quadrilateral element.} \end{cases} \tag{32}$$

In all the simulations,

- element residual vectors are evaluated using three and nine quadrature points, respectively, for the triangular and quadrilateral elements,
- the CFL number is, $C = 0.9$,
- the artificial compressibility parameter is, $\beta = 2.0$ unless stated otherwise,
- the density of the fluid is, $\rho = 1$, and the dynamic viscosity of the fluid, μ, is adjusted based on the Reynolds number, and
- for steady-state solutions, the steady state is assumed to be reached when the relative error norm in the velocity field between the adjacent time steps is lower than 10^{-6} unless specified otherwise.

The artificial compressibility parameter, β, is assumed as a constant value throughout the simulation. This assumption results in a computationally efficient implementation of element subroutines as it allows to compute the

element residual, along with the critical time step, in a single subroutine. The key steps involved in the proposed explicit scheme are presented in the form of a pseudo-code in Algorithm 1. The convergence check is ignored for unsteady solutions. All the simulations are performed on a single Intel i7-7660U CPU (2.5 GHz).

Algorithm 1 Explicit scheme for incompressible Navier-Stokes

Set: ρ, μ, β, ϵ and max_steps
Set: Initial conditions, $\mathbf{v}_0$ and $\mathbf{p}_0$
Compute: $\mathbf{N}_v$, $d\mathbf{N}_v/dx$, $d\mathbf{N}_v/dy$, $\mathbf{N}_p$, A^e and h
Compute: $\mathbf{M}_v$ and $\mathbf{M}_p$
for $steps = 1$ to max_steps **do**
    Compute: $\mathbf{F}_n^{\text{ext}}$, $\mathbf{F}_n^{\text{int},v}$, $\mathbf{F}_n^{\text{int},p}$ and Δt
    Compute: $\dot{\mathbf{v}}_{n+1}$ from (18a) and $\dot{\mathbf{p}}_{n+1}$ from (18b)
    Compute: $\mathbf{v}_{n+1}$ from (18c) and $\mathbf{p}_{n+1}$ from (18d)
    if $\frac{|\mathbf{v}_{n+1}-\mathbf{v}_n|}{|\mathbf{v}_{n+1}|} \leq \epsilon$ **then**
        Converged, exit **for** loop
    end if
    $(\mathbf{v}_n, \dot{\mathbf{v}}_n, \mathbf{p}_n, \dot{\mathbf{p}}_n) \leftarrow (\mathbf{v}_{n+1}, \dot{\mathbf{v}}_{n+1}, \mathbf{p}_{n+1}, \dot{\mathbf{p}}_{n+1})$
end for

4.1. Kovasznay flow - spatial convergence studies

In this example we assess the accuracy of the proposed explicit scheme by computing error norms for the Kovasznay flow problem. The domain of the problem, is $\Omega = [-0.5, 1.5] \times [-0.5, 1.5]$ and the analytical solution is given by,

$$v_x(x, y) = 1.0 - e^{\lambda x} \cos(2\pi y) \tag{33}$$

$$v_y(x, y) = \frac{\lambda}{2\pi} e^{\lambda x} \sin(2\pi y) \tag{34}$$

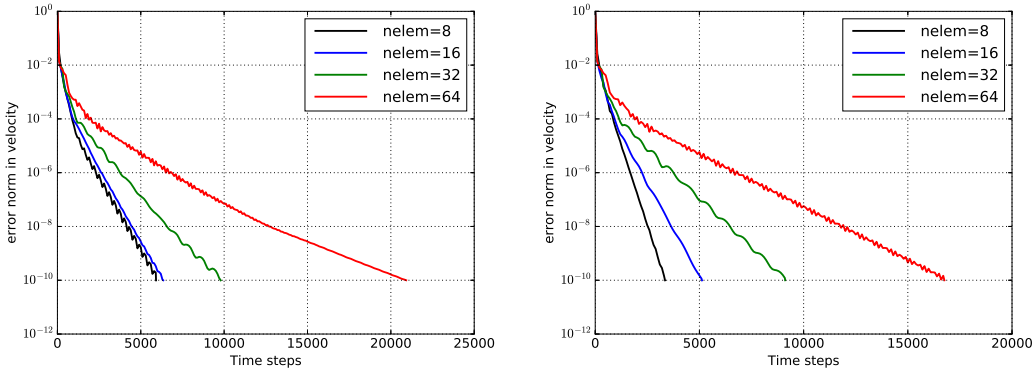
$$p(x, y) = p_0 - \frac{1}{2} e^{2\lambda x} \tag{35}$$

