Available online at www.sciencedirect.com
ScienceDirect
Computer methods in applied mechanics and engineering
Comput. Methods Appl. Mech. Engrg. 279 (2014) 198–211
www.elsevier.com/locate/cma

A finite element pressure correction scheme for the Navier–Stokes equations with traction boundary condition

Eberhard Bänsch*

Applied Mathematics III, University Erlangen–Nürnberg, Cauerstr. 11, 91058 Erlangen, Germany

Received 24 September 2013; received in revised form 4 June 2014; accepted 9 June 2014
Available online 30 June 2014

Abstract

We consider the popular pressure correction scheme for the solution of the time dependent Navier–Stokes equations with *traction* boundary condition. A finite element based method to improve the performance of the classical approach is proposed. The improvement is achieved by modifying the traction boundary condition for the provisional velocity $\tilde{u}^{n+1}$ in each time step. The corresponding term consists of a simple boundary functional involving the normal derivative of the pressure correction that can be evaluated in a natural and easy way in the context of finite elements.

Computational results show a significant improvement of the solution, in particular for the pressure in the case of smooth domains.

MSC: 35K; 65M; 76D

Keywords: Navier–Stokes equations; Traction boundary condition; Open boundary condition; Pressure correction; Finite elements

0. Introduction

Because of their conceptual simplicity and low computational cost projection schemes are very popular methods to solve the unsteady incompressible Navier–Stokes equation. The history of these schemes dates back to the pioneering work of Chorin and Temam [1–3]. However, the price one usually has to pay for the simplicity of these schemes is a strong splitting error becoming manifest in an order reduction of the error. This is caused by the decoupling of velocity and pressure (that is at the core of the methods) resulting for instance in a non-physical behavior of the pressure close to the boundary [4]. Consequently much effort has been spent in removing or at least reducing this effect.

It is out of scope of this presentation to cite even the most relevant papers from the abundant literature on splitting or projection methods. Instead, we refer to the excellent overview paper [5] and the papers cited there for further reference.

* Tel.: +49 9131 85 67 202.
E-mail address: eberhard.baensch@math.fau.de.

http://dx.doi.org/10.1016/j.cma.2014.06.030
0045-7825/

In this paper we are concerned with the time dependent Navier–Stokes equations, where a *traction boundary condition*

$$\sigma(u, p)\mathbf{n} = g \quad \text{on } \Gamma$$

is imposed on a part Γ of the boundary of the domain Ω. Here, $\sigma(u, p) = \frac{1}{\mathrm{Re}} D(u) - p\mathbf{I}$ denotes the stress tensor with the rate of strain tensor $D(u) = (\nabla u + \nabla u^T)$. Re is the Reynolds number and $\mathbf{n}$ the outward unit normal on Γ. The above type of boundary condition is important in itself for instance in certain technical applications and furthermore (maybe even more important) it constitutes the core problem in solving free boundary problems.

Note that replacing $D(u)$ by ∇u in the boundary condition results in a popular *open boundary* (or *do-nothing*) condition, see for instance [6–8]. This slightly simpler problem is also covered by our approach below.

Unlike for a fully coupled discretization, i.e. solving a saddle point problem in every time step, splitting schemes applied to flow problems with traction boundary conditions introduce much bigger errors than for Dirichlet boundary conditions. The reason behind may be seen in the fact that this type of boundary condition additionally couples velocity and pressure. Furthermore, despite its relevance, the question of splitting schemes for problems with traction boundary conditions is much less addressed than for Dirichlet conditions.

In this paper we introduce an improvement to the classical pressure correction schemes with traction boundary condition. Our approach is based on the following simple idea. Naturally, one usually imposes the following boundary condition for the provisional velocity $\tilde{u}$ at time instant t_{n+1}:

$$\sigma(\tilde{u}^{n+1}, p^n)\mathbf{n} = g^{n+1} \quad \text{on } \Gamma.$$

This choice, however, leads to an inconsistent boundary condition for the solution (u^{n+1}, p^{n+1}). Therefore we modify the above boundary condition by adding a still to be determined function l^{n+1} that hopefully leads to a more consistent boundary condition:

$$\sigma(\tilde{u}^{n+1}, p^n)\mathbf{n} = g^{n+1} + l^{n+1} \quad \text{on } \Gamma.$$

By some differential geometry calculus it is possible to determine l^{n+1} such that one even ends up with the *correct* boundary condition for (u^{n+1}, p^{n+1}). However, this would require again a fully coupled approach, since the correct l^{n+1} requires the new value of the pressure correction Φ^{n+1} that is not known at this stage of the scheme, see Section 3.2 below. Instead, an appropriate extrapolation for l^{n+1} can be used.

Let us mention some related work in the context of finite elements. The only rigorous error analysis for a pressure correction scheme with open boundary condition we are aware of is [9], see also Section 3.2. In [10] the Navier–Stokes equations in 2d with open boundary condition on a *straight* part of the boundary are considered. The approach is based on a Neumann-to-Dirichlet operator for the pressure in the context of the so called *unconstrained Navier–Stokes* equation approach introduced in [11,12]. Poux et al. [13] use an extrapolation for the boundary condition for the pressure correction Φ, again for the case of a *straight* part of the boundary of a 2d domain. This approach is somehow similar to ours, but differs even in the case of planar boundaries. Moreover, it is not obvious how to generalize the idea in [13] to curved boundaries.

The rest of this article is organized as follows. In Section 1 we state the problem and introduce some notation. Section 2 gives some results on differential operators on manifolds that are needed to develop our method. In Section 3 the new scheme is introduced. Computational results showing the improvement by the traction correction are discussed in Section 4. The paper is concluded by some final remarks in Section 5.

1. Notation and preliminaries

Let $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$ be an open, connected and bounded domain with a sufficiently smooth traction boundary Γ. The approach and results of this paper hold, if Γ is a sub-manifold of $\partial\Omega$ without boundary and $\partial\Omega \setminus \Gamma$ is a Dirichlet boundary. For ease of presentation, however, in what follows, it is assumed that $\Gamma = \partial\Omega$. Consider the incompressible Navier–Stokes equation on a time interval $]0, T[$: find the velocity u and the pressure p fulfilling

$$\partial_t u + u \cdot \nabla u - \nabla \cdot \sigma(u, p) = f \quad \text{in } \Omega\times]0, T[,$$
$$\operatorname{div} u = 0 \quad \text{in } \Omega\times]0, T[,$$

$$u(t=0) = u_0 \quad \text{in } \Omega,$$
$$\sigma(u,p)\mathbf{n} = g \quad \text{on } \Gamma\times]0,T[.$$

Note that assuming a *smooth* boundary is rather natural in the case of free boundary problems, where surface tension usually tends to smoothen the interface. In the following we skip the nonlinear term, as our focus is the error introduced by time discretization and splitting. Thus, the time-dependent Stokes equations are considered: find u and p fulfilling

$$\partial_t u - \nabla\cdot\sigma(u,p) = f \quad \text{in } \Omega\times]0,T[, \tag{1.1a}$$
$$\operatorname{div} u = 0 \quad \text{in } \Omega\times]0,T[, \tag{1.1b}$$
$$u(t=0) = u_0 \quad \text{in } \Omega, \tag{1.1c}$$
$$\sigma(u,p)\mathbf{n} = g \quad \text{on } \Gamma\times]0,T[. \tag{1.1d}$$

In what follows Sobolev spaces $H^m(\Omega) = H^{m,2}(\Omega)$ $(m = 0,1,\ldots)$ will be used whose norms are denoted by $\|\cdot\|_m$. Moreover, $H_0^1(\Omega)$ denotes the subspace of all functions in $H^1(\Omega)$ with vanishing traces on Γ. At some places we also make use of fractional order Sobolev spaces $H^s(\Omega)$, $s\in\mathbb{R}$, see for instance [14,15]. For a set G the norm and inner product of $L^2(G) = H^0(G)$ are denoted by $\|\cdot\|_G$ and $(\cdot,\cdot)_G$, respectively. If $G = \Omega$, the subscript will be omitted.

