Journal of Computational Physics 228 (2009) 7250–7267

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Open and traction boundary conditions for the incompressible Navier–Stokes equations

Jie Liu *

Department of Mathematics, National University of Singapore, 117543, Singapore

ARTICLE INFO

Article history:
Received 22 October 2008
Received in revised form 13 March 2009
Accepted 19 June 2009
Available online 24 June 2009

Keywords:
Incompressible flow
Traction boundary condition
Open boundary condition
Pressure Poisson formulation
Pressure boundary condition
Flow past a cylinder
Bifurcated tube

ABSTRACT

We present numerical schemes for the incompressible Navier–Stokes equations (NSE) with open and traction boundary conditions. We use pressure Poisson equation (PPE) formulation and propose new boundary conditions for the pressure on the open or traction boundaries. After replacing the divergence free constraint by this pressure Poisson equation, we obtain an unconstrained NSE. For Stokes equation with open boundary condition on a simple domain, we prove unconditional stability of a first order semi-implicit scheme where the pressure is treated explicitly and hence is decoupled from the computation of velocity. Using either boundary condition, the schemes for the full NSE that treat both convection and pressure terms explicitly work well with various spatial discretizations including spectral collocation and C^0 finite elements. Moreover, when Reynolds number is of $O(1)$ and when the first order semi-implicit time stepping is used, time step size of $O(1)$ is allowed in benchmark computations for the full NSE. Besides standard stability and accuracy check, various numerical results including flow over a backward facing step, flow past a cylinder and flow in a bifurcated tube are reported. Numerically we have observed that using PPE formulation enables us to use the velocity/pressure pairs that do not satisfy the standard inf–sup compatibility condition. Our results extend that of Johnston and Liu [H. Johnston, J.-G. Liu, Accurate, stable and efficient Navier–Stokes solvers based on explicit treatment of the pressure term. J. Comp. Phys. 199 (1) (2004) 221–259] which deals with no-slip boundary conditions only.

1. Introduction

Consider the Navier–Stokes equations (NSE) for incompressible fluid flow in $\Omega \subset \mathbb{R}^m (m \geqslant 2)$

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \nu \Delta \boldsymbol{u} + \boldsymbol{f} \quad \text{in } \Omega, \tag{1}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega. \tag{2}$$

Here $\boldsymbol{u}$ is the fluid velocity, p the pressure, and $\nu = 1/\mathrm{Re}$ is the kinematic viscosity coefficient. Assume $\partial\Omega = \Gamma_1 \cup \Gamma_2$. Suppose we are given velocity $\boldsymbol{u}$ on Γ_1 and are given pseudo-traction $\nu \partial_n \boldsymbol{u} - p\boldsymbol{n}$ on Γ_2, namely,

$$\boldsymbol{u} = \boldsymbol{g}_1 \quad \text{on } \Gamma_1, \tag{3}$$

$$\nu \partial_n \boldsymbol{u} - p\,\boldsymbol{n} = \boldsymbol{g}_2 \quad \text{on } \Gamma_2, \tag{4}$$

* Tel.: +65 65162798.
E-mail address: matlj@nus.edu.sg

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.06.021

where $\partial_n \boldsymbol{u} = (\boldsymbol{n}\cdot\nabla)\boldsymbol{u} = (\nabla\boldsymbol{u})\boldsymbol{n} = n_j\partial_j u_i$ and $\boldsymbol{n}$ is the outward normal. In this paper, we call (4) the open boundary condition. In some situation, instead of (4), one might be given the force acting on Γ_2 (equals to stress times the normal vector of Γ_2) which we call the traction boundary condition:

$$\nu(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top)\boldsymbol{n} - p\,\boldsymbol{n} = \boldsymbol{g}_2 \quad \text{on } \Gamma_2. \tag{5}$$

In finite element computations, to incorporate the traction boundary condition (5) into the weak formulation, we will add back the $\nu\nabla\nabla\cdot\boldsymbol{u}$ to the right hand side of (1). So, in this paper, when talking about the traction boundary condition (5), we will use the following momentum equation

$$\partial_t\boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} + \nabla p = \nu\nabla\cdot(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top) + \boldsymbol{f} \quad \text{in } \Omega. \tag{6}$$

Open boundary condition (4) is used to truncate a big physical domain to make the problem tractable [39]. In some situations, when the outflow profile cannot be determined a priori (e.g. thinking about how to prescribe the outflow profile for flow in a bifurcated tube), one can use (4) as the outflow boundary condition (Fig. 1). Traction boundary condition (5) can also be used for the same purposes. Moreover (5) is related to the free boundary problem [13] and the problem of fluid–structure interaction [12,35]. For the well-posedness of NSE with these two types of boundary conditions, we refer to [18] and the references therein.

1.1. Pressure Poisson equation formulation and pressure boundary conditions on Γ_1

Many of the difficulties in incompressible fluid computations are related to how to enforce the incompressibility constraint (2). One approach is to reformulate the original NSE into an equivalent form which replaces $\nabla\cdot\boldsymbol{u} = 0$ by a pressure Poisson equation (PPE). The PPE formulations with various right hand sides of the Poisson equation and various boundary conditions for the pressure have been widely used [34,17,40]. When $\Gamma_2 = \emptyset$, i.e. when velocity is given on the whole boundary, [34] proposed the following Poisson equation with Neumann type boundary conditions for the pressure:

$$\Delta p = \nabla\cdot(\boldsymbol{f} - \boldsymbol{u}\cdot\nabla\boldsymbol{u}) \quad \text{in } \Omega, \tag{7}$$

$$\boldsymbol{n}\cdot\nabla p = \boldsymbol{n}\cdot(\boldsymbol{f} - \boldsymbol{u}\cdot\nabla\boldsymbol{u} - \partial_t\boldsymbol{g}_1) - \nu\boldsymbol{n}\cdot\nabla\times\nabla\times\boldsymbol{u} \quad \text{on } \Gamma_1. \tag{8}$$

In this situation ($\Gamma_2 = \emptyset$), one can prove that the pressure gradient is strictly dominated by the viscosity term modulo lower order terms (see [31] for details). Using this pressure estimate, one can directly prove the well-posedness of the PPE formulation of NSE and establish the unconditional stability of a semi-implicit scheme that decouples of the computation of velocity and pressure. In this paper, semi-implicit means explicit treatment of the pressure term and implicit treatment of the viscosity term. By this semi-implicit decoupling, we gain efficiency, but we do not sacrifice stability because of the pressure estimates obtained from (7) and (8) [27,31–33]. Finally, it is easy to see that consistency can be readily achieved using PPE formulation.

Given the usefulness of this pressure estimate, however, one unpleasant fact is that this estimate requires the velocity to be known on the whole boundary. When $\Gamma_2 \neq \emptyset$ and $\boldsymbol{u}$ satisfies the boundary conditions (4) or (5) on Γ_2, if we keep using (8) on $\Gamma_1 \cup \Gamma_2$, we can no longer apply the aforementioned pressure estimate to analyze the resulting numerical schemes because $\boldsymbol{u}$ on Γ_2 becomes unknown. Moreover, the p solved from (7) and (8) will still be a solution if we add an arbitrary constant α to it. On the other hand, we know we cannot add any α to the p in (4) or (5). So, if the momentum Eq. (1) or (6) are supplemented with boundary conditions (4) or (5), using (7) and (8) to determine p becomes problematic because we do not know how to fix this constant α.

1.2. Pressure boundary condition on Γ_2, pressure estimates and unconditional stability of semi-implicit schemes

Having mentioned the limitation of the boundary condition (8) on Γ_2, we now propose new boundary conditions of p on Γ_2 so that we can still have some pressure estimates and can handle the constant α mentioned before when $\Gamma_2 \neq \emptyset$. The resulting pressure estimates eventually enable us to prove the unconditional stability of a semi-implicit scheme for the Stokes equation. The pressure boundary conditions we proposed are as follows: If on Γ_2, we are given the open boundary condition (4), then

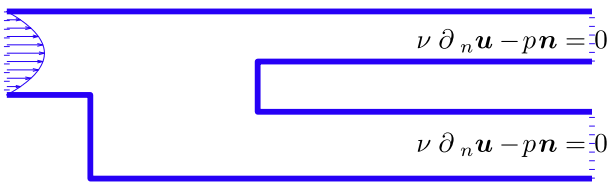


Fig. 1. Boundary condition for flow in a bifurcated tube with given inflow profile.

$$p = \nu \boldsymbol{n} \cdot (\partial_n \boldsymbol{u}) - \nu \nabla \cdot \boldsymbol{u} - \boldsymbol{n} \cdot \boldsymbol{g}_2 \quad \text{on } \Gamma_2. \tag{9}$$

If on Γ_2, we are given the traction boundary condition (5), then

$$p = \nu \boldsymbol{n}^\top (\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\top) \boldsymbol{n} - \nu \nabla \cdot \boldsymbol{u} - \boldsymbol{n} \cdot \boldsymbol{g}_2 \quad \text{on } \Gamma_2. \tag{10}$$

In fact, the $\nu \nabla \cdot \boldsymbol{u}$ in (10) can be $\gamma \nu \nabla \cdot \boldsymbol{u}$ for any $\gamma \in [1, 2]$ in most numerical tests. We will simply take $\gamma = 1$ in our following discussion. It is clear that with these Dirichlet type boundary conditions for p, there is no room for the arbitrary constant α.

1.3. Connections with [20] and Uzawa algorithm

Numerical methods for NSE with open and traction boundary conditions have been studied by [20] which has proved error estimates for the following semi-discrete scheme

$$\begin{cases} \frac{3\tilde{\boldsymbol{u}}^{n+1} - 4\boldsymbol{u}^n + \boldsymbol{u}^{n-1}}{2\Delta t} - \nu \Delta \tilde{\boldsymbol{u}}^{n+1} + \nabla p^n = \boldsymbol{f}^{n+1} \\ \tilde{\boldsymbol{u}}^{n+1}|_{\Gamma_1} = 0, \quad (p^n \boldsymbol{n} - \nu \partial_n \tilde{\boldsymbol{u}}^{n+1})|_{\Gamma_2} = 0 \end{cases} \tag{11}$$

$$\begin{cases} \frac{3\boldsymbol{u}^{n+1} - 3\tilde{\boldsymbol{u}}^{n+1}}{2\Delta t} + \nabla \phi^{n+1} = 0 \\ \nabla \cdot \boldsymbol{u}^{n+1} = 0 \\ \boldsymbol{u}^{n+1} \cdot \boldsymbol{n}|_{\Gamma_2} = 0, \quad \phi^{n+1}|_{\Gamma_2} = 0 \end{cases} \tag{12}$$

$$\phi^{n+1} = p^{n+1} - p^n + \chi \nabla \cdot \tilde{\boldsymbol{u}}^{n+1} \tag{13}$$

where $\chi \in (0, \frac{2}{m}\nu)$ is constant and m is the space dimension. Traction boundary condition (5) can be handled similarly. To our surprise, if we restrict (13) on Γ_2 and use the boundary conditions in (11) and (12), we find

$$p^{n+1} = \nu \boldsymbol{n} \cdot (\partial_n \tilde{\boldsymbol{u}}^{n+1}) - \chi \nabla \cdot \tilde{\boldsymbol{u}}^{n+1} \quad \text{on } \Gamma_2 \tag{14}$$

which is almost exactly (9) except we take $\chi = \nu$. (But the analysis in [20] does not apply to $\chi = \nu$.) In fact, we will obtain similar boundary conditions of p (with a different χ) if we apply Uzawa algorithm [43] to solve NSE with open or traction boundary conditions. Note that the p^{n+1} in (11)–(13) will satisfy some Poisson equation that is slightly different from (7).

