Journal of Computational Physics 229 (2010) 3428–3453

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Stable and accurate pressure approximation for unsteady incompressible viscous flow

Jian-Guo Liu [a], Jie Liu [b], Robert L. Pego [c,*]

[a] Department of Physics and Mathematics, Duke University, Durham, NC 27708, United States
[b] Department of Mathematics, National University of Singapore, Singapore 117543, Singapore
[c] Department of Mathematical Sciences and Center for Nonlinear Analysis, Carnegie Mellon University, Pittsburgh, PA 15213, United States

ARTICLE INFO

Article history:
Received 24 September 2008
Received in revised form 8 November 2009
Accepted 11 January 2010
Available online 20 January 2010

MSC:
76D05
65M15
65M60

Keywords:
Time-dependent incompressible flow
Projection method
Time splitting
Third-order accuracy
Backward-facing step
Driven cavity
Flow past cylinder
Stokes pressure
Leray projection

ABSTRACT

How to properly specify boundary conditions for pressure is a longstanding problem for the incompressible Navier–Stokes equations with no-slip boundary conditions. An analytical resolution of this issue stems from a recently developed formula for the pressure in terms of the commutator of the Laplacian and Leray projection operators. Here we make use of this formula to (a) improve the accuracy of computing pressure in two kinds of existing time-discrete projection methods implicit in viscosity only, and (b) devise new higher-order accurate time-discrete projection methods that extend a slip-correction idea behind the well-known finite-difference scheme of Kim and Moin. We test these schemes for stability and accuracy using various combinations of C^0 finite elements. For all three kinds of time discretization, one can obtain third-order accuracy for both pressure and velocity without a time-step stability restriction of diffusive type. Furthermore, two kinds of projection methods are found stable using piecewise-linear elements for both velocity and pressure.

© 2010 Elsevier Inc. All rights reserved.

1. Introduction

We consider the Navier–Stokes equations (NSE) for incompressible fluid flow in a domain Ω in $\mathbb{R}^N$ ($N = 2$ or 3) with velocity specified on the boundary $\Gamma = \partial\Omega$. We write the momentum equation and boundary conditions in the form

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

$$\boldsymbol{u} = \boldsymbol{g} \quad \text{on } \Gamma. \tag{2}$$

Here $\boldsymbol{u}$ is the fluid velocity, p the pressure, and $\nu = 1/\mathrm{Re}$ is the kinematic viscosity coefficient, taken to be a fixed positive constant. We combine external forcing $\boldsymbol{f}$ and nonlinear terms into one symbol $\boldsymbol{F} = \boldsymbol{f} - \boldsymbol{u}\cdot\nabla\boldsymbol{u}$.

The pressure field p should ensure that the velocity is divergence-free, with

$$\nabla\cdot\boldsymbol{u} = 0 \quad \text{in } \Omega. \tag{3}$$

* Corresponding author. Tel.: +1 412 268 2553; fax: +1 412 268 6380.
E-mail addresses: Jian-Guo.Liu@duke.edu (J.-G. Liu), matlj@nus.edu.sg (J. Liu), rpego@cmu.edu (R.L. Pego).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.01.010

This incompressibility condition is the source of many difficulties associated with the numerical approximation of solutions of NSE, especially in the presence of boundaries. Projection methods, deriving from classic work of Chorin and Temam, aim to deal efficiently with incompressibility through strategies that involve the Helmholtz decomposition of an arbitrary vector field into a sum of a gradient plus a divergence-free field. But for many years, projection methods were plagued by large and poorly understood numerical boundary-layer errors.

The situation improved markedly with the formal analysis of Brown et al. [6], who explained why one could achieve second-order accuracy in time. One point we make in the present paper is that for no-slip boundary conditions, formal accuracy and numerical boundary-layer errors at the time-discrete level can be understood rather simply in terms of the failure of commutativity between the (space-continous) Laplacian Δ and the Leray-Helmholtz projection operator $\mathcal{P}$ onto divergence-free vector fields.

The commutator $\Delta\mathcal{P} - \mathcal{P}\Delta$ directly contributes a term to the Navier–Stokes pressure, as we show in Section 2. We will describe a *formula* for the pressure, in fact, that shows how it is necessarily determined from the current velocity and forcing fields by solving Poisson equations with appropriate boundary conditions. This formula underlies the proof in [19] of local-time well-posedness for an *extended Navier–Stokes dynamics unconstrained by (3)*. This well-posedness proof shows that the pressure formula provides a rigorous resolution of the longstanding issue of how to properly specify boundary conditions for pressure. For discussion of this issue see the book [12], and see the paper [26] for a recent and interesting alternative.