Finally, we write $a \lesssim b$ for two functions or quantities a, b, whenever there is a generic constant C, such that $a \le C\,b$.

2. Some differential geometry

In the sequel a bit of differential geometry is needed. The results used here are all classical and may be found in any textbook on differential geometry. We adopt the notation of the nice presentation in [16].

Let $\mathbf{n}$ denote the outer normal to Γ. Note that $\mathbf{n}$ can be extended to a small neighborhood of Γ, constant in normal direction. We define the tangential projection $P := \mathbf{I} - \mathbf{n}\otimes\mathbf{n}$. With the help of P the tangential gradient of a smooth function ϕ given in a neighborhood of Γ is defined by $\nabla_\Gamma\phi := P\nabla\phi$ and for a vector field $\mathbf{F}$, $\nabla_\Gamma\mathbf{F} := (\nabla\mathbf{F})P$, i.e.$(\nabla_\Gamma\mathbf{F})_{ij} = (\partial_k\mathbf{F}_i)P_{kj}$. The tangential divergence $\nabla_\Gamma\cdot\mathbf{F}$ is given by $\nabla_\Gamma\cdot\mathbf{F} := tr(\nabla_\Gamma\mathbf{F})$ and the *Laplace–Beltrami* operator by $\Delta_\Gamma := \nabla_\Gamma\cdot\nabla_\Gamma$.

The *second fundamental form* H is defined as $H := \nabla\mathbf{n} = \nabla_\Gamma\mathbf{n}$. The tensor H is symmetric and $tr(H) = \nabla_\Gamma\cdot\mathbf{n} = \kappa$ with κ the sum of the principle curvatures.

Alternatively, one can equivalently define all these quantities in an intrinsic way.

The following lemma relates the Cartesian Laplacian and the Laplace–Beltrami operator.

Lemma 2.1. *Let ϕ be a smooth function, defined in a neighborhood of Γ. Then the following representation holds on Γ:*

$$\triangle\phi = \Delta_\Gamma\phi + \partial_{\mathbf{nn}}\phi + \partial_{\mathbf{n}}\phi\kappa,$$

where $\partial_{\mathbf{n}} = \mathbf{n}\cdot\nabla$ denotes the first and $\partial_{\mathbf{nn}}$ the second derivative in normal direction, respectively.

Proof.

$$\begin{aligned}\triangle\phi &= tr(\nabla\nabla\phi) = tr(\nabla(\nabla_\Gamma\phi)) + tr(\nabla(\partial_{\mathbf{n}}\phi\mathbf{n}))\\ &= tr(\nabla(\nabla_\Gamma\phi)P) + \underbrace{tr(\nabla(\nabla_\Gamma\phi)\mathbf{n}\otimes\mathbf{n})}_{=:(I)} + tr(\nabla(\partial_{\mathbf{n}}\phi\mathbf{n})) = (*).\end{aligned}$$

Writing the term (I) in coordinates and using the convention that repeated indices are summed up from 1 to d one gets

$$(I) = \partial_k(\nabla_\Gamma\phi)_i\mathbf{n}_k\mathbf{n}_i = \partial_k(P_{il}\partial_l\phi)\mathbf{n}_k\mathbf{n}_i = \partial_{\mathbf{n}}(P_{il}\partial_l\phi)\mathbf{n}_i = P_{il}(\partial_{\mathbf{n}l}\phi)\mathbf{n}_i = \underbrace{(\mathbf{n}_iP_{il})}_{=0}\partial_{\mathbf{n}l}\phi = 0,$$

where we have used the fact that $\partial_{\mathbf{n}}\mathbf{n} = 0$ and thus $\partial_{\mathbf{n}}(P\psi) = P\partial_{\mathbf{n}}\psi$ for a function ψ. Then

$$(*) = \nabla_\Gamma\cdot\nabla_\Gamma\phi + tr(\nabla(\partial_{\mathbf{n}}\phi)\otimes\mathbf{n}) + \underbrace{tr(\partial_{\mathbf{n}}\phi\nabla\mathbf{n})}_{=\partial_{\mathbf{n}}\phi\nabla\cdot\mathbf{n}} = \Delta_\Gamma\phi + \partial_{\mathbf{nn}}\phi + \partial_{\mathbf{n}}\phi\kappa. \quad \square$$

Lemma 2.2 (*Integration by Parts*)**.** *Let Γ be smooth and closed. For any smooth vector field* $\mathbf{F}$ *on* Γ *the following identity holds:*

$$\int_\Gamma \nabla_\Gamma \cdot \mathbf{F} = \int_\Gamma \kappa \mathbf{F} \cdot \mathbf{n}.$$

Proof. See [16]. □

Also the following formula will be needed.

Lemma 2.3. *Let ϕ be a smooth function, defined in a neighborhood of Γ. On Γ the following relation holds:*

$$\partial_\mathbf{n} \nabla \phi = \nabla_\Gamma \partial_\mathbf{n} \phi - H \nabla_\Gamma \phi + \partial_{\mathbf{nn}} \phi\, \mathbf{n}.$$

Proof.

$$\partial_\mathbf{n} \nabla \phi = P \partial_\mathbf{n} \nabla \phi + \mathbf{n} \otimes \mathbf{n}\, \partial_\mathbf{n} \nabla \phi = P \partial_\mathbf{n} \nabla \phi + \partial_{\mathbf{nn}} \phi \mathbf{n},$$

because $\partial_\mathbf{n} \mathbf{n} = 0$. Writing $\partial_\mathbf{n} \nabla \phi$ in coordinates one gets

$$\partial_\mathbf{n} \partial_i \phi = \mathbf{n}_k \partial_k \partial_i \phi = \mathbf{n}_k \partial_i \partial_k \phi = \partial_i (\mathbf{n}_k \partial_k \phi) - (\partial_i \mathbf{n}_k) \partial_k \phi.$$

In other words:

$$\partial_\mathbf{n} \nabla \phi = \nabla \partial_\mathbf{n} \phi - H \nabla \phi$$

from which it immediately follows

$$P \partial_\mathbf{n} \nabla \phi = P \nabla \partial_\mathbf{n} \phi - P H \nabla \phi.$$

Observing that $H = PH = HP$ the result is proved. □

3. Pressure correction scheme for traction boundary condition

In this section we first describe the pressure correction scheme in its rotational form. The usual and natural way to impose a traction boundary condition turns out to give only poor results, in particular for the pressure. Instead, we modify the scheme by adding a *traction correction* to the boundary condition that leads to an improvement for the solution (u, p). A variational formulation is given that is the basis for the subsequent discretization in space by finite elements. Since integration by parts is used at several places, recall that $\Gamma = \partial \Omega$ and is thus closed.

3.1. Classical pressure correction scheme

An implicit, fully coupled discretization in time of system (1.1). can be formulated as follows: given the values $u^{-1}, u^0, \ldots, u^n$ for $n \geq 0$ compute u^{n+1}, p^{n+1} fulfilling

$$\frac{\alpha u^{n+1} + \beta u^n + \gamma u^{n-1}}{\Delta t} - \nabla \cdot \sigma(u^{n+1}, p^{n+1}) = f^{n+1} \quad \text{in } \Omega, \tag{3.1a}$$

$$\nabla \cdot u^{n+1} = 0 \quad \text{in } \Omega, \tag{3.1b}$$

$$\sigma(u^{n+1}, p^{n+1}) \mathbf{n} = g^{n+1} \quad \text{on } \Gamma \tag{3.1c}$$

with $f^{n+1} := f(t_{n+1}, \cdot) = f((n+1)\Delta t, \cdot)$ and $g^{n+1} = g(t_{n+1}, \cdot)$. Here, we have used a 2-step method given by α, β, γ for time discretization and a fixed time step size Δt for convenience. Choosing $\alpha = 1, \beta = -1, \gamma = 0$ one gets the implicit Euler scheme, while the choice $\alpha = 3/2, \beta = -2, \gamma = 1/2$ yields the backward difference formula BDF2 that is of second order and is strongly A-stable. However, any other reasonable choice for a time discretization scheme and/or variable time steps would be equally suited.