1.4. Unconstrained NSE

After mentioning the above connections, we would like to point out the main difference between our current approach and those in [20] or in Uzawa algorithm: By (7)–(9), we are actually saying that *the pressure at time t is completely determined from the velocity at time t without any time lag*, which can be written as $p(t) = P(\boldsymbol{u}(t))$ (after ignoring the dependence on $\{\boldsymbol{f}, \boldsymbol{g}_1, \boldsymbol{g}_2\}$ for simplicity). Then the NSE (1)–(4) can be written as

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla P(\boldsymbol{u}) = \nu \Delta \boldsymbol{u} + \boldsymbol{f} \quad \text{in } \Omega \tag{15}$$

with boundary condition $\boldsymbol{u} = \boldsymbol{g}_1$ on Γ_1 and $\nu \partial_n \boldsymbol{u} - P(\boldsymbol{u})\boldsymbol{n} = \boldsymbol{g}_2$ on Γ_2 (See Proposition 1). By considering NSE (15) as simply an ODE of $\boldsymbol{u}$ in function space without any constraint, we gain more freedom to design efficient and stable time steppings. In fact, in some sense we can consider (15) as a perturbed heat equation. Problems like designing spectrally accurate numerical schemes [25] and designing efficient time steppings for fluid elastic structure interactions can be re-considered from these new point of views.

1.5. Inf–sup conditions

Another difference between [20] and our paper is that finite element schemes based on [20] ((11)–(13)) require the inf–sup compatibility condition between X_h and Y_h which are the finite element spaces for velocity and pressure:

$$\inf_{q_h \in Y_h} \sup_{\boldsymbol{v}_h \in X_h} \frac{\langle \nabla \cdot \boldsymbol{v}_h, q_h \rangle}{\|\nabla \boldsymbol{v}_h\| \|q_h\|} \geqslant c > 0 \tag{16}$$

where c is a constant independent of h [5,9]. On the other hand, our schemes *might* be able to avoid this inf–sup condition. The following discussion also serves to clarify the differences between [20] and our approach in the fully discrete case.

For finite element discretization, if the pressure finite element space Y_h is too big with respect to the velocity finite element space X_h, there could be some $\check{p}_h \in Y_h, \check{p}_h \neq 0$ such that

$$\int_\Omega (\nabla \cdot \boldsymbol{v}_h) \check{p}_h = 0 \quad \text{for all} \quad \boldsymbol{v}_h \in X_h. \tag{17}$$

These $\check{p}_h$'s are called spurious pressure modes. If using C^0 finite elements, because the $\nabla \cdot \tilde{\boldsymbol{u}}_h^{n+1}$ in (13) is discontinuous, it can contain some small $\check{p}_h$'s which can be passed to p_h^{n+1} by (13). Moreover, any $\check{p}_h$ contained in p_h^n will also be passed to p_h^{n+1} by (13) since it is $p_h^{n+1} - p_h^n$ that matters. Certainly there is no mechanism to kick those $\check{p}_h$'s out of (11)–(13) and there is no mechanism to prevent them from accumulating and growing. Intuitively, the discontinuous $\nabla \cdot \tilde{\boldsymbol{u}}_h^{n+1}$ could be a constant

source to generate those spurious pressure modes. Even for spectral finite elements, there is still no mechanism to kill those spurious modes and the system is still vulnerable. Numerical examples [20] and theoretical analysis [24] indicate that finite element schemes based on (11)–(13) require the velocity/pressure pairs to satisfy (16) which among many things exclude the existence of spurious pressure modes at all.

Unlike [20] which uses p^n to compute p^{n+1} hence leaving room for the *preservation and accumulation* of spurious pressure modes, our p^{n+1} is computed from a Poisson equation using velocity only. One might say that the standard projection method also determines p from velocity only [11,42]. However, our formulation does not contain the trouble-making $\frac{1}{\Delta t}$ factor in the Poisson equation for p [21]. So far, what we can say is the following: In all the computations including stability and accuracy check, backward facing step flow, flow past a cylinder and flow in a bifurcated tube, we are able to use Lagrange finite elements of equal order for both velocity and pressure which are known to not satisfy the inf–sup condition [14]; Following [21], we also tried to integrate with a very small time step on a relatively coarse grid using P1/P1 finite elements, but still did not observe any spurious pressure mode. When $\Gamma_2 = \emptyset$ and when using C^1 finite elements for velocity, stability and error estimates of a first order semi-implicit scheme are proved in [32] where incompatible velocity/pressure pairs can be used. Numerically, this has also been observed for C^0 finite element schemes in [27,33] with no-slip boundary conditions only. The advantage of getting rid of the inf–sup condition is that anyone can then very easily turn his favorite heat equation solver into a NSE solver.

We get this advantage on inf–sup condition as a by-product of the unconstrained formulation (15). If just looking at the inf–sup condition, a more standard approach for using equally ordered finite element pairs is to use the stabilized finite element methods (see [7] and the references therein). There are also studies of inf–sup condition in other problems. In computational elasticity, if stress–displacement formulation is used, Brezzi's stability condition [9] is required [4]. But using purely displacement formulation can avoid the Brezzi's condition completely [8]. Similar things happen to the Poisson equation [3]. What we have learned through these examples is that changing the formulation may give us a chance to avoid some complicated conditions in computation. This is probably the situation we are facing right now.

There are tons of the literatures related to numerical method for NSE, even though many are restricted to no-slip boundary conditions only. In the following, we will only mention those which are most relevant to our current work: Our paper basically can be viewed as a following up paper of [34,27] which deals with no-slip boundary conditions only. Same observations related to inf–sup condition have been made in [27,31–33] where a fully discrete scheme has been rigorously studied in [31,32]. Schemes similar to (11)–(13) that requires inf–sup condition have been studied in [44,19,20,22–24]. Error estimates of (11)–(13) are proved in [20]. Some earlier work on open boundary conditions can be found in [39]. Other related important literatures include [42,11,28,30,29,37] and the references in [19,31].

The rest of the paper is organized as follows: In Section 2, we introduce the pressure boundary condition for PPE on the open or traction boundaries. Then we prove the equivalence between PPE formulation and (1)–(4) or {(6), (2), (3), (5)}. In Section 3, we prove the unconditional stability of the first order semi-implicit scheme for the Stokes equation. Higher order schemes are presented in Setion 4 and how to *further* enforce $\nabla \cdot \boldsymbol{u} = 0$ is discussed in Section 5. C^0 finite element discretization is discussed in Section 6 followed by various numerical tests in Section 7.

2. Pressure boundary condition

In this section, we will propose the pressure boundary conditions used in PPE formulation when there are open and traction boundaries.

2.1. Pressure boundary condition on open boundaries

As we have mentioned in the introduction, in the open boundary condition case, we propose to replace (2), the incompressibility constraint with the following Poisson equation for pressure which combining with the momentum equation will enforce $\nabla \cdot \boldsymbol{u} = 0$ automatically:

$$\Delta p = \nabla \cdot (\boldsymbol{f} - \boldsymbol{u} \cdot \nabla \boldsymbol{u}) \quad \text{in } \Omega \tag{18}$$

$$\partial_n p = \boldsymbol{n} \cdot (\boldsymbol{f} - \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \partial_t \boldsymbol{g}_1) - \nu \boldsymbol{n} \cdot \nabla \times \nabla \times \boldsymbol{u} \quad \text{on } \Gamma_1 \tag{19}$$

$$p = \nu \boldsymbol{n} \cdot (\partial_n \boldsymbol{u}) - \nu \nabla \cdot \boldsymbol{u} - \boldsymbol{n} \cdot \boldsymbol{g}_2 \quad \text{on } \Gamma_2. \tag{20}$$

We have the following equivalency results of the two formulations:

Proposition 1. *Assuming enough regularity of the solution and the data* $\{\boldsymbol{f}, \boldsymbol{g}_1, \boldsymbol{g}_2\}$, *if the initial velocity is divergence free, then* {(1), (3), (4), (18)–(20)} *is equivalent to* (1)–(4).

Proof. Let us first prove {(1), (3), (4), (18)–(20)} implies (1)–(4): Dotting both sides of (1) by $\boldsymbol{n}$, restricting the result on Γ_1 and then subtracting (19) from it, we obtain $\nu \boldsymbol{n} \cdot \Delta \boldsymbol{u} = -\nu \boldsymbol{n} \cdot \nabla \times \nabla \times \boldsymbol{u}$ on Γ_1. Hence

$$\partial_n (\nabla \cdot \boldsymbol{u}) = 0 \quad \text{on } \Gamma_1$$

because $\Delta \boldsymbol{u} = \nabla\nabla\cdot\boldsymbol{u} - \nabla\times\nabla\times\boldsymbol{u}$. Dotting both sides of (4) by n, we get

$$p = \nu \boldsymbol{n}\cdot(\partial_n \boldsymbol{u}) - \boldsymbol{n}\cdot\boldsymbol{g}_2 \quad \text{on } \Gamma_2.$$

Subtracting (20) from the above result, we get

$$\nabla\cdot\boldsymbol{u} = 0 \quad \text{on } \Gamma_2$$

Finally, taking divergence of (1) and subtracting (18) from it, we get

$$\partial_t(\nabla\cdot\boldsymbol{u}) = \nu\Delta(\nabla\cdot\boldsymbol{u}) \quad \text{in } \Omega. \tag{21}$$

Together with the zero Neumann condition on Γ_1 and zero Dirichlet condition on Γ_2 (21) implies $\nabla\cdot\boldsymbol{u} = 0$ in Ω.

The proof of other direction is similar, which we omit. □

Remark 1. We mention in passing that we can change the extra $-\nu\nabla\cdot\boldsymbol{u}$ in (20) to anything like $-20\nu\nabla\cdot\boldsymbol{u}$ or $+\nu\nabla\cdot\boldsymbol{u}$ and will still obtain Proposition 1. The reason why we choose this specific one is because of the stability results we can then obtained (Proposition 3).

Remark 2. We did not precisely state what the regularity requirements for the solution and data in Proposition 1 are. Obviously the regularity we need can be determined from the requirement that all the manipulations in the proof are legal. But in fact much less regularity is required for which we refer the interested readers to the more rigorous treatment in [18]. Same remarks apply to Proposition 2.

2.2. Pressure boundary condition on traction boundaries

We have the equivalency results similar to Proposition 1 with a very similar proof which is then omitted.

Proposition 2. *Assuming enough regularity of the solution and the data* $\{\boldsymbol{f}, \boldsymbol{g}_1, \boldsymbol{g}_2\}$, *if the initial velocity is divergence free, then* {(6), (3), (5), (18), (19), (10)} *is equivalent to* {(6), (3), (5), (2)}.

3. Unconditional stability of a semi-implicit scheme for Stokes equation with open boundaries

In this section, we prove the unconditional stability of a first order semi-implicit scheme for Stokes equation with open boundary conditions. So far, we only have the proof when we keep the spatial variables continuous. The analysis is an extension of that in [27] for the no-slip boundary condition case.