Our goal here is to use the pressure formula to derive a number of improvements to numerical schemes for viscous incompressible flow. We will focus on simple and efficient schemes that involve projection methods for time discretization, implicit in viscosity only. For numerical performance tests we use a variety of C^0 finite elements for spatial discretization. We anticipate that our study will have a number of consequences for other kinds of time stepping and spatial discretization, however. For example, if the Reynolds number is large enough so that the time step is not severely restricted by diffusive stability criteria, one can simply use explicit fourth-order Runge–Kutta time-stepping (like for a gauge method in [9]), since the pressure field is determined by current velocity and forcing.

There are several different kinds of projection methods. We will deal with three types in this paper, which we classify as follows:

(1) *Pressure-approximation* (PA) schemes involve determining an approximation of the true pressure from the current velocity field. A key term in the pressure boundary conditions involves $\boldsymbol{n} \cdot \nabla \times \nabla \times \boldsymbol{u}$, the normal component of the curl of vorticity at the boundary. This term has appeared in the numerical literature on projection methods for many years, starting with work of Orszag, Israeli, and Deville [22] and Karniadakis et al. [17].
(2) *Pressure update* (PU) schemes involve using an existing pressure approximation p^n at time level n in determining p^{n+1}. Such schemes go back to van Kan [30], and include work by Bell et al. [5], Timmermans et al. [29] and Ren et al. [25]
(3) *Slip-correction* (SC) schemes involve adjusting the boundary condition for an intermediate velocity to ensure that the Leray-projected velocity field (which is nominally divergence free) satisfies the desired boundary condition to higher-order. The well-known 2nd-order finite-difference method of Kim and Moin [18] is of this type.

In brief, the main improvements that we propose in this paper involve (a) improved accuracy in computing the pressure in existing pressure-approximation and pressure-update projection methods, and (b) the derivation of new higher-order accurate slip-correction methods. Furthermore, our stability and accuracy tests indicate that with PA or SC schemes, one may obtain up to third-order accuracy in time for both velocity and pressure with no diffusive time-step stability restriction.

A potentially significant finding is that the PA and SC schemes which we test exhibit good performance when discretizing space in simple ways – using Lagrange finite elements of equal order for velocity and pressure, for example. These (but *not* PU) schemes are stable even with piecewise-linear approximation for both velocity and pressure. As is well known, this simple type of approximation violates the classic inf–sup condition, a condition necessary for stability in traditional mixed steady-state formulations. Our tests suggest that the inf–sup condition may not be required for stability in time-dependent computations with certain PA and SC schemes. The issue of obtaining good performance without the inf–sup condition is complex and subtle and worthy of further investigation. For brevity, here we restrict ourselves to reporting a limited number of numerical tests involving equal-order elements.

There is a large literature concerning projection methods, and discussion of it here must necessarily be limited. We refer to [13] for a recent comprehensive review. Many projection methods are closely related to the Navier–Stokes dynamics extended by the pressure formula, and its implicit/explicit discretization appearing in the work of Johnston and Liu [16]. We refer to [19] for discussion of these relations for the second-order projection or time-splitting methods of Kim and Moin [18], Timmermans et al. [29], Henshaw and Petersson [14], Brown et al. [6], and the gauge method of E and Liu [9].

The Johnston–Liu scheme is essentially a pressure-approximation scheme, as are the higher-order schemes introduced by Karniadakis et al. [17]. Leriche et al. [21] recently tested the stability of many schemes from [17] for the Stokes system in a 2D square domain with spectral collocation in space. They found unconditional stability for a certain (3,2) HOS scheme that exhibits third-order temporal accuracy for velocity. Here, we find similar behavior for slip-correction and pressure-update schemes with finite-element spatial discretizations. Moreover, using the pressure formula we find ways to recover full third-order accuracy for pressure as well as velocity. In this regard, it is important to maintain a clear distinction between the true Navier–Stokes pressure on the one hand, and the potential that enforces zero divergence in the Helmholtz decomposition on the other hand.

2. Pressure and well-posedness

We shall derive a very useful expression for the pressure based on a few identities involving the Leray-Helmholtz projection onto divergence-free fields.