In contrast to the coupled scheme the natural pressure correction scheme in *rotation form* takes the form (see [5]): for $n \geq 0$, let $u^{-1}, u^0, \dots, u^n$ and $p^0, \dots, p^n$ be already computed. Then determine $\tilde{u}^{n+1}, u^{n+1}, p^{n+1}$ by

$$\frac{\alpha\tilde{u}^{n+1} + \beta u^n + \gamma u^{n-1}}{\Delta t} - \nabla \cdot \sigma(\tilde{u}^{n+1}, p^n) = f^{n+1} \quad \text{in } \Omega, \tag{3.2a}$$

$$\sigma(\tilde{u}^{n+1}, p^n)\mathbf{n} = g^{n+1} \quad \text{on } \Gamma, \tag{3.2b}$$

$$\alpha \frac{u^{n+1} - \tilde{u}^{n+1}}{\Delta t} + \nabla \Phi^{n+1} = 0 \quad \text{in } \Omega, \tag{3.2c}$$

$$\nabla \cdot u^{n+1} = 0 \quad \text{in } \Omega, \tag{3.2d}$$

$$\Phi^{n+1} = 0 \quad \text{on } \Gamma, \tag{3.2e}$$

$$p^{n+1} = p^n + \Phi^{n+1} - \frac{2}{\mathrm{Re}} \nabla \cdot \tilde{u}^{n+1} \quad \text{in } \Omega. \tag{3.2f}$$

The last term $\frac{2}{\mathrm{Re}} \nabla \cdot \tilde{u}^{n+1}$ on the right hand side in the definition of p^{n+1} (3.2f) yields the so-called *rotational form* of the pressure correction scheme introduced in [17] and analyzed for instance in [18].

Steps (3.2c)–(3.2e) are equivalent to and realized by:

$$\triangle \Phi^{n+1} = \frac{\alpha}{\Delta t} \nabla \cdot \tilde{u}^{n+1} \quad \text{in } \Omega, \tag{3.3a}$$

$$\Phi^{n+1} = 0 \quad \text{on } \Gamma, \tag{3.3b}$$

$$u^{n+1} = \tilde{u}^{n+1} - \frac{\Delta t}{\alpha} \nabla \Phi^{n+1} \quad \text{in } \Omega. \tag{3.3c}$$

Note that (3.2c)–(3.2e) implies that $u^{n+1} = P_H \tilde{u}^{n+1}$ with P_H the orthogonal projection onto H in the *Hodge decomposition*

$$L^2(\Omega)^d = \mathbf{H} \perp \mathbf{H}^\perp$$

with

$$\mathbf{H} = \{v \in L^2(\Omega)^d \mid \operatorname{div} v = 0\},$$
$$\mathbf{H}^\perp = \{\nabla\chi \mid \chi \in H_0^1(\Omega)\}.$$

This decomposition is the appropriate functional setting for the Stokes equations with traction boundary condition in contrast to the *Helmholtz decomposition* for the case with Dirichlet boundary values.

Let us establish some basic relations for the above scheme (see also [5]). Since $\nabla \times \nabla \times = -\triangle + \nabla \operatorname{div}$ and since by (3.2c) $\nabla \times \nabla \times \tilde{u}^{n+1} = \nabla \times \nabla \times u^{n+1}$ we have

$$\nabla \cdot D(u^{n+1}) = \triangle u^{n+1} = \triangle \tilde{u}^{n+1} - \nabla\nabla \cdot \tilde{u}^{n+1} = \nabla \cdot D(\tilde{u}^{n+1}) - 2\nabla\nabla \cdot \tilde{u}^{n+1}. \tag{3.4}$$

Then taking the sum of (3.2a) and (3.2c), taking into account (3.2f), one gets

$$\partial_t^{\Delta t} u^{n+1} - \nabla \cdot \sigma(u^{n+1}, p^{n+1}) = f^{n+1} \quad \text{in } \Omega \tag{3.5}$$

with $\partial_t^{\Delta t} u^{n+1} := \frac{\alpha u^{n+1} + \beta u^n + \gamma u^{n-1}}{\Delta t}$. This explains the superior behavior of the rotational form of the pressure corrections scheme: up to the wrong boundary condition, (u^{n+1}, p^{n+1}) fulfills a time discrete, implicit, coupled equation.

On the other hand, it is possible to completely eliminate the variable u^{n+1} from the computation, which is advantageous in a finite element context, since u^{n+1} is not a standard finite element function. To this end, shift the indices in (3.2a) back by -1 and -2, respectively, and substitute u^n, u^{n-1} by the corresponding expressions for $\tilde{u}$ to get:

$$\partial_t^{\Delta t} \tilde{u}^{n+1} - \nabla \cdot \sigma\left(\tilde{u}^{n+1}, p^n - \frac{\beta}{\alpha}\Phi^n - \frac{\gamma}{\alpha}\Phi^{n-1}\right) = f^{n+1} \quad \text{in } \Omega. \tag{3.6}$$

3.2. Traction correction

In [9] it was pointed out that a Dirichlet boundary condition for Φ is necessary for stability, which is also observed numerically. On the other hand, then the natural boundary condition for $\tilde{u}^{n+1}$ (3.2b)

$$\sigma(\tilde{u}^{n+1}, p^n)\mathbf{n} = g^{n+1} \quad \text{on } \Gamma$$

yields an inconsistent boundary condition for (u^{n+1}, p^{n+1}), which is reflected in an order reduction for the error, both theoretically as well as in computational experiments. More precisely, in [9] the following error estimates were shown (in case the problem enjoys full Stokes regularity):

$$\left(\sum_n \Delta t \|u(t_n) - \tilde{u}^n\|^2\right)^{1/2} \lesssim \Delta t^{3/2},$$

$$\left(\sum_n \Delta t \|\nabla(u(t_n) - \tilde{u}^n)\|^2 + \sum_n \Delta t \|p(t_n) - p^n\|^2\right)^{1/2} \lesssim \Delta t.$$

See also Section 4 for a computational result. The reason behind this poor behavior is the stronger coupling of velocity and pressure by the traction boundary condition compared to the case of Dirichlet boundary values and the homogeneous Dirichlet boundary condition for the pressure correction Φ. In order to get a more consistent boundary condition we modify the above relation by introducing a yet to be determined functional l^{n+1}:

$$\sigma(\tilde{u}^{n+1}, p^n)\mathbf{n} = g^{n+1} + l^{n+1} \quad \text{on } \Gamma. \tag{3.7}$$

Inserting the expressions for u^{n+1} and p^{n+1} in terms of $\tilde{u}^{n+1}$, Φ^{n+1}, p^n and assuming $\Phi^{n+1}_{|\Gamma} = 0$ one computes

$$\begin{aligned}
\sigma(u^{n+1}, p^{n+1})\mathbf{n} &= \frac{1}{\mathrm{Re}} D(u^{n+1})\mathbf{n} - p^{n+1}\mathbf{n} \\
&= \frac{1}{\mathrm{Re}} D(\tilde{u}^{n+1})\mathbf{n} - \frac{2\Delta t}{\alpha \mathrm{Re}} \partial_{\mathbf{n}} \nabla \Phi^{n+1} - \left(p^n + \Phi^{n+1} - \frac{2}{\mathrm{Re}} \nabla \cdot \tilde{u}^{n+1}\right)\mathbf{n} \\
&= g^{n+1} + l^{n+1} - \frac{2\Delta t}{\alpha \mathrm{Re}} \partial_{\mathbf{n}} \nabla \Phi^{n+1} + \frac{2}{\mathrm{Re}} \nabla \cdot \tilde{u}^{n+1}\mathbf{n}.
\end{aligned} \tag{3.8}$$