Let us consider the Stokes equation in a periodic strip $\Omega = [-1,1]\times(0,2\pi)$ where we have periodic boundary condition in y direction, no-slip condition on $\{x=-1\}$ and open boundary condition with $\boldsymbol{g}_2 = 0$ on $\{x=1\}$. Since it is a linear equation, we can take $\nu = 1$, and the results apply to any ν. The PPE formulation {(1), (3), (4), (18)–(20)} is as follows ($\boldsymbol{u} = (u, v)$):

$$\partial_t \boldsymbol{u} + \nabla p = \Delta \boldsymbol{u} \quad \text{in } \Omega \tag{22}$$

$$u = 0, \quad v = 0 \quad \text{on } \{x=-1\} \tag{23}$$

$$\partial_x u - p = 0, \quad \partial_x v = 0 \quad \text{on } \{x=1\} \tag{24}$$

$$\Delta p = 0 \quad \text{in } \Omega \tag{25}$$

$$\partial_x p = \partial_y^2 u - \partial_x\partial_y v, \quad \text{on } \{x=-1\} \tag{26}$$

$$p = -\partial_y v \quad \text{on } \{x=1\}. \tag{27}$$

Since $u = 0$ on $\{x=-1\}$ (26) reduces to

$$\partial_x(p + \partial_y v) = 0 \quad \text{on } \{x=-1\}. \tag{28}$$

Together with (27), we see that $p + \partial_y v$ satisfies

$$\Delta(p+\partial_y v) = \Delta\partial_y v, \quad \partial_x(p+\partial_y v)|_{x=-1} = 0, \quad p + \partial_y v|_{x=1} = 0. \tag{29}$$

Since we have Neumann boundary condition on the left boundary and Dirichlet boundary condition on the right, we define the solution operator of the above equation by Δ_{ND}^{-1}. Then we have

$$p = \left(\Delta_{ND}^{-1}\Delta - I\right)\partial_y v. \tag{30}$$

Now, define the pressure operator

$$\mathcal{B} = \partial_y\left(\Delta_{ND}^{-1}\Delta - I\right)\partial_y. \tag{31}$$

Then $\mathcal{B}v = \partial_y p$ and the v-component of the momentum Eq. (22) can be written as

$$\partial_t v - \Delta v + \mathcal{B}v = 0. \tag{32}$$

Define for any $\alpha > 0$,

$$H_p = \{p \in H^{1+\alpha}(\Omega), \quad \partial_x p|_{x=-1} = 0, \quad p|_{x=1} = 0\}, \tag{33}$$

$$H_v = \{v \in H^{2+\alpha}(\Omega), \quad v|_{x=-1} = 0, \quad \partial_x v|_{x=1} = 0\}. \tag{34}$$

Notice that for any ϕ, ψ belong to H_p at the same time, or any ϕ, ψ belong to H_v at the same time, we always have

$$\langle \Delta\psi, \phi\rangle = -\langle \nabla\psi, \nabla\phi\rangle. \tag{35}$$

Lemma 1. *For any $u, v \in H_v$, we have*

$$\langle \Delta u, \mathcal{B}v\rangle = \langle \Delta v, \mathcal{B}u\rangle, \tag{36}$$

$$|\langle \Delta v, \mathcal{B}v\rangle| \leqslant \|\Delta v\|^2. \tag{37}$$

Proof. The proof is basically the same as the one in [27]. Let $\psi, \phi \in H_p \cap H^{2+\alpha}$, satisfying

$$\Delta\psi = \Delta u, \quad \Delta\phi = \Delta v. \tag{38}$$

When $\phi \in H_p \cap H^{2+\alpha}$, we have $\partial_y\phi \in H_p$ and hence $\Delta_{ND}^{-1}\Delta\partial_y\phi = \partial_y\phi$. So

$$\mathcal{B}v = \partial_y\Delta_{ND}^{-1}\Delta\partial_y v - \partial_y^2 v = \partial_y\Delta_{ND}^{-1}\Delta\partial_y\phi - \partial_y^2 v = \partial_y^2\phi - \partial_y^2 v.$$

Because we have periodic boundary condition in y direction, we can integrate by parts on y freely. So using $\Delta u = \Delta\psi$,

$$\langle \Delta u, \mathcal{B}v\rangle = \langle \Delta\psi, \partial_y^2\phi\rangle - \langle \Delta u, \partial_y^2 v\rangle = -\langle \Delta\partial_y\psi, \partial_y\phi\rangle + \langle \Delta\partial_y u, \partial_y v\rangle = \langle \nabla\partial_y\psi, \nabla\partial_y\phi\rangle - \langle \nabla\partial_y u, \nabla\partial_y v\rangle \tag{39}$$

where we have used (35) in the last step. We can interchange ψ and ϕ and then u and v in (39). This proves (36).

To prove (37), we start from (39) and obtain

$$\langle \Delta v, \mathcal{B}v\rangle \leqslant \langle \nabla\partial_y\phi, \nabla\partial_y\phi\rangle = \langle \partial_x\partial_y\phi, \partial_x\partial_y\phi\rangle + \langle \partial_y^2\phi, \partial_y^2\phi\rangle \tag{40}$$

$$-\langle \Delta v, \mathcal{B}v\rangle \leqslant \langle \nabla\partial_y v, \nabla\partial_y v\rangle = \langle \partial_x\partial_y v, \partial_x\partial_y v\rangle + \langle \partial_y^2 v, \partial_y^2 v\rangle. \tag{41}$$

So, using $\langle \partial_x^2\phi, \partial_y^2\phi\rangle = \langle \partial_x\partial_y\phi, \partial_x\partial_y\phi\rangle \geqslant 0$, we have

$$\|\Delta v\|^2 = \|\Delta\phi\|^2 = \langle \partial_x^2\phi, \partial_x^2\phi\rangle + 2\langle \partial_x\partial_y\phi, \partial_x\partial_y\phi\rangle + \langle \partial_y^2\phi, \partial_y^2\phi\rangle \geqslant \langle \Delta v, \mathcal{B}v\rangle.$$

Similarly, we can prove $\|\Delta v\|^2 \geqslant -\langle \Delta v, \mathcal{B}v\rangle$. □

Note that the v component of the velocity satisfies the boundary condition in H_v. With Lemma 1, we can prove the following stability results for the semi-implicit scheme which treats the pressure term explicitly. The proof is essentially the same as the proof in [27] for the no-slip boundary condition case except that now our results in Lemma 1 are slightly weaker than those from [27] (see Remark 3). But it turns out to be enough.

Proposition 3. *The following semi-implicit scheme (recall that $\mathcal{B}v^n = \partial_y p^n$)*

$$\frac{v^{n+1} - v^n}{\Delta t} - \Delta v^{n+1} + \mathcal{B}v^n = 0 \tag{42}$$

satisfies

$$\|\nabla v^n\|^2 \leqslant \|\nabla v^0\|^2 + \frac{\Delta t}{2}\|\Delta v^0\|^2 + \frac{\Delta t}{2}\langle \Delta v^0, \mathcal{B}v^0\rangle, \quad \forall\, n. \tag{43}$$

Proof. The idea is to dot (42) by $-\Delta(v^{n+1} + v^n)$. Thanks to the boundary condition of v in (23) and (24), we can integrate by parts and obtain

$$\frac{\|\nabla v^{n+1}\|^2 - \|\nabla v^n\|^2}{\Delta t} + \frac{1}{2}(\|\Delta v^{n+1}\|^2 - \|\Delta v^n\|^2) + \frac{1}{2}\|\Delta(v^{n+1} + v^n)\|^2 = \langle \Delta(v^{n+1} + v^n), \mathcal{B}v^n\rangle. \tag{44}$$

Now using (37) and (36), we get

$$\begin{aligned}\langle \Delta(v^{n+1} + v^n), \mathcal{B}v^n\rangle &= \frac{1}{2}\langle \Delta(v^{n+1} + v^n), \mathcal{B}(v^{n+1} + v^n)\rangle - \frac{1}{2}\langle \Delta(v^{n+1} + v^n), \mathcal{B}(v^{n+1} - v^n)\rangle \\ &\leqslant \frac{1}{2}\|\Delta(v^{n+1} + v^n)\|^2 - \frac{1}{2}\big(\langle \Delta v^{n+1}, \mathcal{B}v^{n+1}\rangle - \langle \Delta v^n, \mathcal{B}v^n\rangle\big).\end{aligned}$$

So, in the end, we have

$$\frac{\|\nabla v^{n+1}\|^2 - \|\nabla v^n\|^2}{\Delta t} + \frac{\|\Delta v^{n+1}\|^2 - \|\Delta v^n\|^2}{2} \leqslant -\frac{\langle \Delta v^{n+1}, \mathcal{B}v^{n+1}\rangle - \langle \Delta v^n, \mathcal{B}v^n\rangle}{2}$$

which means

$$a^{n+1} \leqslant a^n, \quad \text{with } a^n = \|\nabla v^n\|^2 + \frac{\Delta t}{2}\|\Delta v^n\|^2 + \frac{\Delta t}{2}\langle \Delta v^n, \mathcal{B} v^n\rangle \geqslant \|\nabla v^n\|^2. \quad \square$$

Remark 3. As we have mentioned many times, the above results generalize those in [27] where they use no-slip boundary conditions on both $x = -1$ and $x = 1$ boundaries and enjoy an extra property $\langle \Delta v, \mathcal{B} v\rangle \geqslant 0$. We can also obtain the same result as in Proposition 3 when there are open boundary conditions on both $x = -1$ and $x = 1$ boundaries. And then we will have an extra property $\langle \Delta v, \mathcal{B} v\rangle \leqslant 0$, after some simple modifications like $\mathcal{B} = \partial_y\left(\Delta_{DD}^{-1}\Delta - I\right)\partial_y, H_p = \{p \in H^1(\Omega), p|_{x=-1} = p|_{x=1} = 0\}$ and $H_v = \{v \in H^2(\Omega), \partial_x v|_{x=-1} = \partial_x v|_{x=1} = 0\}$.

Remark 4. There are only small differences between the traction boundary condition (5) and the open boundary condition (4). Unfortunately, the results of Lemma 1 and Proposition 3 are very sensitive to these subtle differences. For example, if using (5), the $v_x = 0$ in (24) will be changed to $v_x + u_y = 0$ on $\{x = 1\}$. Then the integration by part of $\left\langle \frac{v^{n+1}-v^n}{\Delta t}, \Delta v^{n+1}\right\rangle$ will produce boundary terms. In fact, it is unclear if we should still dot Δv^{n+1} in this case. For these difficulties, at this moment, we do not have a stability proof for schemes with traction boundary conditions. But *numerically* the unconditional stability has been observed with first order semi-implicit time stepping.

4. Higher order semi-implicit schemes

Because the viscosity term can dominate the pressure term, we might have schemes with good stability property if we treat the viscosity term implicitly while treating the pressure term explicitly. Treating the pressure term explicitly allows the decoupling of the computations of velocity and pressure. For simplicity, we will also treat the convection term explicitly.

We use kth order backward differentiation formula for the viscosity term and kth order extrapolation for pressure and nonlinear terms. We will need the extrapolation formula of order k

$$\mathcal{E}_k p^{n+1} = \sum_{j=1}^{k} \beta_j^k p^{n+1-j} \tag{45}$$

where the coefficients β_j^k's are defined in Table 1.