The Laplace–Leray commutator. Recall that any square-integrable velocity field $\boldsymbol{u}$ has a unique Helmholtz decomposition

$$\boldsymbol{u} = \boldsymbol{v} + \nabla\phi, \tag{4}$$

where $\boldsymbol{v}$ is L^2-orthogonal to all square-integrable gradients: $\int_\Omega \boldsymbol{v}\cdot\nabla q = 0$ for all q smooth enough. Then $\boldsymbol{v}$ is divergence-free and at the boundary has vanishing component in the direction of the outward unit normal $\boldsymbol{n}$:

$$\nabla\cdot\boldsymbol{v} = 0 \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\boldsymbol{v} = 0 \quad \text{on } \Gamma. \tag{5}$$

We write $\boldsymbol{v} = \mathcal{P}\boldsymbol{u}$, defining the Leray-Helmholtz projection operator $\mathcal{P}$, and write $\phi = \mathcal{Q}\boldsymbol{u}$ to denote the zero-mean potential field in (4), satisfying

$$\Delta\phi = \nabla\cdot\boldsymbol{u} \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\nabla\phi = \boldsymbol{n}\cdot\boldsymbol{u} \quad \text{on } \Gamma. \tag{6}$$

That is, $\nabla\phi = (I-\mathcal{P})\boldsymbol{u}$. Then $\Delta(I-\mathcal{P})\boldsymbol{u} = \Delta\nabla\phi = \nabla\Delta\phi = \nabla\nabla\cdot\boldsymbol{u}$, and from this, the fact $\mathcal{P}\nabla = 0$, and the vector identity $-\nabla\times\nabla\times\boldsymbol{u} = \Delta\boldsymbol{u} - \nabla\nabla\cdot\boldsymbol{u}$, one immediately obtains the following identities described in [19]:

$$\Delta\mathcal{P}\boldsymbol{u} = (\Delta - \nabla\nabla\cdot)\boldsymbol{u} = -\nabla\times\nabla\times\boldsymbol{u}, \tag{7}$$

$$(\Delta\mathcal{P} - \mathcal{P}\Delta)\boldsymbol{u} = (I-\mathcal{P})(\Delta - \nabla\nabla\cdot)\boldsymbol{u} = -(I-\mathcal{P})\nabla\times\nabla\times\boldsymbol{u}. \tag{8}$$

Due to (8), we see that the commutator of the Laplacian and Leray projection operators is the gradient of a potential field $p_S(\boldsymbol{u})$ satisfying

$$p_S(\boldsymbol{u}) = \mathcal{Q}(\Delta - \nabla\nabla\cdot)\boldsymbol{u}, \quad \nabla p_S(\boldsymbol{u}) = (\Delta\mathcal{P} - \mathcal{P}\Delta)\boldsymbol{u}. \tag{9}$$

From (9) it follows that $p_S(\boldsymbol{u})$ is well-defined for all velocity fields with square-integrable second derivatives, and as discussed in [19], it is the unique zero-mean solution of the boundary value problem

$$\Delta p_S(\boldsymbol{u}) = 0 \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\nabla p_S(\boldsymbol{u}) = \boldsymbol{n}\cdot(\Delta - \nabla\nabla\cdot)\boldsymbol{u} \quad \text{on } \Gamma. \tag{10}$$

Formulae for pressure. Suppose now that $\boldsymbol{u}$ is a solution of the Navier–Stokes Eqs. (1)–(3). Suppose for simplicity at first that the boundary is no-slip, that is,

$$\boldsymbol{g} = 0. \tag{11}$$

Then $\boldsymbol{u} = \mathcal{P}\boldsymbol{u}$, but on the other hand, if we apply $\mathcal{P}$ to (1) and use (9) to say $\mathcal{P}\Delta\boldsymbol{u} = \Delta\mathcal{P}\boldsymbol{u} - \nabla p_S(\boldsymbol{u})$, since $\mathcal{P}\nabla p = 0$ what we find is that

$$\partial_t\boldsymbol{u} = \nu\Delta\boldsymbol{u} - \nu\nabla p_S(\boldsymbol{u}) + \mathcal{P}\boldsymbol{F}. \tag{12}$$

Since $\mathcal{P} = I - \nabla\mathcal{Q}$, comparison of (12) with (1) immediately yields the expression for pressure that we seek, assuming it is normalized to mean zero: Necessarily,

$$p = \nu p_S(\boldsymbol{u}) + \mathcal{Q}\boldsymbol{F}. \tag{13}$$

This expression is explicit in terms of solutions of boundary value problems involving the velocity and forcing fields, as in (10) and (6). We refer to $p_S(\boldsymbol{u})$ as the *Stokes pressure* since the other terms vanish in the absence of forcing and nonlinear convection terms.