On Γ one has the identity

$$\begin{aligned}
\partial_{\mathbf{n}} \nabla \Phi^{n+1} &\overset{\text{(Lemma 2.3)}}{=} \nabla_\Gamma \partial_{\mathbf{n}} \Phi^{n+1} + \partial_{\mathbf{nn}} \Phi^{n+1}\mathbf{n} - \underbrace{H \nabla_\Gamma \Phi^{n+1}}_{=0} \\
&\overset{\text{(Lemma 2.1)}}{=} \nabla_\Gamma \partial_{\mathbf{n}} \Phi^{n+1} + \big(\triangle \Phi^{n+1} - \underbrace{\Delta_\Gamma \Phi^{n+1}}_{=0} - \partial_{\mathbf{n}} \Phi^{n+1} \kappa\big)\mathbf{n} \\
&= \nabla_\Gamma \partial_{\mathbf{n}} \Phi^{n+1} + \frac{\alpha}{\Delta t} \nabla \cdot \tilde{u}^{n+1}\mathbf{n} - \partial_{\mathbf{n}} \Phi^{n+1} \kappa \mathbf{n}.
\end{aligned}$$

Using the above identity in (3.8) one finds

$$\begin{aligned}
\sigma(u^{n+1}, p^{n+1})\mathbf{n} &= g^{n+1} + l^{n+1} - \frac{2\Delta t}{\alpha \mathrm{Re}} \partial_{\mathbf{n}} \nabla \Phi^{n+1} + \frac{2}{\mathrm{Re}} \nabla \cdot \tilde{u}^{n+1}\mathbf{n} \\
&= g^{n+1} + l^{n+1} + \frac{2\Delta t}{\alpha \mathrm{Re}} \big(-\nabla_\Gamma \partial_{\mathbf{n}} \Phi^{n+1} + \partial_{\mathbf{n}} \Phi^{n+1} \kappa \mathbf{n}\big).
\end{aligned}$$

Thus we have the following astonishing result.

Proposition 3.1. *If l^{n+1} is defined by*

$$l^{n+1} = \frac{2\Delta t}{\alpha \mathrm{Re}} \big(\nabla_\Gamma \partial_{\mathbf{n}} \Phi^{n+1} - \partial_{\mathbf{n}} \Phi^{n+1} \kappa \mathbf{n}\big),$$

then by (3.5) Scheme (3.2) is equivalent to an implicit, coupled discretization for (u^{n+1}, p^{n+1}).

Unfortunately, the above scheme needs Φ^{n+1} in order to compute $\tilde{u}^{n+1}$. This, however, is not in the spirit of projection schemes, since it would require an expensive iteration procedure. A natural way out is to use an extrapolation of Φ^{n+1} in the definition of l^{n+1}, i.e.

$$l^{n+1} := \frac{2\Delta t}{\alpha \mathrm{Re}}\big(\nabla_\Gamma \partial_{\mathbf{n}} \Phi^* - \partial_{\mathbf{n}} \Phi^* \kappa \mathbf{n}\big).$$

There are several reasonable choices for Φ^*:

$$\Phi^* := 2\Phi^n - \Phi^{n-1},$$
$$\Phi^* := \Phi^n,$$
$$\Phi^* := (\Phi^n + \Phi^{n-1})/2.$$

The first choice is a second order extrapolation for Φ^{n+1}, which unfortunately turns out to be unstable. However, heuristically it is sufficient to have a first order extrapolation of Φ^{n+1} in order to get overall second order accuracy (in case of BDF2), since l^{n+1} itself is a second order correction to the scheme. Thus, the second choice might be used. Computational experiments led to the third choice that gave slightly better results. Let us further note that with one of the latter two choices we never experienced any stability problems.

3.3. Variational formulation

In this section we derive a variational formulation of the pressure correction scheme (in the variant of (3.6)) with traction correction. To this end we introduce the spaces $X := H^1(\Omega)^d$, $Y := L^2(\Omega)$ and $W := H^1_0(\Omega)$.

Upon multiplying (3.6) by a test function $v \in X$, integrating by parts and taking into account boundary condition (3.7) one arrives at:

$$(\partial_t^{\Delta t}\tilde{u}^{n+1}, v) + \frac{1}{2\mathrm{Re}}(D(\tilde{u}^{n+1}), D(v)) - \left(p^n - \frac{\beta}{\alpha}\Phi^n - \frac{\gamma}{\alpha}\Phi^{n-1}, \operatorname{div} v\right)$$
$$= (f^{n+1}, v) + (g^{n+1}, v)_\Gamma + (l^{n+1}, v)_\Gamma.$$

Integrating by parts, the expression for l^{n+1} can be simplified to:

$$\begin{aligned}\int_\Gamma l^{n+1}\cdot v &= \frac{2\Delta t}{\alpha \mathrm{Re}}\int_\Gamma (\nabla_\Gamma \partial_{\mathbf{n}}\Phi^* - \partial_{\mathbf{n}}\Phi^*\kappa\mathbf{n})\cdot v \\ &= \frac{2\Delta t}{\alpha \mathrm{Re}}\int_\Gamma \big(\nabla_\Gamma\cdot(\partial_{\mathbf{n}}\Phi^* v) - \partial_{\mathbf{n}}\Phi^*\nabla_\Gamma\cdot v - \partial_{\mathbf{n}}\Phi^*\kappa v\cdot\mathbf{n}\big) \\ &= -\frac{2\Delta t}{\alpha \mathrm{Re}}\int_\Gamma \partial_{\mathbf{n}}\Phi^*\nabla_\Gamma\cdot v,\end{aligned}$$

where the last step follows from the integration by parts formula Lemma 2.2 with $\mathbf{F} = \partial_{\mathbf{n}}\Phi^* v$. Note that thanks to this form of l^{n+1} it is not necessary to compute the curvature of Γ.

Thus the pressure correction scheme with traction correction can be written in the following variational form: for $n \geq 0$, let $\tilde{u}^{-1}, \tilde{u}^0, \ldots, u^n, p^0, \ldots, p^n$ and $0 := \Phi^{-1} = \Phi^0, \Phi^1, \ldots, \Phi^n$ be already computed. Then determine $\tilde{u}^{n+1} \in X$, $\Phi^{n+1} \in W$, $p^{n+1} \in Y$ by

$$(\partial_t^{\Delta t}\tilde{u}^{n+1}, v) + \frac{1}{2\mathrm{Re}}(D(\tilde{u}^{n+1}), D(v)) - \left(p^n - \frac{\beta}{\alpha}\Phi^n - \frac{\gamma}{\alpha}\Phi^{n-1}, \operatorname{div} v\right)$$
$$= (f^{n+1}, v) + (g^{n+1}, v)_\Gamma - \frac{2\Delta t}{\alpha \mathrm{Re}}\int_\Gamma \partial_{\mathbf{n}}\Phi^*\nabla_\Gamma\cdot v \quad \text{for all } v \in X, \tag{3.9a}$$

$$(\nabla\Phi^{n+1}, \nabla\psi) = -\frac{\alpha}{\Delta t}(\nabla\cdot\tilde{u}^{n+1}, \psi) \quad \text{for all } \psi \in W \tag{3.9b}$$

and

$$p^{n+1} = p^n + \Phi^{n+1} - \frac{2}{\mathrm{Re}}\nabla\cdot\tilde{u}^{n+1} \quad \text{in } Y. \tag{3.9c}$$

If $l^{n+1} \in X^*$, the above scheme is well defined in the respective spaces. Indeed, if $\tilde{u}^n, \tilde{u}^{n-1} \in X$, $p^n \in Y$, $\Phi^n, \Phi^{n-1} \in W$ then by (3.9a) $\tilde{u}^{n+1} \in X$. Furthermore, by (3.9b) $\Phi^{n+1} \in W$ and finally $p^{n+1} \in Y$.