Let us introduce the shorthand

$$\boldsymbol{h}^n = \boldsymbol{u}^n \cdot \nabla \boldsymbol{u}^n, \quad \boldsymbol{F}^{n+1} = \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}^{n+1}. \tag{46}$$

Then a kth order temporally accurate scheme for NSE with open boundary condition is as follows: given $\boldsymbol{u}^n$, we first solve for p^n from the pressure Poisson equation

$$\Delta p^n = \nabla \cdot (\boldsymbol{f}^{\,n} - \boldsymbol{h}^n) \quad \text{in } \Omega \tag{47}$$

$$\boldsymbol{n} \cdot \nabla p^n = \boldsymbol{n} \cdot (\boldsymbol{f}^{\,n} - \boldsymbol{h}^n - \partial_t \boldsymbol{g}_1(t_n)) - \nu \boldsymbol{n} \cdot \nabla \times \nabla \times \boldsymbol{u}^n \quad \text{on } \Gamma_1 \tag{48}$$

$$p^n = \nu \boldsymbol{n} \cdot (\boldsymbol{n} \cdot \nabla \boldsymbol{u}^n) - \nu \nabla \cdot \boldsymbol{u}^n - \boldsymbol{n} \cdot \boldsymbol{g}_2^n \quad \text{on } \Gamma_2. \tag{49}$$

Then we calculate $\boldsymbol{u}^{n+1}$ by the momentum equation

$$\frac{1}{\Delta t}\left(\alpha_0^k \boldsymbol{u}^{n+1} + \sum_{j=1}^{k} \alpha_j^k \boldsymbol{u}^{n+1-j}\right) + \nabla \mathcal{E}_k p^{n+1} = \nu \Delta \boldsymbol{u}^{n+1} + \boldsymbol{F}^{n+1} \quad \text{in } \Omega \tag{50}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{g}_1^{n+1} \quad \text{on } \Gamma_1 \tag{51}$$

$$\nu\, \boldsymbol{n} \cdot \nabla \boldsymbol{u}^{n+1} = \mathcal{E}_k p^{n+1}\, \boldsymbol{n} + \boldsymbol{g}_2^{n+1} \quad \text{on } \Gamma_2, \tag{52}$$

Table 1
Coefficients for backward differentiation and extrapolation.

α_j^k	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$k = 1$	1	−1	0	0	0
$k = 2$	3/2	−2	1/2	0	0
$k = 3$	11/6	−3	3/2	−1/3	0
$k = 4$	25/12	−4	3	−4/3	1/4
β_j^m	$j = 1$	$j = 2$	$j = 3$	$j = 4$	
$m = 1$	1	0	0	0	
$m = 2$	2	−1	0	0	
$m = 3$	3	−3	1	0	
$m = 4$	4	−6	4	−1	

where $\boldsymbol{F}^{n+1}$ is defined in (46) and the α_j^k's are given in Table 1. Note that both pressure and nonlinear terms are treated explicitly. The associated schemes for traction boundary condition are obvious, which we omit.

5. Divergence suppression and Leray projection

We have observed in our numerical examples that when the solution is smooth, the above schemes based on PPE formulation do a good job to enforce divergence to be close to zero. However, when the solution is less regular, divergence errors can be large at places (reentrant corners for example) where the solution is less regular. In all the benchmark computations we will present later, we have found that it is very useful to introduce further divergence suppressions, using the techniques in [37] or [32,33].

To do that, let us recall the Leray projection operator $\mathcal{P}$ which projects $\boldsymbol{u}^n$ onto the space of divergence free vector field. The result is denoted by $\mathcal{P}\boldsymbol{u}^n$. So, $\mathcal{P}\boldsymbol{u}^n = \boldsymbol{u}^n - \nabla q^n$ with q^n satisfying

$$\Delta q^n = \nabla \cdot \boldsymbol{u}^n \quad \text{in } \Omega, \quad \boldsymbol{n} \cdot \nabla q^n = 0 \quad \text{on } \partial\Omega. \tag{53}$$

Note that $\boldsymbol{n} \cdot \mathcal{P}\boldsymbol{u}^n = \boldsymbol{n} \cdot \boldsymbol{u}^n$ on $\partial\Omega$. With this additional Poisson equation in each step, for the divergence suppression, we need and only need to change (50) to

$$\frac{1}{\Delta t}\left(\alpha_0^k \boldsymbol{u}^{n+1} + \sum_{j=1}^{k} \alpha_j^k (\boldsymbol{u}^{n+1-j} - \nabla q^{n+1-j})\right) + \nabla \mathcal{E}_k p^{n+1} = \nu \Delta \boldsymbol{u}^{n+1} + \boldsymbol{F}^{n+1}. \tag{54}$$

So, the only modification is to replace the $\boldsymbol{u}^{n+1-j}$ in the time derivative approximation by $\boldsymbol{u}^{n+1-j} - \nabla q^{n+1-j}$.

In practice, since the value of $\boldsymbol{u}^n$ is not given on the open or traction boundaries, the numerical solution $\boldsymbol{u}^n$ may not satisfy the compatibility condition $0 = \int_\Omega \nabla \cdot \boldsymbol{u}^n = \int_{\partial\Omega} \boldsymbol{n} \cdot \boldsymbol{u}$ which is required by (53) in order to have a solution q^n. In the computations, either we ignore this discrepancy, or we play some tricks, like changing the right hand side of (53) to $\nabla \cdot \boldsymbol{u}^n - \frac{1}{|\Omega|}\int_\Omega \nabla \cdot \boldsymbol{u}^n$, or we use another way to define q^n. In the last approach, we try to preserve the tangential velocity on the open or traction boundaries in the Leray projection step, namely, we use

$$\Delta q^n = \nabla \cdot \boldsymbol{u}^n \quad \text{in } \Omega, \quad \boldsymbol{n} \cdot \nabla q^n = 0 \quad \text{on } \Gamma_1, \quad q^n = 0 \quad \text{on } \Gamma_2. \tag{55}$$

Note that the boundary condition $q^n = 0$ on Γ_2 is perfectly physical because $q \equiv 0$ for the exact solution. With the q^n defined by (55), $\mathcal{P}\boldsymbol{u}^n = \boldsymbol{u}^n - \nabla q^n$ would still be divergence free, but now $\boldsymbol{\tau} \cdot \mathcal{P}\boldsymbol{u}^n = \boldsymbol{\tau} \cdot \boldsymbol{u}^n$ on Γ_2 where $\boldsymbol{\tau}$ is any tangential vector on Γ_2.

It turns out that using (55) gives better results than using (53) in our time accuracy check. So, the semi-implicit scheme with further divergence suppression used in our computations will be {(47)–(49), (55), (54), (51), (52)}. For efficiency, we can combine (47)–(49) and (55) into one step. Namely, we can define

$$\bar{p} = \mathcal{E}_k p^{n+1} - \frac{1}{\Delta t}\sum_{j=1}^{k} \alpha_j^k q^{n+1-j} \tag{56}$$

and solve only one pressure-like quantity ($\bar{p}$) per time step. Again, this divergence suppression can be applied to traction boundary condition case too.

6. C^0 finite element formulation

In this section, we will derive the finite element approximation. To simplify the presentation, we will focus on the open boundary condition case (47)–(52) and discuss the divergence suppression and the traction boundary condition case very briefly in the end. We start from a scheme where we use C^1 finite elements for velocity and C^0 finite elements for pressure, and then show how we can use C^0 finite elements for both velocity and pressure.

6.1. Finite element schemes with C^1 finite elements for velocity

Define

$$X_0 = \{\boldsymbol{v} \in H^2(\Omega; \mathbb{R}^m); \boldsymbol{v}|_{\Gamma_1} = 0\}, \quad Y_0 = \{\phi \in H^1(\Omega); \phi|_{\Gamma_2} = 0\} \tag{57}$$

and let X_h and Y_h be the finite element spaces for velocity and pressure. For the moment, we assume $X_h \subset H^2(\Omega; \mathbb{R}^m)$ and $Y_h \subset H^1(\Omega)$. (Later, we will relax it to the case where both X_h and Y_h are C^0 finite element spaces.)

For the pressure equation, we first dot (47) with $\phi_h \in Y_{0,h} := Y_0 \cap Y_h$ and then replace the p^n and $\boldsymbol{u}^n$ in (47) by $p_h^n \in Y_h$ and $\boldsymbol{u}_h^n \in X_h$. We get

$$\langle \nabla p_h^n, \nabla \phi_h \rangle = \langle \boldsymbol{f}^{\,n} - \boldsymbol{h}_h^n, \nabla \phi_h \rangle - \langle \boldsymbol{n} \cdot \partial_t \boldsymbol{g}_1(t_n), \phi_h \rangle_{\Gamma_1} - \nu \langle \boldsymbol{n} \cdot \nabla \times \nabla \times \boldsymbol{u}_h^n, \phi_h \rangle_{\Gamma_1} \tag{58}$$

for any $\phi_h \in Y_{0,h}$. We require the boundary condition $p_h^n = \nu \boldsymbol{n} \cdot (\partial_n \boldsymbol{u}_h^n) - \nu \nabla \cdot \boldsymbol{u}_h^n - \boldsymbol{n} \cdot \boldsymbol{g}_2^n$ on Γ_2 to be valid in $H^{1/2}(\Gamma_2)$.

For the momentum equation, we first dot (50) with $\boldsymbol{v}_h \in X_{0,h} := X_0 \cap X_h$ and then replace the p^n and $\boldsymbol{u}^n$ in (50) by $p_h^n \in Y_h$ and $\boldsymbol{u}_h^n \in X_h$. After integration by parts and using boundary condition (52), we get

$$\frac{1}{\Delta t}\left(\alpha_0^k\langle \boldsymbol{u}_h^{n+1},\boldsymbol{v}_h\rangle+\sum_{j=1}^{k}\alpha_j^k\langle \boldsymbol{u}_h^{n+1-j},\boldsymbol{v}_h\rangle\right)-\langle \mathcal{E}_k p_h^{n+1},\nabla\cdot\boldsymbol{v}_h\rangle+\nu\langle\nabla\boldsymbol{u}_h^{n+1},\nabla\boldsymbol{v}_h\rangle=\langle \boldsymbol{F}_h^{n+1},\boldsymbol{v}_h\rangle+\langle \boldsymbol{g}_2^{n+1},\boldsymbol{v}_h\rangle_{\Gamma_2} \tag{59}$$

for any $\boldsymbol{v}_h\in X_{0,h}$. In the above equation, we have used the shorthand $\boldsymbol{F}_h^{n+1}=\boldsymbol{f}^{\,n+1}-\mathcal{E}_k\boldsymbol{h}_h^{n+1}$. For the boundary value, we require $\boldsymbol{u}_h^{n+1}=\boldsymbol{g}_1^{n+1}$ on Γ_1.

6.2. Finite element schemes with C^0 finite elements for velocity

So far, every inner product and every boundary condition make sense, but we have to use C^1 finite elements for the velocity $\boldsymbol{u}_h^n$ because the last term in (58) contains second order derivatives. C^1 finite element schemes are usually considered to be complicated to implement. In the following, we will try to find a way to use C^0 finite elements only.