where,

$$\lambda = \frac{Re}{2} - \sqrt{\frac{Re^2}{4} + 4\pi^2}. \tag{36}$$

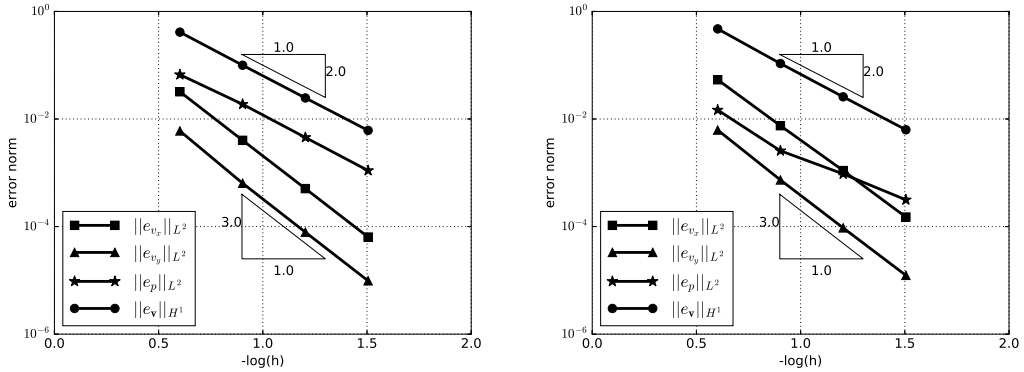
in which Re is the Reynolds number and p_0 is the reference pressure.

As the interest of this example is spatial convergence studies, a lower convergence tolerance of 10^{-10} is considered. Convergence studies are carried out for $Re = 40$ over a set of four successively refined uniform meshes and the evolution of velocity error against the number of time steps is plotted in Fig. 8 for triangular and quadrilateral elements. The values of error norms in the velocity and the pressure fields for the converged solutions obtained using the proposed explicit scheme are presented in Figs. 9 and 10, respectively, for TRIB6/3 and QUAB9/4 elements, along with the error norms obtained with implicit scheme. These graphs indicate that optimal convergence rates are obtained in the velocity field; the slight decrease in the convergence rate in the pressure field for the explicit scheme is attributed to the effects of artificial compressibility in the explicit scheme. Nevertheless, despite the drop in the optimal convergence rate for the pressure field, the absolute values of error norms in pressure obtained with the explicit scheme are about one order of magnitude lower than those obtained with the implicit scheme. Typical contour plots of velocity, pressure and streamlines are shown in Fig. 11 for the $32 \times 32 \times 2$ triangular mesh. As shown, the pressure field obtained using the proposed scheme is completely free from spurious oscillations, which is a direct consequence of using inf-sup stable velocity-pressure combinations.



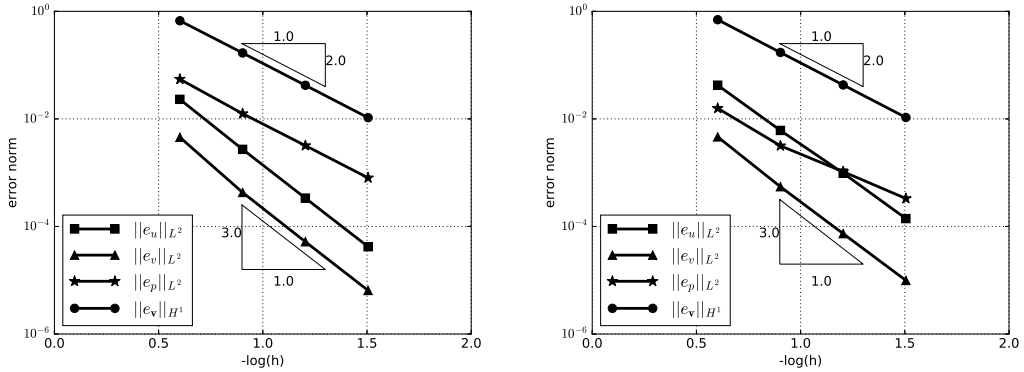
(a) TRIA6/3 elements (b) QUAD9/4 elements

Figure 8: Kovasznay flow: convergence of error norm in velocity. *nelem* refers to the number of elements per side of the domain.



(a) Implicit scheme (b) Explicit scheme

Figure 9: Kovasznay flow: convergence of error norms with TRIA6/3 elements.



(a) Implicit scheme (b) Explicit scheme

Figure 10: Kovasznay flow: convergence of error norms with QUAD9/4 elements.