It remains to show that $l^{n+1} \in X^*$. Let $\tilde{u}^* \in X$ denote either $\tilde{u}^* = \tilde{u}^n$ or $\tilde{u}^* = (\tilde{u}^n + \tilde{u}^{n-1})/2$ depending on the choice of Φ^*. Now, $\nabla \cdot \tilde{u}^* \in L^2(\Omega)$ and then by regularity (since Γ is assumed to be smooth) $\Phi^* \in H^2(\Omega)$ fulfilling the estimate

$$\|\Phi^*\|_2 \lesssim \frac{\alpha}{\Delta t}\|\nabla \cdot \tilde{u}^*\|.$$

From this one infers $\partial_{\mathbf{n}}\Phi^* = \mathbf{n} \cdot \mathrm{tr}_\Gamma \nabla \Phi^* \in H^{1/2}(\Gamma)$ with

$$\|\partial_{\mathbf{n}}\Phi^*\|_{1/2,\Gamma} \lesssim \|\Phi^*\|_2 \lesssim \frac{\alpha}{\Delta t}\|\nabla \cdot \tilde{u}^*\|.$$

Using now Lemma 3.3 below one concludes

$$\left|\int_\Gamma \partial_{\mathbf{n}}\Phi^* \nabla_\Gamma \cdot v\right| \lesssim \|\partial_{\mathbf{n}}\Phi^*\|_{1/2,\Gamma}\|\nabla_\Gamma \cdot v\|_{-1/2,\Gamma} \lesssim \|\partial_{\mathbf{n}}\Phi^*\|_{1/2,\Gamma}\|v\|_{1/2,\Gamma} \lesssim \|\partial_{\mathbf{n}}\Phi^*\|_{1/2,\Gamma}\|v\|_{1,\Omega}.$$

With the help of the previous estimate this yields

$$|\langle l^{n+1}, v\rangle| = \frac{2\Delta t}{\alpha \mathrm{Re}}\left|\int_\Gamma \partial_{\mathbf{n}}\Phi^* \nabla_\Gamma \cdot v\right| \lesssim \frac{1}{\mathrm{Re}}\|\nabla \cdot \tilde{u}^*\| \, \|v\|_{1,\Omega}$$

for all $v \in X$.

Using Korn's inequality we thus have proved:

Proposition 3.2. *If Γ is smooth, $f^{n+1} \in X^*$, $g^{n+1} \in H^{-1/2}(\Gamma)$ for all $n \geq 0$ and l^{n+1} is defined via $\Phi^* = \Phi^n$ or $\Phi^* = (\Phi^n + \Phi^{n-1})/2$, then for all $n \geq 0$ Scheme (3.9) admits unique solutions $\tilde{u}^{n+1} \in X$, $p^{n+1} \in Y$, $\Phi^{n+1} \in W$.*

Lemma 3.3. *Define the bilinear form*

$$b(\rho, v) := \int_\Gamma \rho \nabla_\Gamma \cdot v$$

for smooth functions ρ, v, v being vector valued. Then $b(\cdot,\cdot)$ can be uniquely extended to functions $\rho \in H^{1/2}(\Gamma)$, $v \in H^{1/2}(\Gamma)^d$ fulfilling the estimate

$$|b(\rho, v)| \lesssim \|\rho\|_{1/2,\Gamma}\|v\|_{1/2,\Gamma}.$$

Proof. See Appendix. □

3.4. Finite element implementation

With the help of the variational formulation from the previous section, a finite element formulation for the pressure correction traction correction scheme is now at hand. Let $X_h \subseteq H^1(\Omega)^d$, $Y_h \subseteq H^1(\Omega)$ be an inf–sup stable pair of elements with globally continuous pressures corresponding to a conforming triangulation $\mathcal{T}_h$. We chose the $\mathcal{P}_2 - \mathcal{P}_1$ Taylor–Hood element for the computational results in Section 4 below, but any other choice would do. Furthermore let $W_h := Y_h \cap H_0^1(\Omega)$ and $\mathcal{N}_h := \{\phi_h : \Gamma \mapsto \mathbb{R} \mid \phi_h = \mathrm{tr}_\Gamma q_h,\ q_h \in Y_h\}$, where tr_Γ is the trace of a function in $Y_h \subseteq H^1(\Omega)$ on Γ.

The fully discrete pressure correction scheme with traction correction now reads: for $n \geq 0$, let $\tilde{u}_h^{-1}$, $\tilde{u}_h^0, \ldots, u_h^n, p_h^0, \ldots, p_h^n$ and $0 := \Phi_h^{-1} = \Phi_h^0, \Phi_h^1, \ldots, \Phi_h^n$ be already computed. Then determine $\tilde{u}_h^{n+1} \in X_h$, $\Phi_h^{n+1} \in W_h$, $p_h^{n+1} \in Y_h$ by

$$\begin{aligned}(\partial_t^{\Delta t}\tilde{u}_h^{n+1}, v_h) &+ \frac{1}{2\mathrm{Re}}(D(\tilde{u}_h^{n+1}), D(v_h)) - \left(p_h^n - \frac{\beta}{\alpha}\Phi_h^n - \frac{\gamma}{\alpha}\Phi_h^{n-1}, \operatorname{div} v_h\right)\\ &= (f^{n+1}, v_h) + (g^{n+1}, v_h)_\Gamma + \langle l_h^{n+1}, v_h\rangle_\Gamma \quad \text{for all } v_h \in X_h,\end{aligned} \tag{3.10a}$$

$$(\nabla \Phi_h^{n+1}, \nabla \psi_h) = -\frac{\alpha}{\Delta t}(\nabla \cdot \tilde{u}_h^{n+1}, \psi_h) \quad \text{for all } \psi_h \in W_h \tag{3.10b}$$

and

$$(p_h^{n+1}, q_h) = \left(p_h^n + \Phi_h^{n+1} - \frac{2}{\mathrm{Re}}\nabla \cdot \tilde{u}_h^{n+1}, q_h\right) \quad \text{for all } q_h \in Y_h. \tag{3.10c}$$

In (3.10a) l_h^{n+1} denotes an approximation of l^{n+1}. It is computed in the following way. First, we need to approximate the derivative of Φ^*. This can be accomplished in a variational way. To this end, let $\rho_h \in \mathcal{N}_h$ denote the approximation to $\partial_{\mathbf{n}} \Phi^*$. Determine ρ_h by

$$(\rho_h, \mathrm{tr}_\Gamma q_h)_\Gamma + \epsilon(\nabla_\Gamma \rho_h, \nabla_\Gamma \mathrm{tr}_\Gamma q_h)_\Gamma = \int_\Omega \nabla \Phi_h^* \cdot \nabla q_h + \frac{\alpha}{\Delta t}\int_\Omega \nabla \cdot \tilde{u}_h^* q_h \quad \text{for all } q_h \in Y_h. \tag{3.11}$$

Here, $\tilde{u}_h^*$ denotes either $\tilde{u}_h^* = \tilde{u}_h^n$ or $\tilde{u}_h^* = (\tilde{u}_h^n + \tilde{u}_h^{n-1})/2$ depending on the choice of Φ_h^*. $\epsilon > 0$ is a small parameter. The meaning of the term on the left hand side multiplied by ϵ is a smoothing by the discrete Laplace–Beltrami operator of the otherwise possibly wiggly ρ_h due to discretization effects, see also Section 4.

Note that (3.11) constitutes a linear system involving the mass and a stiffness matrix on the boundary triangulation $\mathcal{S}_h$ induced by the triangulation $\mathcal{T}_h$ of Ω. Thus $\mathcal{S}_h$ and in turn the space $\mathcal{N}_h$ can be rather easily realized. In our case $\mathcal{N}_h$ is the space of globally continuous, piecewise linear functions on Γ corresponding to the boundary triangulation $\mathcal{S}_h$. Since Γ is a hypersurface and thus $\mathcal{S}_h$ has much less degrees of freedom than $\mathcal{T}_h$, the computational cost to solve (3.11) is negligible.

Finally, l_h^{n+1} is defined by

$$\langle l_h^{n+1}, v_h\rangle := -\frac{2\Delta t}{\alpha \mathrm{Re}}\int_\Gamma \rho_h \nabla_\Gamma \cdot (\mathrm{tr}_\Gamma v_h) \quad \text{for all } v_h \in X_h. \tag{3.12}$$

Remark 3.4. Although its derivation is somehow involved, the final form of l^{n+1} is amazingly simple. Note that in 2D the above integral is given by

$$\int_\Gamma \rho_h \nabla_\Gamma \cdot (\mathrm{tr}_\Gamma v_h) = \int_\Gamma \rho_h \tau \cdot \partial_\tau v_h,$$

where τ is a tangential vector on Γ. Thus the evaluation of the integral is straightforward: looping over all triangles, one can access all boundary edges and then do the computations boundary edge by boundary edge. Thus, in the 2d case, each boundary edge is a straight line (if one does not use isoparametric elements, for which the computations are similar) and the tangential component and tangential derivative of a finite element function can be computed easily. In 3D this computation is more technical, but can be done in a similar fashion, see [19] for details.