Following [27], when $\boldsymbol{u}_h^n\in H^2(\Omega,\mathbb{R}^m)$, we can apply the following identity to reduce the last term in (58) to a term involving first order derivatives only:

$$\begin{aligned}\langle \boldsymbol{n}\cdot\nabla\times\nabla\times\boldsymbol{u}_h^n,\phi_h\rangle_{\Gamma_1}&=\langle \boldsymbol{n}\cdot\nabla\times\nabla\times\boldsymbol{u}_h^n,\phi_h\rangle_{\partial\Omega}=\langle\nabla\times\nabla\times\boldsymbol{u}_h^n,\nabla\phi_h\rangle=\langle \boldsymbol{n}\times\nabla\times\boldsymbol{u}_h^n,\nabla\phi_h\rangle_{\partial\Omega}\\&=-\langle\nabla\times\boldsymbol{u}_h^n,\boldsymbol{n}\times\nabla\phi_h\rangle_{\partial\Omega}=-\langle\nabla\times\boldsymbol{u}_h^n,\boldsymbol{n}\times\nabla\phi_h\rangle_{\Gamma_1}\quad\forall\phi_h\in Y_{0,h}\end{aligned} \tag{60}$$

where in the last step, we have used the fact that $\boldsymbol{n}\times\nabla\phi_h=\boldsymbol{n}\times\left((\boldsymbol{n}\cdot\nabla\phi_h)\boldsymbol{n}+\sum_{i=1}^{m-1}(\boldsymbol{\tau}_i\cdot\nabla\phi_h)\boldsymbol{\tau}_i\right)=0$ on Γ_2 because $\phi_h|_{\Gamma_2}=0$. With (60) replacing the last term in (58), we can now try to use C^0 finite elements for velocity.

So, Define

$$Z_0=\{\boldsymbol{v}\in H^1(\Omega;\mathbb{R}^m);\boldsymbol{v}|_{\Gamma_1}=0\},\quad Y_0=\{\phi\in H^1(\Omega);\phi|_{\Gamma_2}=0\}. \tag{61}$$

Let Z_h and Y_h be the C^0 finite element spaces for velocity and pressure and define $Z_{0,h}=Z_h\cap Z_0, Y_{0,h}=Y_h\cap Y_0$. Then, given $\boldsymbol{u}_h^n\in Z_h$, determine the pressure approximation $p_h^n\in Y_h$ by requiring

$$\langle\nabla p_h^n,\nabla\phi_h\rangle=\langle \boldsymbol{f}^{\,n}-\boldsymbol{h}_h^n,\nabla\phi_h\rangle-\langle \boldsymbol{n}\cdot\partial_t\boldsymbol{g}_1(t_n),\phi_h\rangle_{\Gamma_1}+\nu\langle\nabla\times\boldsymbol{u}_h^n,\boldsymbol{n}\times\nabla\phi_h\rangle_{\Gamma_1} \tag{62}$$

for any $\phi_h\in Y_{0,h}$. When using C^0 Lagrange finite elements for both $\boldsymbol{u}_h^n$ and ϕ_h, $\nabla\times\boldsymbol{u}_h^n$ and $\boldsymbol{n}\times\nabla\phi_h$ are both piecewise polynomials on Γ_1, which enables us to implement $\langle\nabla\times\boldsymbol{u}_h^n,\boldsymbol{n}\times\nabla\phi_h\rangle_{\Gamma_1}$ in (62) without any problem. But in analysis, if we want to bound $\langle\nabla\times\boldsymbol{u}_h^n,\boldsymbol{n}\times\nabla\phi_h\rangle_{\Gamma_1}$, we need, say, $\phi_h\in H^1(\Omega)$ (hence $\nabla\phi_h\in H(\mathrm{curl},\Omega)$) and $\boldsymbol{u}_h^n\in H^2(\Omega)$ which is not available when C^0 finite elements are used for $\boldsymbol{u}_h^n$. In this paper, we will ignore this flaw in analysis and keep using C^0 finite elements because the numerical results turn out to be all right.

In order to enforce the boundary condition (49), we need to project $\boldsymbol{n}\cdot(\partial_n\boldsymbol{u}_h^n)-\nabla\cdot\boldsymbol{u}_h^n$ onto $Y_h|_{\Gamma_2}$ with $Y_h|_{\Gamma_2}=\{\phi|_{\Gamma_2};\phi\in Y_h\}$. If we call the result r_h^n, then the boundary value of p_h^n on Γ_2 is

$$p_h^n=\nu r_h^n-\boldsymbol{n}\cdot\boldsymbol{g}_2^n\quad\text{on }\Gamma_2. \tag{63}$$

After we obtain p_h^n, we can solve for $\boldsymbol{u}_h^{n+1}\in Z_h$ by requiring

$$\frac{1}{\Delta t}\left(\alpha_0^k\langle \boldsymbol{u}_h^{n+1},\boldsymbol{v}_h\rangle+\sum_{j=1}^{k}\alpha_j^k\langle \boldsymbol{u}_h^{n+1-j},\boldsymbol{v}_h\rangle\right)-\langle \mathcal{E}_k p_h^{n+1},\nabla\cdot\boldsymbol{v}_h\rangle+\nu\langle\nabla\boldsymbol{u}_h^{n+1},\nabla\boldsymbol{v}_h\rangle=\langle \boldsymbol{F}_h^{n+1},\boldsymbol{v}_h\rangle+\langle \boldsymbol{g}_2^{n+1},\boldsymbol{v}_h\rangle_{\Gamma_2} \tag{64}$$

for any $\boldsymbol{v}_h\in Z_{0,h}$. The grid values of $\boldsymbol{u}_h^{n+1}$ on Γ_1 is determined by

$$\boldsymbol{u}_h^{n+1}=\boldsymbol{g}_1^{n+1}\quad\text{on }\Gamma_1. \tag{65}$$

6.3. Divergence suppression

So far, we have not included the divergence suppression which can be turned on in a fairly straightforward way. So, given $\boldsymbol{u}_h^n$, we first calculate p_h^n by (62) and (63). Then we determine $q_h^n\in Y_{0,h}$ by requiring

$$\langle\nabla q_h^n,\nabla\phi_h\rangle=-\langle\nabla\cdot\boldsymbol{u}_h^n,\phi_h\rangle \tag{66}$$

for any $\phi_h\in Y_{0,h}$. Finally we determine $\boldsymbol{u}_h^{n+1}\in Z_h$ by solving

$$\frac{1}{\Delta t}\left(\alpha_0^k\langle \boldsymbol{u}_h^{n+1},\boldsymbol{v}_h\rangle+\sum_{j=1}^{k}\alpha_j^k\langle \boldsymbol{u}_h^{n+1-j}-\nabla q_h^{n+1-j},\boldsymbol{v}_h\rangle\right)-\langle \mathcal{E}_k p_h^{n+1},\nabla\cdot\boldsymbol{v}_h\rangle+\nu\langle\nabla\boldsymbol{u}_h^{n+1},\nabla\boldsymbol{v}_h\rangle=\langle \boldsymbol{F}_h^{n+1},\boldsymbol{v}_h\rangle+\langle \boldsymbol{g}_2^{n+1},\boldsymbol{v}_h\rangle_{\Gamma_2} \tag{67}$$

for any $\boldsymbol{v}_h\in Z_{0,h}$ and also using (65). After writing $\langle\nabla q_h^{n+1-j},\boldsymbol{v}_h\rangle$ as $-\langle q_h^{n+1-j},\nabla\cdot\boldsymbol{v}_h\rangle$ using $\boldsymbol{v}_h=0$ on Γ_1 and $q_h^{n+1-j}=0$ on Γ_2, we can define $\bar{p}_h$ like in (56) so that only one pressure-like quantity is need to be solved in each step.

6.4. Traction boundary condition case

All the derivations we have done for the open boundary condition case can be readily applied to the traction boundary condition case. We can obtain the following equations for $\boldsymbol{u}_h^{n+1}$:

$$\frac{1}{\Delta t}\left(\alpha_0^k\langle \boldsymbol{u}_h^{n+1},\boldsymbol{v}_h\rangle+\sum_{j=1}^{k}\alpha_j^k\langle \boldsymbol{u}_h^{n+1-j}-\nabla q_h^{n+1-j},\boldsymbol{v}_h\rangle\right)-\langle \mathcal{E}_k p_h^{n+1},\nabla\cdot\boldsymbol{v}_h\rangle+\nu\langle\nabla\boldsymbol{u}_h^{n+1}+\nabla\boldsymbol{u}_h^{n+1,\top},\nabla\boldsymbol{v}_h\rangle$$
$$=\langle \boldsymbol{F}_h^{n+1},\boldsymbol{v}_h\rangle+\langle \boldsymbol{g}_2^{n+1},\boldsymbol{v}_h\rangle_{\Gamma_2} \tag{68}$$

for any $\boldsymbol{v}_h\in Z_{0,h}$. Because $\langle\nabla\boldsymbol{u}+\nabla\boldsymbol{u}^\top,\nabla\boldsymbol{v}-\nabla\boldsymbol{v}^\top\rangle=0$ for any $\boldsymbol{u},\boldsymbol{v}\in H^1(\Omega,\mathbb{R}^m)$, we can rewrite the $\nu\langle\nabla\boldsymbol{u}_h^{n+1}+\nabla\boldsymbol{u}_h^{n+1,\top},\nabla\boldsymbol{v}_h\rangle$ in (68) into the symmetric form

$$\frac{\nu}{2}\langle\nabla\boldsymbol{u}_h^{n+1}+\nabla\boldsymbol{u}_h^{n+1,\top},\nabla\boldsymbol{v}_h+\nabla\boldsymbol{v}_h^\top\rangle.$$

The finite element equations for p_h^n and q_h^n remain the same, except the r_h^n in (63) is now the projection of $\boldsymbol{n}^\top(\nabla\boldsymbol{u}_h^n+\nabla\boldsymbol{u}_h^{n,\top})\boldsymbol{n}-\nabla\cdot\boldsymbol{u}_h^n$ onto $Y_h|_{\Gamma_2}$.

7. Numerical tests

In this section, we will test the stability, accuracy and benchmark performance of the semi-implicit schemes for open and traction boundary conditions.

To simplify the presentation, we will use the divergence suppression mentioned in Section 5 in all the computations, even though this is not necessary when the exact solution is smooth. For open boundary condition (4), the scheme that will be tested is {(47)–(49), (55), (54), (51), (52)} and its finite element version {(62), (63), (66), (67), (65)}. The scheme used for traction boundary condition (5) is {(62), (63), (66), (68), (65)}. We will test the unconditional stability of the first order scheme and verify the 2nd order temporal accuracy of the 2nd order scheme. The latter is also used in most of the benchmark computations. To save space, for the traction boundary condition, we only present results of the benchmark computations.

In our numerical study, when the viscosity is large, say, $\nu\geqslant 0.1$, we can take very large time step in the first order semi-implicit scheme for the fully nonlinear NSE and seem to have unconditional stability which is consistent with Proposition 3 (see Table 2 and Fig. 8). However, starting form the 2nd order scheme, some preliminary numerical tests indicate that we might have diffusive time step restrictions when the Dirichlet and Neuman boundaries intersect, i.e. when $\overline{\Gamma}_1\bigcap\overline{\Gamma}_2\neq\emptyset$, even though we do have higher order accuracy in time. In large Reynolds number computations, the diffusive time step restriction is not a big problem. For small Reynolds number computations with open or traction boundaries, we suggest to use the first order scheme if one does not want to use a very small time step. An alternative choice is to use the scheme discussed in [25] which has inherited the stability of the first order scheme and at the same time is temporally spectrally accurate and is also very efficient. When there are no open or traction boundaries, stable schemes up to 3rd order have been obtained in [32,33].

We use equal order velocity/pressure pairs in all the finite element computations which are known to not satisfy the inf–sup condition [14]. The finite element package we have implemented is in some sense an upgraded version of iFEM due to Long Chen (work in preparation, see http://math.uci.edu/~chenlong/). iFEM is an adaptive piecewise linear finite element package based on MATLAB. It uses a beautiful data structure to represent the mesh and also provides efficient MATLAB subroutines to manipulate the mesh (e.g. see [10,1]). In particular, local refinement and coarsening can be done fairly easily. For our purposes, we have extended it to isoparametric Lagrange elements up to P4 [41,16]. The finite element mesh is generated using DistMesh of Persson and Strang [36]. The contour plots on unstructured mesh are generated by the MATLAB routine `tricontour.m` due to Darren Engwirda [15].