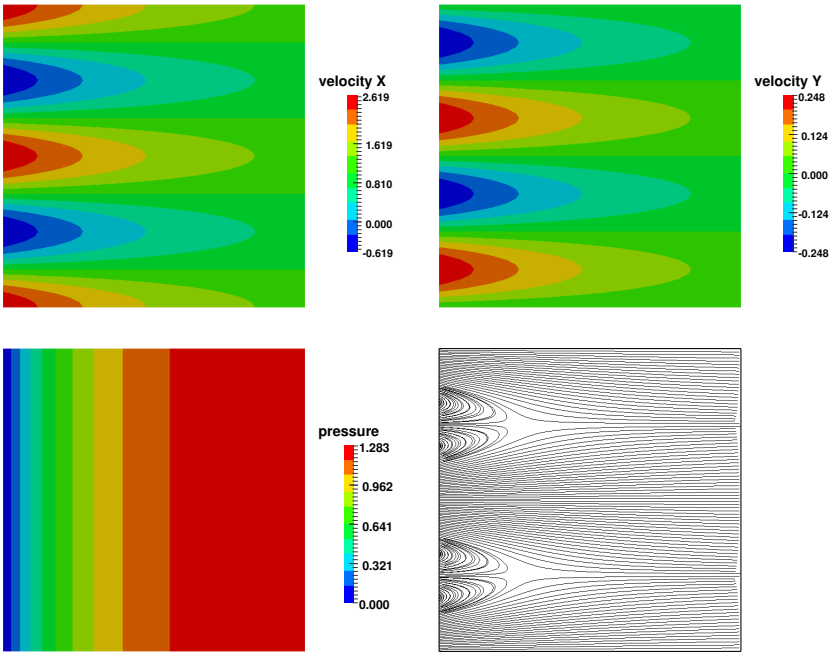


Figure 11: Kovasznay flow: velocity and pressure contour plots, and streamlines with $32 \times 32 \times 2$ TRIA6/3 mesh.

4.2. Lid-driven cavity

The geometry and boundary conditions of the problem are as shown in Fig. 12. The horizontal velocity of the lid is taken as,

$$v_{\text{lid}} = 1 - e^{50(x-1)} - e^{-50x}, \text{for} \quad 0 \leq x \leq 1, \tag{37}$$

so that the velocity transitions smoothly to zero at the side walls. Simulations are performed using three different finite element meshes shown in Fig. 13 for Reynolds numbers 100, 400, 1000, 3200 and 5000. It is important to point out that the finite element meshes used for this example in the present work are significantly coarser than the meshes

usually required with the linear elements. Initial conditions for all the field variables are set as zero, and the velocity of the lid is increased smoothly during the first 5000 time steps using a sinusoidal profile so that the convergence is smooth and accelerated. The artificial compressibility parameter is, $\beta = 0.9$.

Plots of the evolution of velocity error, plots of velocity profiles against reference data and typical contour plots for a subset of simulations are presented in Figs. 14, 15, 16, 17 and 18; and the computer time taken for all the simulations is tabulated in Table. 1. The following observations are made from these results:

- Velocity profiles along horizontal and vertical center lines presented in Figs. 15 and 16 illustrate that the results obtained with the proposed scheme match well with the reference solution of Ghia et al. [47].
- Plots of streamlines in Fig. 17 demonstrate that corner vertices are captured accurately.
- Contour plots of pressure in Fig. 18 prove that the pressure field obtained with the proposed scheme is free from spurious oscillations.
- The results obtained with mesh C indicate that accurate numerical results can be obtained using uniform meshes using the proposed scheme. The benefits of using uniform meshes can be seen in the lesser amount of computational time taken by the mesh C which is uniform throughout when compared with the mesh B which is refined near the walls. Although mesh C has more nodes and elements when compared with the mesh B, it requires only about 50% of the time taken by mesh B which is owing to the direct dependence of the stable time size on the smallest element in the mesh.
- For the same number of nodes (meshes A and B), quadrilateral elements are computationally more expensive when compared with the triangular elements. So, the use of quadrilateral elements may be limited to the zones in the flow field where their use can result in justifiable benefits, for example, in capturing steep gradients in the boundary layers.

From these results and observations, it can be deduced that accurate numerical results can be obtained with the proposed scheme using very coarse meshes, the benefits of which are quite significant for large-scale explicit simulations.

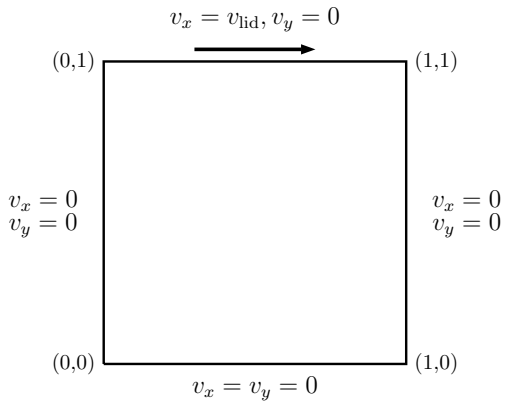
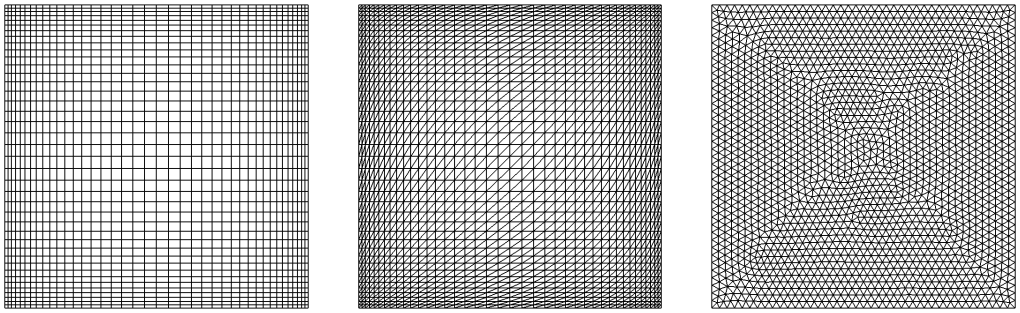


Figure 12: Lid-driven cavity: Geometry and boundary conditions.