4. Computational results

In this section the scheme with traction correction is computationally compared to the classical scheme. To this end let Ω be the ellipse $\Omega := \{(x_1, x_2) \mid (x_1/a)^2 + (x_2/b)^2 < 1\}$ with $a = 1/1.2$ and $b = 1.2$. As in [9] u and p are defined by

$$u(t, x) = [\sin(x_1)\sin(x_2 + t), \cos(x_1)\cos(x_2 + t)]^T,$$
$$p(t, x) = \cos(x_1)\sin(x_2 + t)/5 + 1.$$

The right hand sides f, g are chosen such that the above pair (u, p) is a solution to Eq. (1.1) for $\mathrm{Re} = 10$. In order to get close to the semi-discrete case, a fine grid consisting of $2 \times 97,201$ degrees of freedom for the velocity and $25,325$ degrees of freedom for the pressure was used. Consequently the error due to space discretization was very small. For time discretization the BDF2 scheme was used. The examples were computed with the solver Navier [20].

Fig. 1 shows error plots versus time step size. The velocity error in the H^1-norm is hardly influenced by the traction correction. In the L^2-norm there is no improvement in the order of convergence, but the quantitative error is noticeably smaller, see Fig. 2.

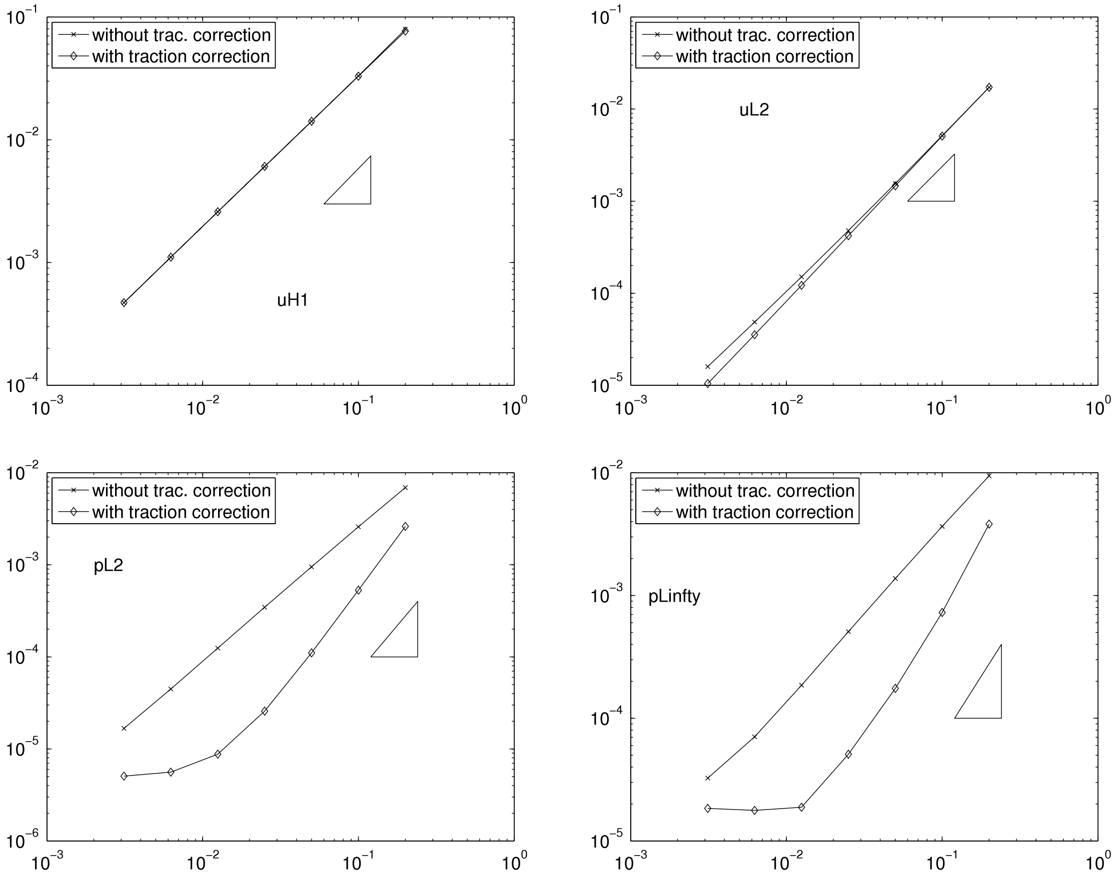

Fig. 1. Errors versus time step size for the scheme with and without traction correction; H^1, L^2-errors for the velocity, L^2, L^∞–errors for the pressure at time $t_n = 1.0$, respectively. Triangles have respective slopes 1.3, 1.7, 2.0, 2.0 (from top left to bottom right); $\epsilon = 10^{-3}$.

The pressure, however, is significantly improved by the new method, both in the $L^\infty(L^2)$-norm as well as in the $L^\infty(L^\infty)$-norm. Tables 1 and 2 show that the order of convergence for the pressure is improved from about 1.5 to 2.0. The stagnation of the pressure error for smaller time step sizes is caused by the space discretization error that eventually becomes dominant for very small values of Δt.

Fig. 3 shows the L^2–error for the pressure using different meshes (the *fine* one being the one described above). This proves that the stagnation in the error reduction is indeed caused by the space discretization.

Fig. 4 illustrates the influence of the method on the choice of ϵ. As can be seen, for large values as well as for very small values of ϵ the behavior of the method is quite poor. This becomes clear from Fig. 5: for very large values of ϵ, $\partial_{\mathbf{n}} \Phi^*$ is over-smoothed, while for very small values of $\epsilon \, \partial_{\mathbf{n}} \Phi^*$ is still wiggly.

The results of a quantitative test with various values of the mesh size h and of ϵ, respectively, are given in Table 3. As a measure for the oscillation of $\partial_{\mathbf{n}} \Phi_h^*$ and thus in turn for the needed regularization, we computed the quantity OSC, defined by

$$\mathrm{OSC} = \mathrm{OSC}(h, \epsilon) := \|\partial_{\mathbf{n}} \Phi_h^*(\epsilon) - \partial_{\mathbf{n}} \Phi_h^*(\epsilon_0)\|_{L^2(\Gamma)}, \tag{4.1}$$

where $\epsilon_0 := 1\mathrm{e} - 3$ corresponds to a quite regularized solution. As can been deduced from Table 3, the oscillation becomes smaller for smaller mesh sizes. Thus less regularization is needed for finer grids.

5. Conclusion

In this paper pressure correction schemes for the computational solution of the time dependent (Navier)–Stokes equations with *traction boundary* condition have been considered. We have introduced a finite element based method to improve the performance of the classical approach, outlined for instance in [9].

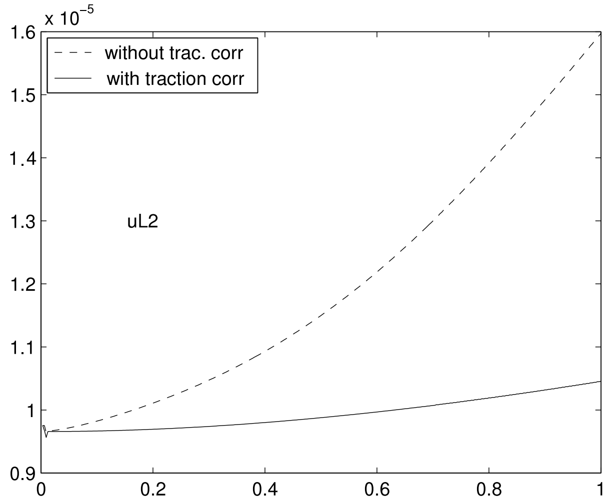

Fig. 2. Time evolution of the L^2 velocity errors for the scheme with and without traction correction; $\Delta t = 3.12510^{-3}$.