For details about collocative spectral methods, we refer to [45] or [38].

The stability and accuracy tests are performed using the following exact solution

$$u=\cos(t)\cos^2(\pi x/2)\sin(\pi y), \tag{69}$$
$$v=-\cos(t)\sin(\pi x)\cos^2(\pi y/2), \tag{70}$$
$$p=\cos(t)\cos(\pi x/2)\sin(\pi y/2). \tag{71}$$

The computational domain is either $[-0.5,0.5]\times[-0.5,0.5]$ for collocative spectral method [45,38] or $[-0.5,0.5]\times[-0.5,0.5]\setminus\{x^2+y^2\leqslant 0.2^2\}$ for finite element method. In both cases, the boundary $\{x=0.5\}$ is taken to be

Table 2
Stability check for first order scheme ($k=1$) with open boundary conditions. Collocative spectral or P2/P2 finite element (FE) discretization. $-\log_{10}E$ vs. Δt. $T=10,000$. $\nu=1$. We get similar results for traction boundary conditions.

Scheme	$E\setminus\Delta t$	1/4	1/2	1	2	4	8
Spectral	$\|p-p_h\|_{L^\infty}$	0.115	0.178	0.0415	−0.366	−0.153	−0.224
	$\|u-u_h\|_{L^\infty}$	1.7	1.51	1.25	0.968	0.794	0.95
	$\|v-v_h\|_{L^\infty}$	2.11	1.69	1.23	0.682	0.517	0.768
P2/P2 FE	$\|p-p_h\|_{L^\infty}$	0.232	0.142	0.125	−0.0618	0.0889	0.0256
	$\|u-u_h\|_{L^\infty}$	1.7	1.47	1.24	0.947	0.916	0.965
	$\|v-v_h\|_{L^\infty}$	1.79	1.66	1.42	0.923	0.903	0.956

Γ_2 while the rest of the boundary is Γ_1. The exact value of $\nu\partial_n \boldsymbol{u} - p\,\boldsymbol{n}$ or $\nu(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\top)\boldsymbol{n} - p\,\boldsymbol{n}$ is calculated and then enforced at the boundary $\{x = 0.5\}$. Fig. 2 shows the finite element mesh used in stability check. It is also the coarsest mesh used in temporal accuracy check.

7.1. Stability

For the stability check of the first order scheme, we integrate to $T = 10,000$ with Δt as large as 8. For spectral method, the grid is fixed to be 36×36. For finite element method, the mesh is shown in Fig. 2. If any element has an edge on the circle, it is an isoparametric element. There are 324 P2 finite elements (dof = 740) for each variable.

The stability results are listed in Table 2. In the table, the main quantity tabulated is $-\log_{10}E$, where E is the quantity listed in the 2nd column. We seem to have unconditional stability of the first order scheme even for the full NSE which is consistent with the analysis for the Stokes equation in Proposition 3. Please see also Fig. 8 for a more interesting stability test where we take $\Delta t = 1$.

In Tables 2–4, we have used u, v, p to denote the exact solution (69)–(71) and use u_h, v_h, p_h to denote the numerical solution.

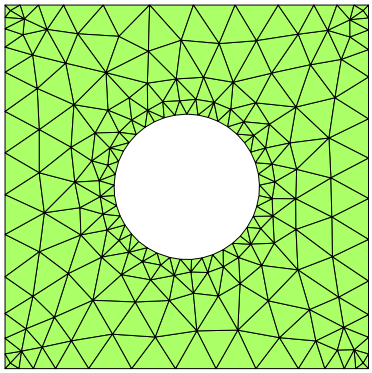


Fig. 2. Mesh used in stability and temporal accuracy check.

Table 3
Temporal accuracy of the first order scheme with open boundary conditions. Collocative spectral discretization with 90×90 mesh. Spatial errors can be ignored comparing with temporal errors. $-\log_{10}E$ (and local order α) vs. Δt. $T = 4$. $\nu = 1$.

$E \setminus \Delta t$	3^{-3}	3^{-4}	3^{-5}	3^{-6}
$\|p - p_h\|_{L^\infty}$	0.52 (0.905)	0.952 (1.02)	1.44 (1.01)	1.92 (1.01)
$\|\nabla(p - p_h)\|_{L^2}$	0.734 (0.944)	1.18 (1.01)	1.67 (1.01)	2.15 (1)
$\|u - u_h\|_{L^\infty}$	1.97 (0.838)	2.37 (0.943)	2.82 (0.981)	3.29 (0.994)
$\|\nabla(u - u_h)\|_{L^2}$	1.73 (0.855)	2.13 (0.947)	2.59 (0.982)	3.06 (0.994)
$\|v - v_h\|_{L^\infty}$	2.27 (0.983)	2.74 (0.996)	3.21 (0.999)	3.69 (1)
$\|\nabla(v - v_h)\|_{L^2}$	1.89 (0.985)	2.36 (0.996)	2.83 (0.999)	3.31 (1)

Table 4
Temporal accuracy of the 2nd order scheme with open boundary conditions. Finite element discretization. $-\log_{10}E$ (and local order α) vs. Δt. The coarsest mesh is shown in Fig. 2 which is used when $\Delta t = 0.01$. When Δt is reduced by half, one triangle breaks into 4 triangles ($h \to h/2$). P2/P2 isoparametric finite elements are used so that spatial errors are less than or equal to the temporal errors. This table also shows the P2/P2 discretization has 2nd order accuracy in space for velocity and pressure gradients in L^2 norm. $T = 2$. $\nu = 0.001$. We get similar results for traction boundary conditions.

$E \setminus \Delta t$	0.01	0.005	0.0025	0.00125
$\|p - p_h\|_{L^\infty}$	3.98 (2.83)	4.83 (1.95)	5.42 (2.04)	6.03 (2.02)
$\|\nabla(p - p_h)\|_{L^2}$	2.9 (2.17)	3.55 (2.33)	4.25 (2.23)	4.92 (2.08)
$\|u - u_h\|_{L^\infty}$	3.26 (2.18)	3.92 (2.16)	4.57 (2.02)	5.18 (2.02)
$\|\nabla(u - u_h)\|_{L^2}$	2.26 (2.23)	2.93 (2.38)	3.64 (2.26)	4.32 (2.1)
$\|v - v_h\|_{L^\infty}$	3.32 (3.05)	4.24 (2.14)	4.88 (2.07)	5.51 (2.04)
$\|\nabla(v - v_h)\|_{L^2}$	2.28 (2.18)	2.93 (2.4)	3.65 (2.3)	4.35 (2.13)

7.2. Temporal accuracy

In Table 3, we take time steps $\Delta t_k = 1/3^k$ for $k = 3$ to 7 and integrate to $T = 4$ to do a temporal accuracy check for the first order scheme. We use collocative spectral discretization and use a fixed 90×90 grid. So, the spatial errors can be ignored comparing with the temporal errors. The main quantity tabulated is $-\log_{10}E$, where E is the quantity listed in the first column. In parentheses we also list the local convergence rate α for E, determined from the formula

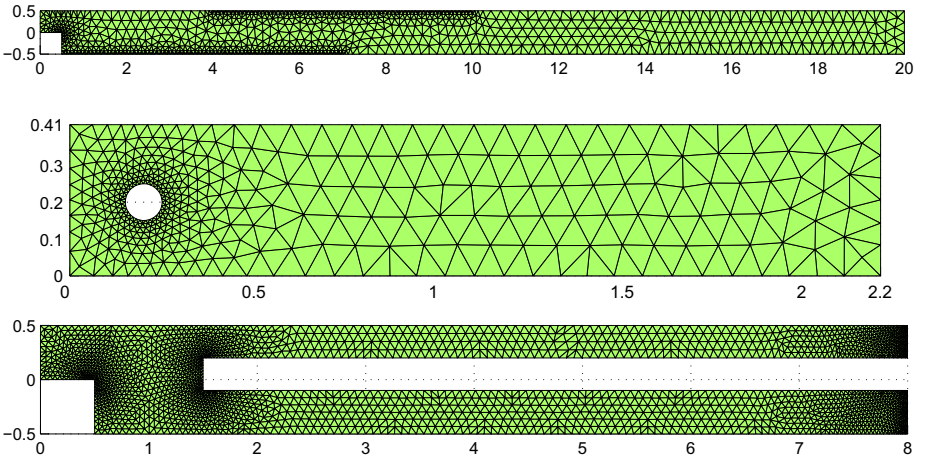


Fig. 3. Meshes used in backward facing step flow computation when $\nu = 1/600$, in flow pass a cylinder calculation when $\nu = 1/1000$ and in flow in a bifurcated tube calculation when $\nu = 1/10, 1/600$.

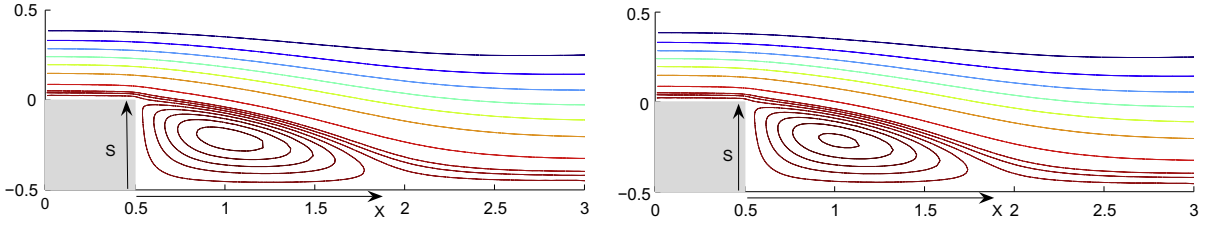


Fig. 4. Backward facing step. $\nu = 1/100$. 2236 P2 elements (dof = 4691) for each variable. 2nd order scheme. $\Delta t = 0.005$, $T = 20$. $X/S = 2.87$. Left: open boundary condition. Right: zero traction boundary condition.