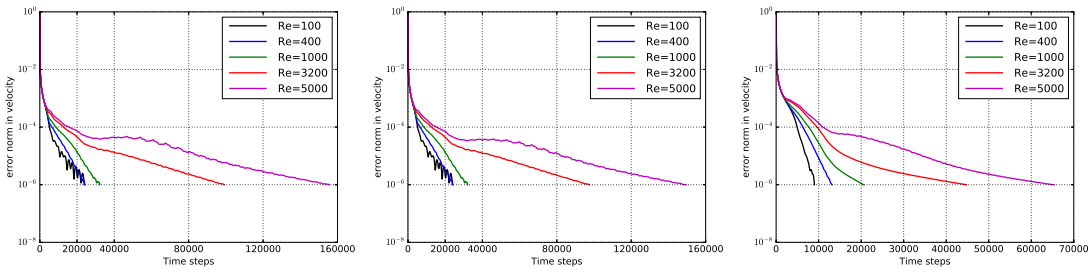


(a) Mesh A (b) Mesh B (c) Mesh C

Figure 13: Lid-driven cavity: finite element meshes used for the simulations. The number of nodes in the meshes A, B and C is 6561, 6561 and 7473, respectively; and the number of elements in the corresponding meshes is 1600, 3200 and 3656, respectively.

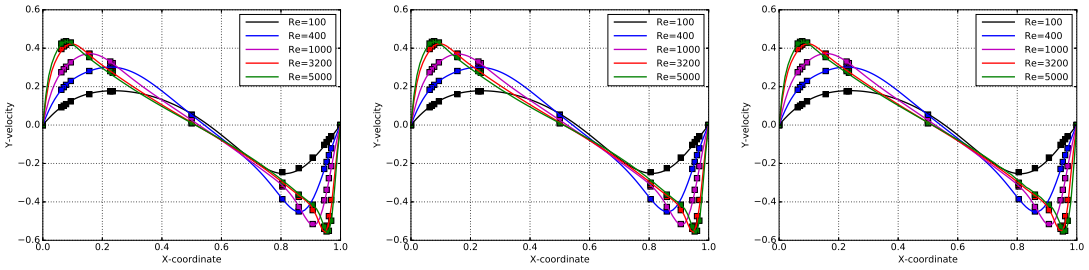
	Reynolds number				
	$Re = 100$	$Re = 400$	$Re = 1000$	$Re = 3200$	$Re = 5000$
Mesh A	32	33	45	139	220
Mesh B	24	24	35	106	157
Mesh C	11	16	25	54	79

Table 1: Lid-driven cavity: time taken in seconds.



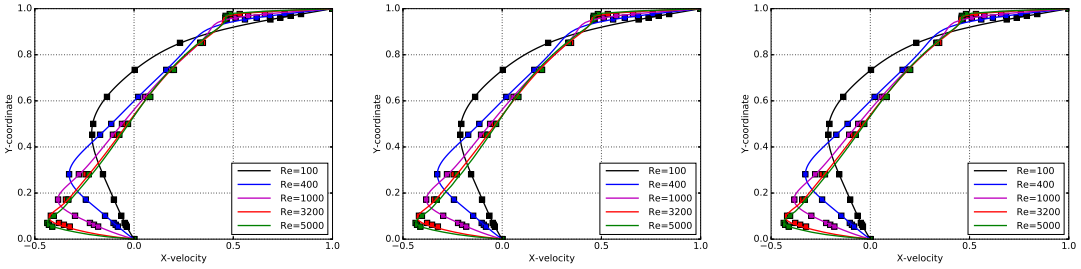
(a) Mesh A (b) Mesh B (c) Mesh C

Figure 14: Lid-driven cavity: convergence of error norm in velocity for the six meshes with different Reynolds numbers.



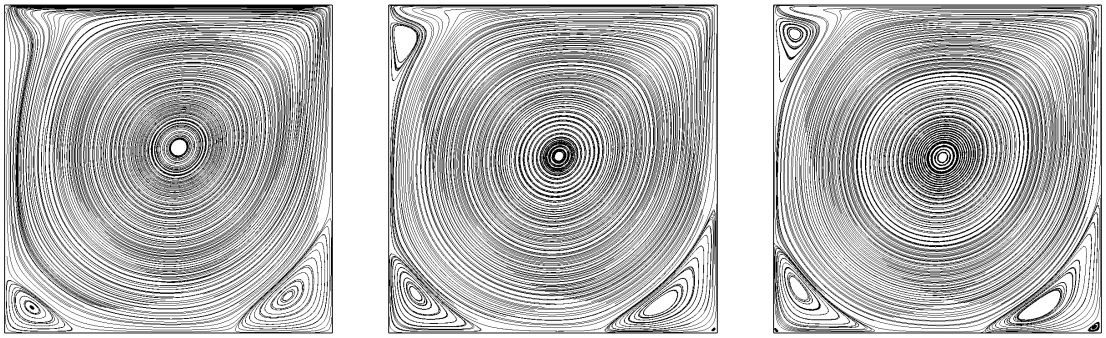
(a) Mesh A (b) Mesh B (c) Mesh C

Figure 15: Lid-driven cavity: Y-velocity profiles along the horizontal center line.