The expression (13) is altered as follows in case the boundary data $\boldsymbol{g}$ is nonzero and satisfies the natural compatibility condition

$$\int_\Gamma \boldsymbol{n}\cdot\boldsymbol{g} = 0 \quad \text{for } t \geqslant 0. \tag{14}$$

Let $\mathcal{R}(\boldsymbol{g})$ denote the zero-mean harmonic function whose normal derivative at the boundary is $\boldsymbol{n}\cdot\boldsymbol{g}$, meaning that

$$\Delta\mathcal{R}(\boldsymbol{g}) = 0 \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\nabla\mathcal{R}(\boldsymbol{g}) = \boldsymbol{n}\cdot\boldsymbol{g} \quad \text{on } \Gamma. \tag{15}$$

Now, if $\boldsymbol{u}$ satisfies (1)–(3), then

$$\boldsymbol{u} - \nabla\mathcal{R}(\boldsymbol{g}) = \mathcal{P}(\boldsymbol{u} - \nabla\mathcal{R}(\boldsymbol{g})) = \mathcal{P}\boldsymbol{u}, \tag{16}$$

and applying $\mathcal{P}$ to (1) yields

$$\partial_t(\boldsymbol{u} - \nabla\mathcal{R}(\boldsymbol{g})) + \nu\nabla p_S(\boldsymbol{u}) = \nu\Delta\boldsymbol{u} + \mathcal{P}\boldsymbol{F}. \tag{17}$$

Hence the pressure is now given by

$$p = \nu p_S(\boldsymbol{u}) - \mathcal{R}(\partial_t\boldsymbol{g}) + \mathcal{Q}\boldsymbol{F}. \tag{18}$$

Finite-element computation of this pressure is best based on discretization of the following *equivalent* weak-form characterization that involves only first derivatives: For all test functions ψ with square-integrable gradient,

$$\int_\Omega \nabla p \cdot \nabla \psi = \int_\Gamma (\nu(\nabla \times \boldsymbol{u}) \cdot (\boldsymbol{n} \times \nabla \psi) - (\boldsymbol{n} \cdot \partial_t \boldsymbol{g})\psi) + \int_\Omega \boldsymbol{F} \cdot \nabla \psi. \tag{19}$$

For sufficiently regular data, (18) is also equivalent to the single boundary value problem

$$\Delta p = \nabla \cdot \boldsymbol{F} \quad \text{in } \Omega, \tag{20}$$

$$\boldsymbol{n} \cdot \nabla p = -\boldsymbol{n} \cdot (\nu \nabla \times \nabla \times \boldsymbol{u} + \partial_t \boldsymbol{g}) + \boldsymbol{n} \cdot \boldsymbol{F} \quad \text{on } \Gamma. \tag{21}$$

The appearance of the curl of vorticity in the boundary condition for pressure is familiar, dating back to Orszag et al. [22]. For purposes of analysis, however, we most often use the operator representation in (18), which distinguishes contributions to pressure by source, particularly that from the Laplace–Leray commutator.

Extended Navier–Stokes dynamics. There is a long history of using various pressure Poisson equations like (20) and (21) in computation; see [26] for a discussion and further references. In this context, it is notable that (18) has recently been placed on a sound analytical footing in relation to Navier–Stokes well-posedness theory. It was proved in [19] that the initial-boundary value problem is well-posed locally in time for Eqs. (1) and (2) *without* the divergence constraint (3)—instead using (19) to *define* the pressure. This was done in bounded domains with C^3 boundaries, for suitably regular forcing and boundary data, and for any initial velocity $\boldsymbol{u}_{\text{in}}$ having square-integrable gradient and satisfying the boundary condition (2), regardless of what its divergence is. One obtains unique strong solutions with spatial second derivatives square-integrable in space-time.