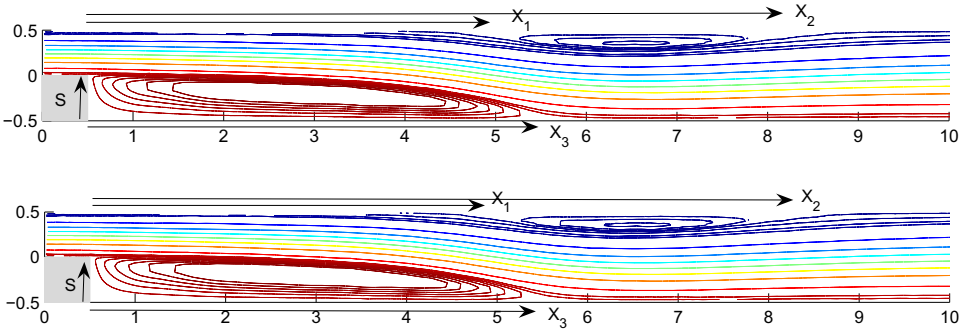
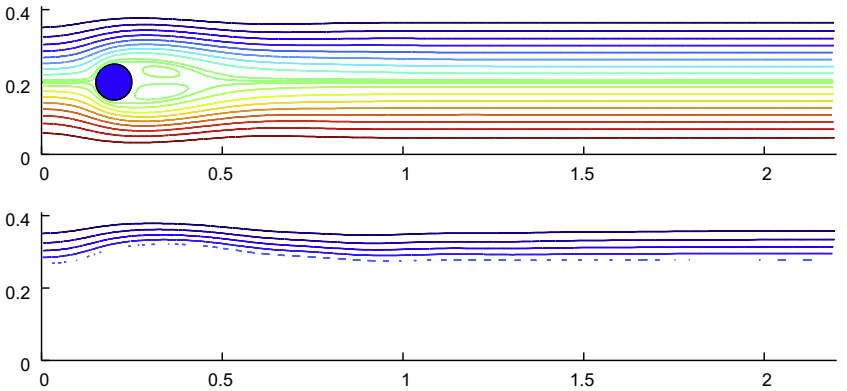


Fig. 5. Backward facing step. $\nu = 1/600$. 1700 P2 elements (dof = 3925) for each variable. 2nd order scheme. $\Delta t = 0.003, T = 120$. $X_1/S = 8.95, X_2/S = 15.54, X_3/S = 9.9$. Top: open boundary condition. Bottom: zero traction boundary condition.

$$\alpha = \frac{\log(E_k/E_{k+1})}{\log(\Delta t_k/\Delta t_{k+1})}. \tag{72}$$

Table 3 shows that we do have first order accuracy in time for the first order scheme for all flow variables in both L^∞ and H^1 norms.

We also tested the accuracy of our finite element schemes described in Sections 6.3 and 6.4 with the exact solution (69)–(71). The temporally 2nd order semi-implicit scheme being tested is also the scheme used in most benchmark computations. We use P2/P2 finite elements which have 2nd order spatial accuracy for the H^1 norm of all flow variables. We vary Δt but also refine the mesh so that $\Delta t/h$ remains constant during the process. This guarantees that the spatial errors are smaller than or equal to the temporal errors. The numerical results in Table 4 show that we do have 2nd order accuracy in time for the 2nd order scheme. As a by-product, Table 4 also indicates that the spatial accuracy of velocity and pressure gradients in L^2 norm is at least 2nd order with P2/P2 discretization.



$$\boldsymbol{u}(t,0,y) = 0.41^{-2}\sin(\pi t/8)(6y(0.41-y),0) \tag{73}$$

is prescribed. The outflow boundary condition is either $v\partial_n\boldsymbol{u} - p\,\boldsymbol{n} = 0$ or $v(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top)\boldsymbol{n} - p\,\boldsymbol{n} = 0$ at outflow boundary $\{x = 2.2\}$. Based on the maximum velocity $U_{\max} = 1$ and the diameter of the cylinder $L = 0.1$, the Reynolds number of the flow is 100. The computational mesh is shown in Fig. 3 and the streamline plot at $t = [2,4,5,6,7,8]$ is shown in Fig. 6. However, if one compares Fig. 6 with the results we obtained using Dirichlet type outflow boundary condition where a parabolic velocity profile is prescribed at $\{x = 2.2\}$ [33,26], one realizes that at $t = 8$ there is a significant difference: The last eddy is cut through by the outflow boundary $\{x = 2.2\}$ if we use the open boundary or zero traction boundary conditions, but the

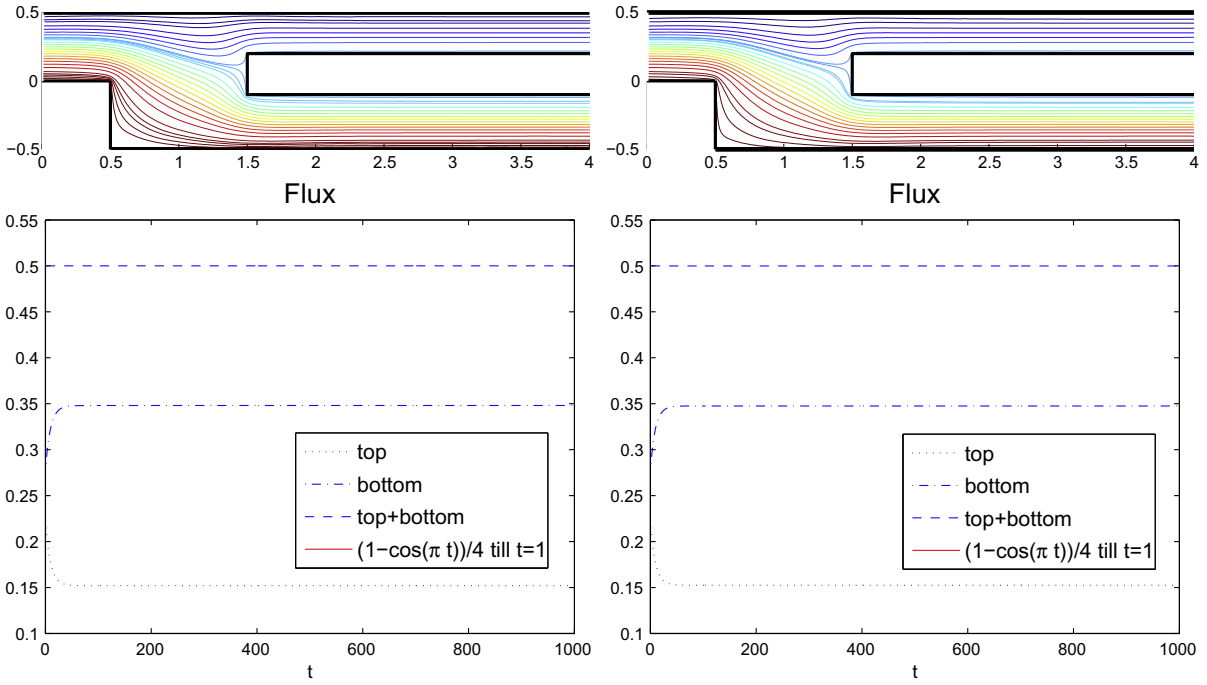


Fig. 8. Bifurcated tube. 4808 isoparametric P2 elements (dof = 10209) for each variable. First order scheme. $v = 0.1$. $\Delta t = 1$ and integrate to $T = 1000$. So, we start from zero velocity at $t = 0$ and then impulsively impose the inflow boundary condition $(u,v) = (12y(1-2y),0)$ on $x = 0$ at $t = \Delta t = 1$. The flux plots show the fluxes of the top and bottom branches and also the sum of the two which turns out to equal to the total influx 0.5. Left: open boundary condition. Right: zero traction boundary condition.

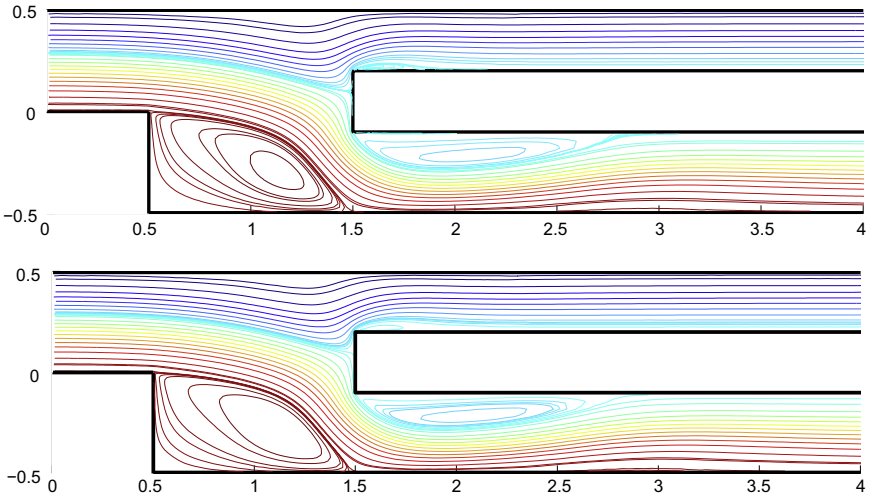


Fig. 9. Bifurcated tube. 4808 isoparametric P2 elements (dof = 10209) for each variable. 2nd order scheme. $v = 1/600$. $\Delta t = 0.0005$. $T = 60$. Top: open boundary condition. Bottom: zero traction boundary condition.

last eddy will remain on the left hand side of $\{x = 2.2\}$ completely if we prescribe a parabolic velocity profile at $\{x = 2.2\}$. This indicates that we might not get a right picture near the outflow boundary if we prescribe a parabolic outflow profile brutally. We think the Dirichlet type parabolic outflow profile is less physical because it is hard to believe that both eddies near the top and bottom walls will vanish at the same position at $\{x = 2.2\}$, given their previous alternating pattern. Nevertheless, this seems to have little influence on the upstream flow that is around the cylinder. For a quantitative comparison, we calculate the x and y components of the following quantity, which are denoted as $c_d(t)$ and $c_l(t)$, the drag and lift coefficients, respectively:

$$\frac{2}{LU_{\max}^2}\int_S \nu\partial_n \boldsymbol{u} - p\,\boldsymbol{n} \tag{74}$$

where S is the surface of the cylinder. Note that one can show with $\nabla\cdot\boldsymbol{u} = 0$ and $\boldsymbol{u} = 0$ on S, $(\nabla\boldsymbol{u}^\top)\boldsymbol{n} = (\partial_i u_j)n_j = 0$ on S and hence the pseudo-traction used in (74) is the traction [6]. We faithfully calculate the above quantity by surface integration, instead of transforming it into volume integration. We have also calculated when the maxima of c_d and c_l occur. These values are shown in Fig. 7 together with the pressure difference between the front and the back of the cylinder

$$\Delta p(t) = p(t, 0.15, 0.2) - p(t, 0.25, 0.2). \tag{75}$$

Results shown in Fig. 7 agree rather well with results in [26,33]. We mention in passing that if we do not turn on the divergence suppression (66), the flow remains nearly symmetric and the vortex street will not appear.