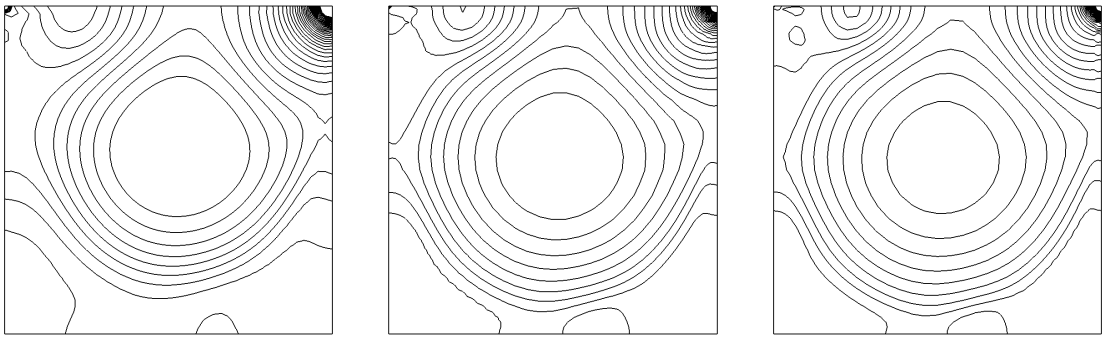
(a) Mesh A (b) Mesh B (c) Mesh C

Figure 16: Lid-driven cavity: X-velocity profiles along the vertical center line.



(a) $Re = 1000$ with mesh C (b) $Re = 3200$ with mesh B (c) $Re = 5000$ with mesh A

Figure 17: Lid-driven cavity: plots of streamlines.



(a) $Re = 1000$ with mesh C (b) $Re = 3200$ with mesh B (c) $Re = 5000$ with mesh A

Figure 18: Lid-driven cavity: contour plots of pressure.

4.3. Flow past a circular cylinder - steady flow

The geometry and boundary conditions for this problem are shown in Fig. 19, and the finite element meshes used for simulations are shown in Fig. 20. For this problem, the initial velocity in X-direction is taken as unity and the other field variable as zero. The inlet velocity is kept constant throughout the simulation. The convergence of velocity norm and the drag coefficient are shown in Fig. 21 for two different values of Reynolds number. The converged values of the drag coefficient and the length of the recirculation zone, presented in Table. 2, are in good agreement with the reference values from the literature. Plots of streamlines and contour plots of pressure obtained with the mesh A are presented, respectively, in Figs. 22 and 23.

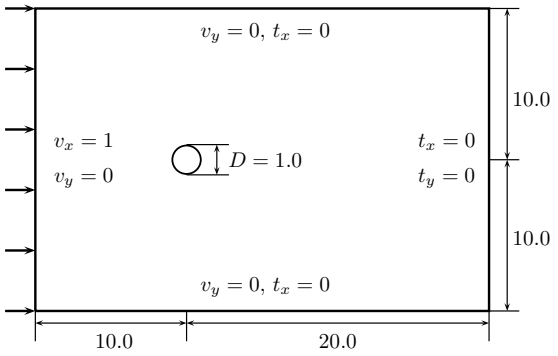
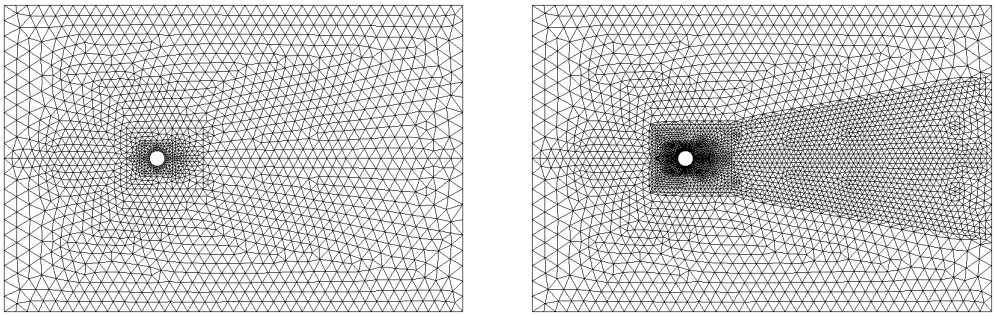


Figure 19: Flow past a cylinder: geometry and boundary conditions.



(a) Mesh A (b) Mesh B

Figure 20: Flow past a cylinder: finite element mesh used for the analysis. Mesh A consists of 6932 nodes and 3394 elements and mesh B consists of 14984 nodes and 7402 elements.