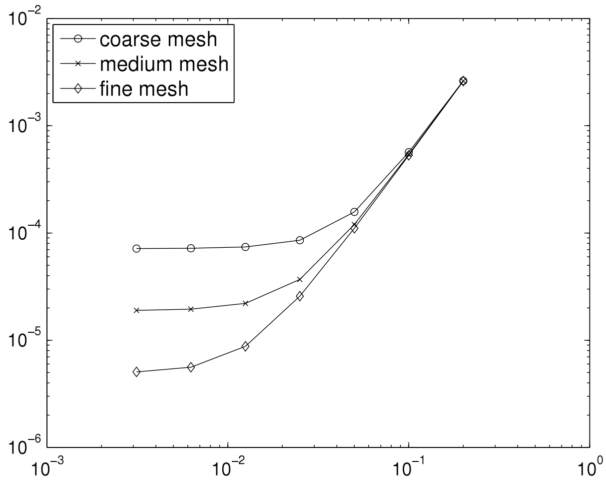

Fig. 3. L^2-errors for the pressure versus time step size for different meshes.

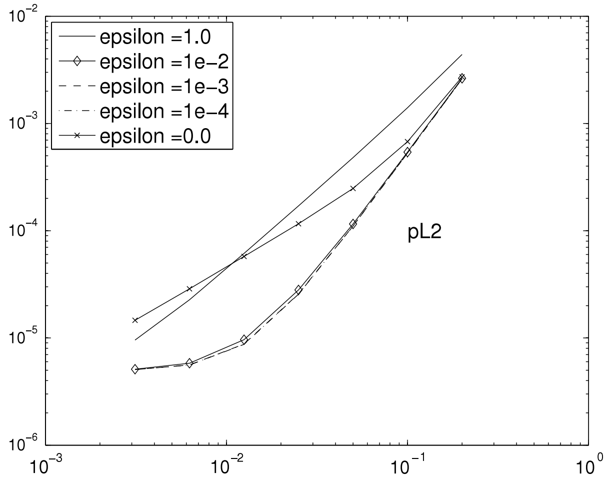

Fig. 4. L^2-errors for the pressure versus time step size for different choices of ϵ at time $t_n = 1.0$, $\Delta t = 0.0125$.

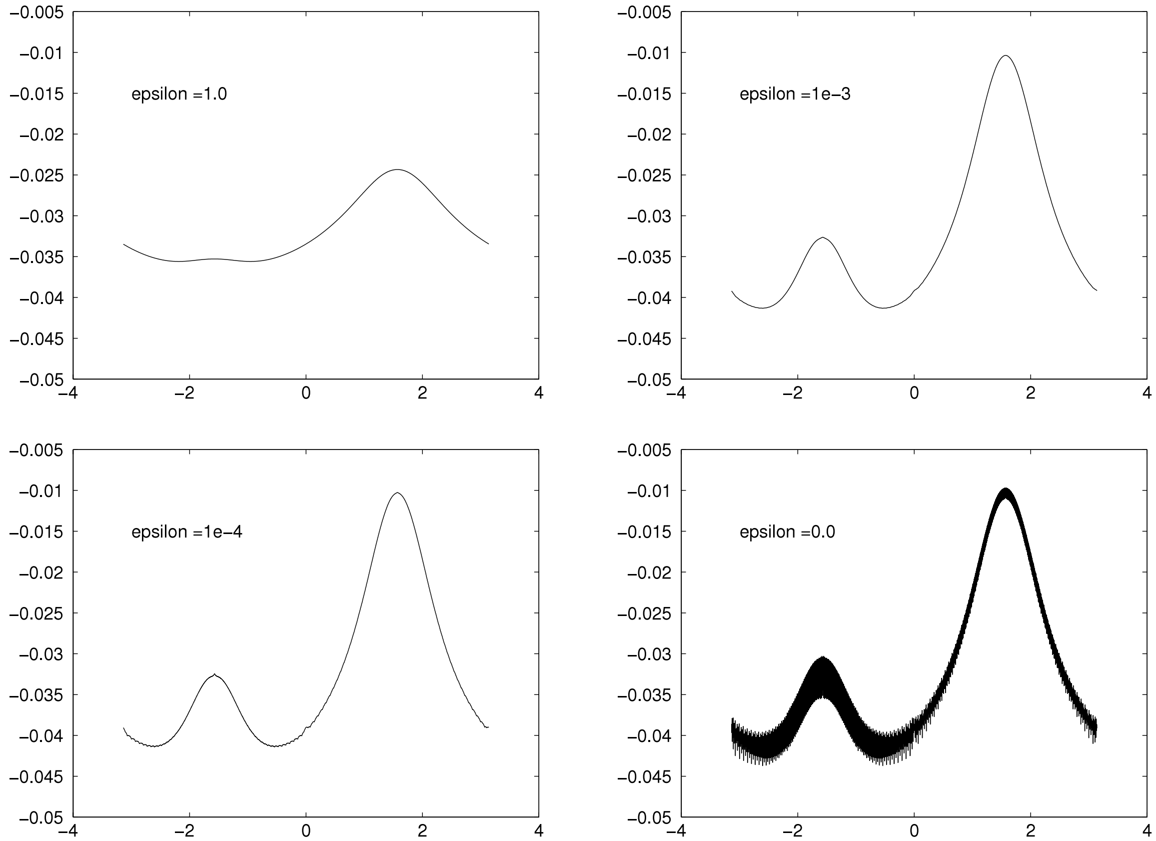

Fig. 5. Normal derivative $\partial_{\mathbf{n}} \Phi_h^*$ on Γ versus the azimuthal angle for different values of ϵ at time $t = 0.25$.

Table 1
Errors and experimental orders of convergence (EOC) for the scheme without and with traction correction; $\epsilon = 10^{-3}$; $E_p = \|p(t_n) - p^n\|$, $t_n = 1.0$.

Δt	Trac. corr.	E_p	EOC	Δt	Trac. corr.	E_p	EOC
0.2	no	6.8982e−03	–	0.2	yes	2.6139e−03	–
0.1	no	2.5875e−03	1.4147	0.1	yes	5.3031e−04	2.30130
0.05	no	9.5205e−04	1.4424	0.05	yes	1.1069e−04	2.26031
0.025	no	3.4577e−04	1.4612	0.025	yes	2.5763e−05	2.10315
0.0125	no	1.2460e−04	1.4725	0.0125	yes	8.7694e−06	1.55475
0.00625	no	4.4799e−05	1.4758	0.00625	yes	5.5988e−06	0.64736
0.003125	no	1.6730e−05	1.4210	0.003125	yes	5.0699e−06	0.14316

Table 2
L^2-errors in time and experimental orders of convergence (EOC) for the scheme without and with traction correction; $\epsilon = 10^{-3}$; $\tilde{E}_p = (\sum_n \Delta t \|p(t_n) - p^n\|^2)^{1/2}$.

Δt	Trac. corr.	$\tilde{E}_p$	EOC	Δt	Trac. corr.	$\tilde{E}_p$	EOC
0.2	no	6.6140e−03	–	0.2	yes	3.8197e−03	–
0.1	no	2.3499e−03	1.4929	0.1	yes	9.0861e−04	2.0717
0.05	no	8.3667e−04	1.4899	0.05	yes	2.1426e−04	2.0843
0.025	no	2.9765e−04	1.4910	0.025	yes	5.1540e−05	2.0556
0.0125	no	1.0579e−04	1.4924	0.0125	yes	1.3673e−05	1.9144
0.00625	no	3.7738e−05	1.4871	0.00625	yes	5.9854e−06	1.1918
0.003125	no	1.4161e−05	1.4141	0.003125	yes	5.0330e−06	0.2500

Table 3
Oscillation OSC, defined in Eq. (4.1), for various values of h and ϵ. Here, h_0 is the boundary mesh size of a rather coarse initial grid that is successively refined at the boundary ($t_n = 0.25$, $\Delta t = 0.0125$).