For the flow in a bifurcated tube with $\nu = 1/10$ and $1/600$, the computational domain is

$$\Omega = [0, 8]\times[-0.5, 0.5]\setminus\{[0, 0.5]\times[-0.5, 0]\cup[1.5, 8]\times[-0.1, 0.2]\}.$$

The computational mesh is shown in Fig. 3. The same inflow boundary condition as in the backward facing step flow is used at the $\{x = 0\}$ boundary. The outflow boundary $\{x = 8\}$ contains two parts: $\{x = 8, 0.2 \leqslant y \leqslant 0.5\}$ (top) and

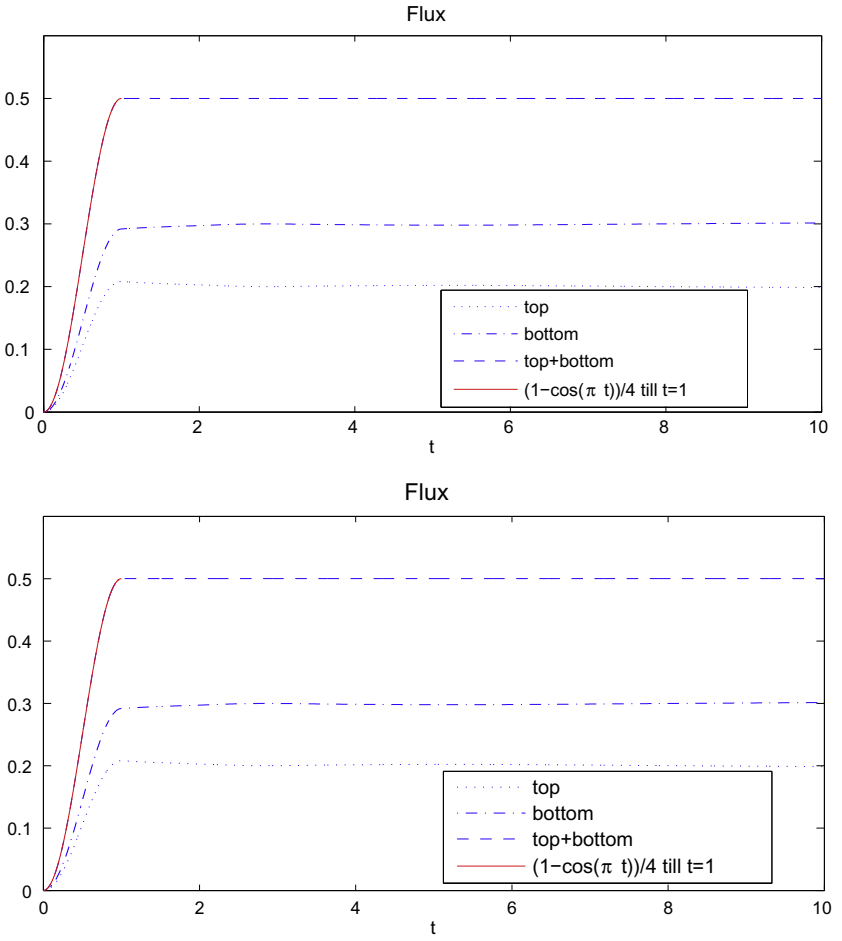


Fig. 10. Bifurcated tube. 4808 isoparametric P2 elements (dof = 10209) for each variable. 2nd order scheme. $\nu = 1/600$. $\Delta t = 0.0005$. The plots show the fluxes of the top and bottom branches and also the sum of the two which turns out to equal to the total influx at any time. Top: open boundary condition. Bottom: zero traction boundary condition.

$\{x = 8, -0.5 \leqslant y \leqslant -0.1\}$ (bottom). Since the tube bifurcates, we have no way to determine a priori what is the outflow profile for each branch and hence it is impossible to use schemes that require velocity to be given on the whole boundary. Nevertheless, we can use condition $\nu\partial_n \boldsymbol{u} - p\,\boldsymbol{n} = 0$ or $\nu(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\top)\boldsymbol{n} - p\,\boldsymbol{n} = 0$ on both the top and bottom parts of the outflow boundaries. Recall that this is one of our main motivations to study the open and traction boundary conditions. The stability of the first order semi-implicit scheme can be clearly seen from the $\nu = 0.1$ results in Fig. 8 where we take $\Delta t = 1$ and impulsively impose the inflow profile $(u, v) = (12y(1-2y), 0)$ at the inflow boundary. In Fig. 8, the fluxes are calculated based on $\boldsymbol{u}^* := \boldsymbol{u} - \nabla q$ in each step so that the influx and outflux are balanced from very beginning even though it is a impulsively started problem and we do not know the outflow profile. The q is defined by (55).

Fig. 9 shows the streamline of the steady state when $\nu = 1/600$. In particular, we see three eddies even though the eddy near corner $\{x = 1.5, y = 0.2\}$ is very small. In Fig. 10, we also plot the outfluxes at $\{x = 8, 0.2 \leqslant y \leqslant 0.5\}$ (top) and $\{x = 8, -0.5 \leqslant y \leqslant -0.1\}$ (bottom) and show that their sum equals to the influx at the $\{x = 0\}$ boundary. In Fig. 10, the fluxes are calculated using $\boldsymbol{u}$, not $\boldsymbol{u} - \nabla q$. When $t \in [0, 1]$, we see that the curve of the total outflux coincides with the curve $(1 - \cos(\pi t))/4$ which is the total influx.

Acknowledgments

The author thanks Prof. Long Chen for the finite element package iFEM based on which the finite element calculations in this paper are done. The author wishes to thank Prof. J.-G. Liu and Prof. R. L. Pego for their helpful comments and encouragements. The author also thanks the reviewers for their helpful comments and for pointing out the reference [20].

References

[1] J. Alberty, C. Carstensen, S.A. Funken, Remarks around 50 lines of Matlab: short finite element implementation, Numer. Algorithms 20 (1999) 117–137.
[2] B.F. Armaly, F. Durst, J.C.F. Pereira, B. Schönung, Experimental and theoretical investigation of backward-facing step flow, J. Fluid Mech. 127 (1983) 473–496.
[3] D.N. Arnold, R.S. Falk, R. Winther, Differential complexes and stability of finite element methods I: the de Rham complex, in: Compatible Spatial Discretizations, vol. 142 of The IMA Volumes in Mathematics and Its Applications, Springer, Berlin, 2006, pp. 23–46.
[4] D.N. Arnold, R.S. Falk, R. Winther, Differential complexes and stability of finite element methods II: the elasticity complex, in: Compatible Spatial Discretizations, vol. 142 of The IMA Volumes in Mathematics and Its Applications, Springer, Berlin, 2006, pp. 47–68.
[5] I. Babuska, A.K. Aziz, Lectures on the mathematical foundations of the finite element method. University of Maryland, College Park, 1972, Technical Note BN-748.
[6] J.T. Beale, J. Strain, Locally corrected semi-Lagrangian methods for Stokes flow with moving elastic interfaces, J. Comput. Phys. 227 (2008) 3896–3920.
[7] P.B. Bochev, C.R. Dohrmann, M.D. Gunzburger, Stabilization of low-order mixed finite elements for the Stokes equations, SIAM J. Numer. Anal. 44 (2006) 82–101.
[8] S.C. Brenner, L.R. Scott, The mathematical theory of finite element methods, Springer-Verlag, New York, 1994.
[9] F. Brezzi, On the existence uniqueness and approximation of saddle-point problems arising from Lagrange multipliers, R.A.I.R.O. Anal. Numer. R2 (1974) 129–151.
[10] L. Chen, C.-S. Zhang, A coarsening algorithm and multilevel preconditioners on adaptive grids by newest vertex bisection, submitted for publication.
[11] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comput. 22 (1968) 745–762.
[12] D. Coutand, S. Shkoller, The interaction between quasilinear elastodynamics and the Navier–Stokes equations, Arch. Rational Mech. Anal. 179 (2006) 303–352.
[13] C. Cuvelier, R.M.S.M. Schulkes, Some numerical methods for the computation of capillary free boundaries governed by the Navier–Stokes equations, SIAM Review 32 (1990) 355–423.
[14] H. Elman, D. Silvester, A. Wathen, Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics, Oxford University Press, 2005.
[15] D. Engwirda, Contours for triangular grids <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=10408>, MATLAB Central File Exchange. Retrieved Sept. 15, 2008, 2006.
[16] M.S. Gockenbach, Understanding and Implementing the Finite Element Method, SIAM, Philadelphia, 2006.
[17] P.M. Gresho, Incompressible fluid dynamics: some fundamental formulation issues, Annu. Rev. Fluid Mech. 23 (1991) 413–453.
[18] G. Grubb, V.A. Solonnikov, Boundary value problems for the nonstationary Navier–Stokes equations treated by pseudodifferential methods, Math. Scand. 69 (1991) 217–290.
[19] J.L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Eng. 195 (44–47) (2006) 6011–6045.
[20] J.L. Guermond, P. Minev, J. Shen, Error analysis of pressure-correction schemes for the time-dependent Stokes equations with open boundary conditions, SIAM J. Numer. Anal. 43 (2005) 239–258.
[21] J.L. Guermond, L. Quartapelle, On stability and convergence of projection methods based on pressure Poisson equation, Int. J. Numer. Methods Fluids 26 (1998) 1039–1053.
[22] J.L. Guermond, J. Shen, A new class of truly consistent splitting schemes for incompressible flows, J. Comput. Phys. 192 (2003) 262–276.
[23] J.L. Guermond, J. Shen, Velocity-correction projection methods for incompressible flows, SIAM J. Numer. Anal. 41 (1) (2003) 112–134.
[24] J.L. Guermond, J. Shen, X. Yang, Error analysis of fuly discrete velocity-correction methods for incompressible flows, Math. Comp. 77 (2008) 1387–1405.
[25] J. Jia, J. Liu, Stable and spectrally accurate schemes for Navier–Stokes equations with no-slip and open Boundaries, submitted for publication.
[26] V. John, Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder, Int. J. Num. Meth. Fluids 44 (2004) 777–788.
[27] H. Johnston, J.-G. Liu, Accurate, stable and efficient Navier–Stokes solvers based on explicit treatment of the pressure term, J. Comp. Phys. 199 (1) (2004) 221–259.
[28] G.E. Karniadakis, M. Israeli, S.A. Orszag, High-order splitting methods for the incompressible Navier–Stokes equations, J. Comp. Phys. 97 (1991) 414–443.
[29] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Comp. Phys. 59 (1985) 308–323.
[30] E. Leriche, E. Perchat, G. Labrosse, M.O. Deville, Numerical evaluation of the accuracy and stability properties of high-order direct Stokes solvers with or without temporal splitting, J. Sci. Comput. 26 (2006) 25–43.
[31] J.-G. Liu, J. Liu, R. Pego, Stability and convergence of efficient Navier–Stokes solvers via a commutator estimate, Comm. Pure Appl. Math. 60 (2007) 1443–1487.
[32] J.-G. Liu, J. Liu, R. Pego, Error estimates for finite-element Navier–Stokes solvers without standard inf–sup conditions, submitted for publication.

[33] J.-G. Liu, J. Liu, R. Pego, Stable and accurate pressure approximation for unsteady incompressible viscous flow, submitted for publication.
[34] S.A. Orszag, M. Israeli, M. Deville, Boundary conditions for incompressible flows, J. Sci. Comput. 1 (1986) 75–111.
[35] G. Pedrizzetti, K. Perktold (Eds.), Cardiovascular fluid mechanics, CISM Courses and Lectures, No.446, Springer-Verlag, Wien, New York, 2003.
[36] P.-O. Persson, G. Strang, A simple mesh generator in MATLAB, SIAM Review 46 (2004) 329–345.
[37] N.A. Petersson, Stability of pressure boundary conditions for Stokes and Navier–Stokes equations, J. Comp. Phys. 172 (2001) 40–70.
[38] R. Peyret, Spectral methods for incompressible viscous flow, Appl. Math. Sci., vol. 148, Springer, New York, 2002.
[39] R.L. Sani, P.M. Gresho, Resume and remarks on the open boundary condition minisymposium, Int. J. Numer. Methods Fluids 18 (1994) 983–1008.
[40] R.L. Sani, J. Shen, O. Pironneau, P.M. Gresho, Pressure boundary condition for the time-dependent incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 50 (2006) 673–682.
[41] R. Scott, Finite element techniques for curved boundaries, Thesis, Massachusetts Institute of Technology, Cambridge, 1973.
[42] R. Temam, Sur l'approximation de la solution des equations de Navier–Stokes par la méthode des fractionnarires II, Arch. Rational Mech. Anal. 33 (1969) 377–385.
[43] R. Temam, Navier–Stokes equations: theory and numerical analysis. AMS Chelsea, Providence, 2001.
[44] L.J.P. Timmermans, P.D. Minev, F.N. Van De Vosse, An approximate projection scheme for incompressible flow using spectral elements, Int. J. Numer. Methods Fluids 22 (1996) 673–688.
[45] L.N. Trefethen, Spectral methods in MATLAB, Software Environments and Tools 10, SIAM, Philadelphia, 2000.