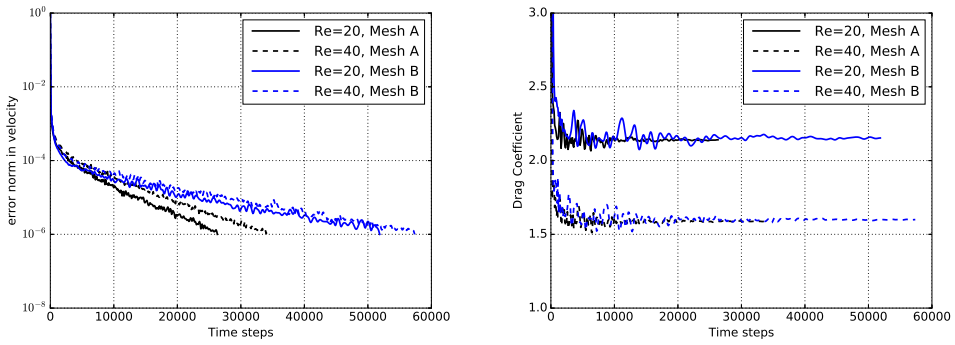


Figure 21: Flow past a cylinder: convergence of error norm in velocity (left) and the drag coefficient (right).

	$Re = 20$		$Re = 40$	
	C_D	L_r	C_D	L_r
Dennis and Chang [48]	2.05	0.94	1.52	2.35
Fornberg [49]	2.00	0.91	1.50	2.24
Linnick and Fasel [50]	2.06	0.93	1.61	2.23
Russell and Wang [51]	2.13	0.93	1.63	2.25
Kadapa et al. [52]	2.15	-	1.60	-
Present scheme - Mesh A	2.14	0.93	1.59	2.23
Present scheme - Mesh B	2.15	0.93	1.60	2.23

Table 2: Flow past a cylinder: drag coefficient (C_D) and length of recirculation zone (L_r) for $Re = 20$ and $Re = 40$.

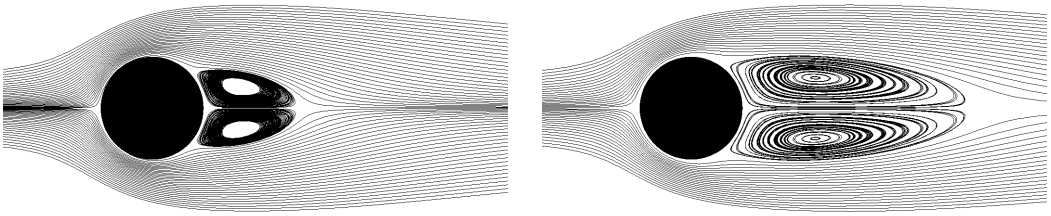


Figure 22: Flow past a cylinder: Streamlines with mesh A for $Re = 20$ (left) and $Re = 40$ (right).

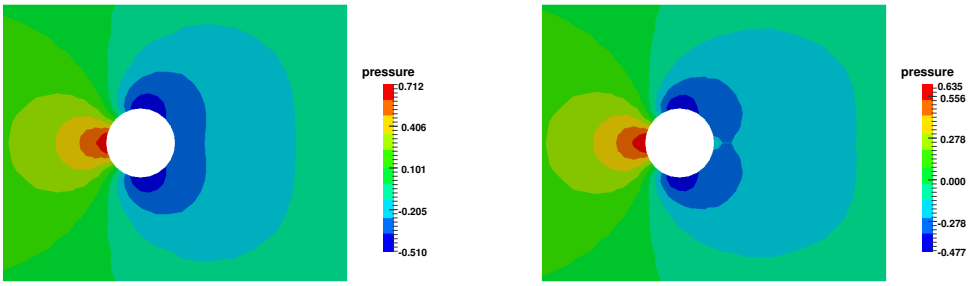


Figure 23: Flow past a cylinder: contour plot of pressure with mesh A for $Re = 20$ (left) and $Re = 40$ (right).

5. Summary and conclusions

We have presented a novel finite element scheme for fully explicit solutions of incompressible Navier-Stokes using Bézier elements and a single-step explicit time integration scheme. We have demonstrated the accuracy of the proposed scheme using the benchmark examples and proved that the proposed scheme is computationally very efficient and is applicable to real-world problems made of complex geometries.

The important characteristics of the present work can be summarised as:

- The scheme is robust and computationally efficient: robustness of the scheme is due to the inf-sup stable velocity pressure combinations employed and the computational efficiency of the scheme results from the combined effect of its single-step nature, the use of second-order spatial discretisation for the velocity field and the absence of additional stabilisations usually employed to circumvent inf-sup stability criterion.

- Finite element discretisations for the proposed work can be generated by applying the simple mapping technique over the existing mesh generators. Moreover, since the standard elimination approach is used for applying Dirichlet boundary conditions, the proposed scheme can be implemented into existing finite element codes with fewer resources.

- The scheme does not require any additional techniques or data structures for the treatment of boundary conditions.

- The simplistic nature of the proposed scheme renders it highly suitable for exploiting the full potential of modern high-performance computing architectures.

In conclusion, the potential of the proposed scheme to compute accurate numerical results using fewer computational resources and its ability to exploit the existing mesh generators for generating finite element meshes, make the proposed scheme a cost-effective numerical framework for the simulation of laminar incompressible Navier-Stokes.

The proposed framework offers numerous possibilities for developing computationally efficient explicit, implicit and semi-explicit/semi-implicit finite element schemes for compressible and incompressible flow problems. The ongoing work is focussed on extending the present work to three-dimensional problems as well as fluid-structure interaction problems.