The well-posedness theory of [19] involves using the pressure formula (19) to prove the stability of a basic implicit/explicit time-difference scheme for (1) that treats the pressure and nonlinear terms explicitly in time, and only the viscosity term implicitly. Bounds on the pressure gradient come from the following (essentially sharp) estimate for the Laplace–Leray commutator, or Stokes pressure gradient, which shows that for no-slip boundary conditions, the commutator is strictly controlled at leading order by the viscosity term:

Theorem 1. *Suppose Ω is a bounded domain with C^3 boundary, and $\varepsilon > 0$. Then there is a constant C such that for all $\boldsymbol{u}$ that vanish on the boundary and have square integrable second derivatives,*

$$\int_\Omega |\nabla p_S(\boldsymbol{u})|^2 \leqslant \left(\frac{1}{2} + \varepsilon\right) \int_\Omega |\Delta \boldsymbol{u}|^2 + C \int_\Omega |\nabla \boldsymbol{u}|^2. \tag{22}$$

We refer to (1) with (18) (or equivalently (17)) as the *extended Navier–Stokes equations*. For a general solution of these equations, it follows easily from (17), (10) and (5) that the divergence $w = \nabla \cdot \boldsymbol{u}$ satisfies a diffusion equation with no-flux boundary conditions:

$$\partial_t w = \nu \Delta w \quad \text{in } \Omega, \quad \boldsymbol{n} \cdot \nabla w = 0 \quad \text{on } \Gamma. \tag{23}$$

The divergence $w \equiv 0$ initially if and only if $w \equiv 0$ for all later time, and this produces a solution of NSE including the incompressibility constraint. The dynamics of the extended NSE extends the constrained dynamics of NSE in a well-posed manner off the "invariant manifold" of states satisfying (3).

3. Time discretization by projection methods

Since pressure is determined explicitly by (19) (or equivalently (18)), it is straightforward to discretize the extended Navier–Stokes Eqs. (1), (2) and (19) in time. To achieve efficiency at low to moderate Reynolds number, it is convenient to treat the viscosity term implicitly and treat the pressure and convection terms explicitly. In this section we will describe three classes of time discretizations of this type, and examine their formal accuracy in light of the formulae from the previous section that involve pressure. We treat semi-discrete schemes that are discrete in time and continuous in space. Time discretization has long been considered a main source of numerical boundary-layer error. This is already apparent in the studies by Orszag et al. [22] and Brown et al. [6], for example, and is consistent with our numerical tests. We will discuss implementation details for space discretization by finite elements in Section 4.

It will be better to use a rotational form of the nonlinear term, writing

$$\boldsymbol{F} = \boldsymbol{f} - \boldsymbol{h}, \quad \boldsymbol{h} = (\nabla \times \boldsymbol{u}) \times \boldsymbol{u}. \tag{24}$$

This is different from the previous expression for $\boldsymbol{F}$, but is related by the identity

$$\boldsymbol{u} \cdot \nabla \boldsymbol{u} = (\nabla \times \boldsymbol{u}) \times \boldsymbol{u} + \frac{1}{2} \nabla |\boldsymbol{u}|^2. \tag{25}$$

The pressure formulae in (18)–(21) remain valid with the new $\boldsymbol{F}$, the pressure changing by a term $\frac{1}{2}|\boldsymbol{u}|^2$ up to a constant.

For each kind of projection method, we will find a divergence-free approximate velocity $\boldsymbol{u}^j$ at time $t_j = j\Delta t$ by decomposing an intermediate velocity $\boldsymbol{u}^j_*$ according to

$$\boldsymbol{u}_*^j = \boldsymbol{u}^j + \nabla q^j, \quad \Delta q^j = \nabla \cdot \boldsymbol{u}_*^j \quad \text{in } \Omega, \quad \boldsymbol{n} \cdot \nabla q^j = 0 \quad \text{on } \Gamma. \tag{26}$$

We denote the nonlinear terms corresponding to the divergence-free velocities $\boldsymbol{u}^j$ by

$$\boldsymbol{h}^j = (\nabla \times \boldsymbol{u}^j) \times \boldsymbol{u}^j = (\nabla \times \boldsymbol{u}_*^j) \times (\boldsymbol{u}_*^j - \nabla q^j). \tag{27}$$

We will indicate backward differentiation formulas of order k and extrapolation formulas of order m using the notation

$$\mathcal{D}_k \boldsymbol{u}^{n+1} = \frac{1}{\Delta t} \sum_{j \geqslant 0} \alpha_j^k \boldsymbol{u}^{n+1-j}, \quad \mathcal{E}_m \boldsymbol{u}^{n+1} = \sum_{j \geqslant 1} \beta_j^m \boldsymbol{u}^{n+1-j}. \tag{28}$$

The nonzero coefficients for these formulae are listed in Table 1.