	$\epsilon = 1\mathrm{e}-7$	$\epsilon = 1\mathrm{e}-6$	$\epsilon = 1\mathrm{e}-5$
h_0	9.7445e−03	9.0924e−03	5.5216e−03
$h_0/2$	5.9016e−03	4.6531e−03	1.5958e−03
$h_0/4$	3.0774e−03	1.6191e−03	3.7214e−04
$h_0/8$	1.2671e−03	4.0770e−04	1.4434e−04

The improvement is accomplished by extrapolating the traction boundary condition in each time step. The corresponding term consists of a simple boundary functional involving the normal derivative of the pressure correction Φ and can be evaluated in a natural and easy way in the context of finite elements.

Computational results show a significant improvement of the solution, in particular for the pressure in the case of smooth domains.

Appendix

It remains to prove Lemma 3.3.

Proof. (Lemma 3.3)

The following proof relies on the techniques and results outlined in [15], Chapter 1.

Let ρ, v be smooth functions. First, it is not difficult to see that

$$\left|\int_\Gamma \rho \nabla_\Gamma \cdot v\right| \le \|\rho\|_{1/2,\Gamma} \|\nabla_\Gamma \cdot v\|_{-1/2,\Gamma} \lesssim \|\rho\|_{1/2,\Gamma} \|\nabla_\Gamma v\|_{-1/2,\Gamma}.$$

Since Γ is assumed to be smooth and compact, there are $V_1, \dots, V_m$ open in $\mathbb{R}^d$ covering Γ and smooth, invertible ϕ_j,

$$\phi_j : V_j \to \mathcal{D} := \{y \mid y = (y', y_n), -1 < y_n < 1\}$$

with the property that

$$\phi_{j\,|V_j\cap\Gamma} : V_j \cap \Gamma :\to \mathcal{D} \cap \{y_n = 0\},$$

i.e.

$$x \in \Gamma \cap V_j \Leftrightarrow x = \phi_j^{-1}(y', 0),\ (y', 0) \in \mathcal{D}.$$

Furthermore there is a smooth partition of unity α_j with compact support subordinate to the covering $V_1, \dots, V_m$. Any function u on Γ can thus be decomposed as

$$u = \sum_{j=1}^m \alpha_j u.$$

Define

$$\phi_j^*(\alpha_j u)(y') := (\alpha_j u) \circ \phi_j^{-1}(y', 0) \quad \text{for } (y', 0) \in \mathcal{D}.$$

For arbitrary $s \in \mathbb{R}$, $H^s(\Gamma)$ may now be defined by

$$H^s(\Gamma) := \{u \mid \phi_j^*(\alpha_j u) \in H^s(\mathbb{R}^{d-1}), j = 1, \dots, m\}$$

with corresponding norm

$$\|u\|_{s,\Gamma} := \left(\sum_{j=1}^m \|\phi_j^*(\alpha_j u)\|^2_{s,\mathbb{R}^{d-1}}\right)^{1/2}.$$

Using the intrinsic definition of ∇_Γ (see for instance [19]) one can write

$$(\nabla_\Gamma v)(x) = G_j(y')\nabla_{y'}(v \circ \phi_j)(y', 0) \quad \text{for } x = \phi_j^{-1}(y', 0) \in \Gamma$$

and with some tensor-valued function G_j depending on the local geometry of Γ. In other words:

$$\phi_j^*(\nabla_\Gamma v)(y') = G_j(y')\nabla_{y'}(v \circ \phi_j)(y', 0).$$

On the other hand, for a bounded C^1 function μ on $\mathbb{R}^{d-1}$ with bounded derivative the mapping Λ,

$$\Lambda : u \mapsto \mu \nabla_{y'} u,$$

is a continuous linear mapping both from

$$H^1(\mathbb{R}^{d-1}) \to L^2(\mathbb{R}^{d-1}) \quad \text{as well as}$$
$$L^2(\mathbb{R}^{d-1}) \to H^{-1}(\mathbb{R}^{d-1}).$$

Interpolating between $H^1(\mathbb{R}^{d-1})$ and $L^2(\mathbb{R}^{d-1})$ one therefore gets

$$\Lambda \in \mathcal{L}([H^1(\mathbb{R}^{d-1}), L^2(\mathbb{R}^{d-1})]_\theta, [L^2(\mathbb{R}^{d-1}), H^{-1}(\mathbb{R}^{d-1})]_\theta)$$

for all $0 < \theta < 1$, see [15]. In particular, choosing $\theta = 1/2$, one finds

$$\Lambda \in \mathcal{L}(H^{1/2}(\mathbb{R}^{d-1}), H^{-1/2}(\mathbb{R}^{d-1})).$$

Applying this result to $\mu = \alpha_j G_j$ the desired estimate follows. □

References

[1] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comp. 22 (1968) 745–762.
[2] A.J. Chorin, On the convergence of discrete approximations to the Navier–Stokes equations, Math. Comp. 23 (1969) 341–353.
[3] R. Temam, Sur l'approximation de la solution des équation de Navier–Stokes par la méthode des pas fractionaires ii, Arch. Ration. Mech. Anal. 33 (1969) 377–385.
[4] R. Rannacher, On Chorin's projection method for the incompressible Navier–Stokes equations, Lecture Notes in Math. 1530.
[5] J.L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Engrg. 195 (2006) 6011–6045.
[6] M.O. Bristeau, R. Glowinski, J. Périaux, Numerical methods for the Navier–Stokes equations. Applications to the simulation of compressible and incompressible viscous flows, in: Finite Elements in Physics (Lausanne, 1986), North-Holland, Amsterdam, 1987, pp. 73–187.
[7] R.L. Sani, P.M. Gresho, Résumé and remarks on the open boundary condition minisymposium, Internat. J. Numer. Methods Fluids 18 (10) (1994) 983–1008.
[8] J. Heywood, R. Rannacher, S. Turek, Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations, Internat. J. Numer. Methods Fluids 22 (5) (1996) 325–352.
[9] 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.
[10] J. Liu, Open and traction boundary conditions for the incompressible Navier-Stokes equations, J. Comput. Phys. 228 (19) (2009) 7250–7267.
[11] J.-G. Liu, J. Liu, R. Pego, Stability and convergence of efficient Navier-Stokes solvers via a commutator estimate, Comm. Pure Appl. Math. 60 (10) (2007) 1443–1487.
[12] J.-G. Liu, J. Liu, R. Pego, Stable and accurate pressure approximation for unsteady incompressible viscous flow, J. Comput. Phys. 229 (9) (2010) 3428–3453.
[13] A. Poux, S. Glockner, M. Azaïez, Improvements on open and traction boundary conditions for Navier-Stokes time-splitting methods, J. Comput. Phys. 230 (10) (2011) 4011–4027.
[14] R. Adams, Sobolev Spaces, Academic press, New York–London, 1975.
[15] J.L. Lions, E. Magenes, Non-homogeneous Boundary Value Problems and Applications, Springer, New York–Heidelberg, 1972.
[16] G.C. Buscaglia, R.F. Ausas, Variational formulations for surface tension, capillarity and wetting, Comput. Methods Appl. Mech. Engrg. 200 (2011) 3011–3025.
[17] L. Timmermans, P. Minev, F.V.D. Vosse, An approximate projection scheme for incompressible flow using spectral elements, Internat. J. Numer. Methods Fluids 22 (7) (1996) 673–688.
[18] J.L. Guermond, J. Shen, On the error estimates for the rotational pressure-correction projection methods, Math. Comp. 73 (248) (2004) 1719–1737. (electronic).
[19] E. Bänsch, Numerical methods for the instationary navier–stokes equations with a free capillary surface (Habilitation thesis), Univ. Freiburg, Germany, 1998.
[20] E. Bänsch, Simulation of instationary, incompressible flows, Acta Math. Univ. Comenian. 67 (1) (1998) 101–114.