Acknowledgements

We acknowledge the support of the Supercomputing Wales project, which is part-funded by the European Regional Development Fund (ERDF) via the Welsh Government.

[1] Anderson Jr. J. D.. *Computational Fluid Dynamics - The basics with applications*. United States of America: Mc Graw-Hill Book Company; 1995.

[2] Blazek J.. *Computational Fluid Dynamics*. Great Britain: Elsevier; 2005.

[3] Chung T. J.. *Computational Fluid Dynamics*. United Kingdom: Cambridge University Press; 2002.

[4] Connor J. J., Brebbia C. A.. *Finite Element Techniques for Fluid Flow*. Newnes-Butterworths; 1976.

[5] Deville M. O., Fischer P. F., Mund E. H.. *High-order Methods for Incompressible Fluid Flow*. United Kingdom: Cambridge University Press; 2002.

[6] Versteeg H., Malalasekera W.. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson; 2007.

[7] Donea J., Huerta A.. *Finite Element Methods for Flow Problems*. Chichester, UK: John Wiley and Sons; 2003.

[8] Elman H. C., Silvester D. J., Wathen A. J.. *Finite Element Methods and Fast Iterative Solvers: with applications in incompressible fluid dynamics*. United States of America: Oxford University Press; 2005.

[9] Ferziger J. H., Perić M.. *Computational Methods for Fluid Dynamics*. Germany: Springer; third ed.2002.

[10] Pletcher R. H., Tannehill J. C., Anderson D. A.. *Computational Fluid Mechanics and Heat Transfer*. United States of America: CRC Press; third ed.2013.

[11] Zienkiewicz O. C., Taylor R. L., Nithiarasu P.. *The Finite Element Method for Fluid Dynamics*. United States: Butterworth and Heinemann; Seventh ed.2014.

[12] Kuzmin D., Hämäläinen J.. *Finite Element Methods for Computational Fluid Dynamics: A Practical Guide*. SIAM; 2015.

[13] Gresho P., Sani R. L.. *Incompressible Flow and the Finite Element Method, Volume 1: Advection-Diffusion and Isothermal Laminar Flow*. Wiley; 2000.

[14] Gresho P. M., Sani R. L.. *Incompressible Flow and the Finite Element Method, Volume 2: Isothermal Laminar Flow*. Chichester, UK: Wiley; 2000.

[15] Bazilevs Y., Takizawa K., Tezduyar T. E.. *Computational Fluid-Structure Interaction: Methods and Applications*. Wiley; 2013.

[16] Sigrist J.. *Fluid-Structure Interaction: An Introduction to Finite Element Coupling*. Wiley; 2015.

[17] Cottrell J. A., Bazilevs Y., Hughes T. J. R.. *Isogeometric analysis: toward integration of CAD and FEA*. Chichester, England: John Wiley & Sons; 2009.

[18] B. Timothy J., Herman D., eds.*High-Order Methods for Computational Physics*. Germany: Springer; 1999.

[19] Brezzi F., Fortin M.. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag; 1991.

[20] Zienkiewicz O. C., Codina R.. A general algorithm for compressible and incompressible flow, Part I. The split characteristic based scheme. *International Journal for Numerical Methods in Fluids*. 1995;20:869-885.

[21] Zienkiewicz O. C., Satya Sai B. V. K., Morgan K., Codina R., Vázquez M.. A general algorithm for compressible and incompressible flow, Part II. Tests on the explicit form. *International Journal for Numerical Methods in Fluids*. 1995;20:887-913.

[22] Massarotti N., Nithiarasu P., Zienkiewicz O. C.. Characteristic-based-split (CBS) algorithm for incompressible flow problems with heat transfer. *International Journal for Numerical Methods for Heat & Fluid Flow*. 1998;8(8):969-990.

[23] Zienkiewicz P.. The characteristic-based-split (CBS) algorithm, stability and boundary conditions. *Archives of Mechanics*. 2000;52(4-5):857-887.

[24] Nithiarasu P.. An efficient artificial compressibility (AC) scheme based on the characteristic based split (CBS) method for incompressible flows. *International Journal for Numerical Methods in Engineering*. 2003;.

[25] Nithiarasu P.. A fully explicit characteristic based split (CBS) scheme for viscoelastic flow calculations. *International Journal for Numerical Methods in Engineering*. 2004;60:949-978.

[26] Bevan R. L. T., Boileau E., Loon R. , Lewis R. W., Nithiarasu P.. A comparative study of the fractional step method in quasi-implicit, semi-implicit and fully-explicit forms for incompressible flows. *International Journal for Numerical Methods for Heat & Fluid Flow*. 2016;26:595-623.

[27] Rüberg T., Cirak F.. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Computer Methods in Applied Mechanics and Engineering*. 2012;209-212:266-283.

[28] Rüberg T., Cirak F.. A fixed-grid b-spline finite element technique for fluid-structure interaction. *International Journal for Numerical Methods in Fluids*. 2014;74:623-660.

[29] Kadapa C., Dettmer W. G., Perić D.. Subdivision based mixed methods for isogeometric analysis of linear and nonlinear nearly incompressible materials. *Computer Methods in Applied Mechanics and Engineering*. 2016;305:241-270.

[30] Jaxon N., Qian X.. Isogeometric analysis on triangulations. *Computer-Aided Design*. 2014;46:45-57.

[31] Engvall L., Evans J. A.. Isogeometric triangular Bernstein-Bézier discretizations: Automatic mesh generation and geometrically exact finite element analysis. *Computer Methods in Applied Mechanics and Engineering.* 2016;304:378-407.
[32] Xia S., Qian X.. Isogeometric analysis with Bézier tetrahedra. *Computer Methods in Applied Mechanics and Engineering.* 2017;316:782-816.
[33] Kanduč C., Pelosi F., Speleers H.. Adaptive isogeometric analysis with hierarchical box splines. *Computer Methods in Applied Mechanics and Engineering.* 2017;316:817-838.
[34] Kadapa C.. Novel quadratic Bézier triangular and tetrahedral elements using existing mesh generators: Applications to linear nearly incompressible elastostatics and implicit and explicit elastodynamics. *International Journal for Numerical Methods in Engineering.* 2019;117:543-573.
[35] Jansen K. E., Whiting C. H., Hulbert G. M.. A generalized-α method for integrating filtered Navier-Stokes equations with a stabilized finite element method. *Computer Methods in Applied Mechanics and Engineering.* 2000;190:305-319.
[36] Chorin A. J.. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics.* 1967;2(1):12-26.
[37] Glowinski R.. Finite element methods for incompressible viscous flow. In: Ciarlet P. G., Lions J. L., eds. *Numerical Methods for Fluids (Part 3)*, Handbook of Numerical Analysis, vol. 9: Elsevier 2003 (pp. 3–1176).
[38] Rahman M. M., Siikonen T.. An artificial compressibility method for incompressible flows. *Numerical Heat Transfer, Part B: Fundamentals.* 2001;40(5):391-409.
[39] Louda P., Kozel K., Příhoda J.. Numerical solution of 2D and 3D viscous incompressible steady and unsteady flows using artificial compressibility method. *International Journal for Numerical Methods in Fluids.* 2008;56(8):1399-1407.
[40] Liang C., Chan A., Liu X., Jameson A.. An Artificial Compressibility Method for the Spectral Difference Solution of Unsteady Incompressible Navier-Stokes Equations on Multiple Grids. In: ; 2011.
[41] Ohwada T., Asinari P.. Artificial compressibility method revisited: Asymptotic numerical method for incompressible NavierStokes equations. *Journal of Computational Physics.* 2010;229(5):1698-1723.
[42] Ohwada T., Asinari P., Yabusaki D.. Artificial compressibility method and lattice Boltzmann method: Similarities and differences. *Computers and Mathematics with Applications.* 2011;61:3461-3474.
[43] Piegl L., Tiller W.. *The NURBS Book (Monographs in Visual Communication)*. New York: Springer-Verlag; 1997.
[44] Chapelle D., Bathe K. J.. The inf-sup test. *Computers and Structures.* 1993;47:537-545.
[45] Kovasznay L. I. G.. Laminar flow behind a two-dimensional grid. *Mathematical Proceedings of the Cambridge Philosophical Society.* 1948;44(1):58-62.
[46] Kozel K., Louda P., Prihoda J.. Numerical solution of an unsteady flow using artificial compressibility method. In: ; 2006.
[47] Ghia U., Ghia K. N., Shin C. T.. High-resolution for incompressible flow using the Navier-Stokes equations and the multigrid method. *Journal of Computational Physics.* 1982;48:387-411.
[48] Dennis S. C. R., Chang G.. Numerical solutions for steady flow past a circular cylinder at Reynolds number up to 100. *Journal of Fluid Mechanics.* 1970;42:471-489.
[49] Fornberg B.. A numerical study of steady viscous flow past of a circular cylinder. *Journal of Fluid Mechanics.* 1980;98:819-855.
[50] Linnick M. N., Fasel H. F.. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *Journal of Computational Physics.* 2005;204:157-192.
[51] Russell D., Wang Z. J.. A cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow. *Journal of Computational Physics.* 2003;191:177-205.
[52] Kadapa C., Dettmer W. G., Perić D.. A fictitious domain/distributed Lagrange multiplier based fluid-structure interaction scheme with hierarchical B-Spline grids. *Computer Methods in Applied Mechanics and Engineering.* 2016;301:1-27.
[53] Liu C., Sheng X., Sung C. H.. Preconditioned multigrid methods for unsteady incompressible flows. *Journal of Computational Physics.* 1998;139:35-57.
[54] Calhoun D.. A cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions. *Journal of Computational Physics.* 2002;176:231-275.
[55] Le D. V., Khoo B. C., Péraire J.. An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries. *Journal of Computational Physics.* 2006;220:109-138.
[56] Kadapa C., Dettmer W. G., Perić D.. A stabilised immersed boundary method on hierarchical b-spline grids for fluid-rigid body interaction with solid-solid contact. *Computer Methods in Applied Mechanics and Engineering.* 2017;318:242-269.