3.1. Slip-corrected projection methods

One can describe rather simple time-discrete schemes that formally achieve high-order accuracy without using any explicit approximation to pressure at all. The idea is to adjust slip at the boundary, in a fashion similar to that in the classic 2nd-order projection method of Kim & Moin [18].

To determine the approximate velocity $\boldsymbol{u}^{n+1}$ at time $t_{n+1} = (n+1)\Delta t$, suppose that for all $j \leqslant n$ the decomposition (26) is available. Fix a pair of integers (k, m) with $k \geqslant m > 0$. We discretize (1) in time using the kth-order backward differentiation formula for the time derivative and kth-order extrapolation to approximate $\boldsymbol{h}^{n+1}$. We use mth-order extrapolation to approximate ∇q^{n+1} on the boundary. We update velocity by computing $\boldsymbol{u}_*^{n+1}$ and q^{n+1} to satisfy

$$\frac{1}{\Delta t}\left(\alpha_0^k \boldsymbol{u}_*^{n+1} + \sum_{j \geqslant 1} \alpha_j^k \boldsymbol{u}^{n+1-j}\right) = \nu \Delta \boldsymbol{u}_*^{n+1} + \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}^{n+1} \quad \text{in } \Omega, \tag{29}$$

$$\boldsymbol{u}_*^{n+1} = \boldsymbol{g}^{n+1} + \nabla \mathcal{E}_m q^{n+1} \quad \text{on } \Gamma, \tag{30}$$

$$\Delta q^{n+1} = \nabla \cdot \boldsymbol{u}_*^{n+1} \quad \text{in } \Omega, \quad \boldsymbol{n} \cdot \nabla q^{n+1} = 0 \quad \text{on } \Gamma. \tag{31}$$

Then we can write

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}_*^{n+1} - \nabla q^{n+1}. \tag{32}$$

We refer to this scheme as the (k, m) SC scheme. (SC is for slip correction.) The case $(k, m) = (3, 2)$ is perhaps most interesting, yielding third-order accuracy with good stability in tests.

This scheme is just as efficient as many classic projection methods, involving the solution of one scalar Poisson equation in addition to N decoupled scalar elliptic equations for velocity per time step. Note that with finite-element discretization, the quantities $\boldsymbol{u}^j$ need not be computed every time step. One can use $\boldsymbol{u}_*^j - \nabla q^j$ for $\boldsymbol{u}^j$ ($j \leqslant n$) in (29), as in (27). Also note $\nabla \cdot \boldsymbol{u}^{n+1} = 0$, and $\boldsymbol{n} \cdot \boldsymbol{u}^{n+1} = \boldsymbol{n} \cdot \boldsymbol{g}^{n+1}$ on Γ, but tangential components of $\boldsymbol{u}^{n+1}$ and $\boldsymbol{g}^{n+1}$ may not match. Following the advice in [22] for avoiding a weak instability in the Kim–Moin scheme, in implementing (30) one should enforce normal-component matching explicitly, requiring

$$\boldsymbol{n} \cdot \boldsymbol{u}_*^{n+1} = \boldsymbol{n} \cdot \boldsymbol{g}^{n+1} \quad \text{and } \boldsymbol{n} \times \boldsymbol{u}_*^{n+1} = \boldsymbol{n} \times (\boldsymbol{g}^{n+1} + \nabla \mathcal{E}_2 q^{n+1}) \quad \text{on } \Gamma, \tag{33}$$

and not rely on the boundary conditions in (31).

Accuracy for velocity. Though the pressure is neglected in the step (29), it is easy to see that the velocity $\boldsymbol{u}^n$ satisfies a formally kth-order-accurate discretization of the momentum equation in the following way. Apply the projection $\mathcal{P}$ to (29), noting that $\boldsymbol{n} \cdot \boldsymbol{u}^j = \boldsymbol{n} \cdot \boldsymbol{g}^j$, hence as in (16),

$$\mathcal{P}\boldsymbol{u}_*^j = \mathcal{P}\boldsymbol{u}^j = \boldsymbol{u}^j - \nabla \mathcal{R}(\boldsymbol{g}^j). \tag{34}$$

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	

Because $\mathcal{P}\Delta \boldsymbol{u}_*^{n+1} = \Delta \mathcal{P}\boldsymbol{u}_*^{n+1} - \nabla p_S(\boldsymbol{u}_*^{n+1}) = \Delta \boldsymbol{u}^{n+1} - \nabla p_S(\boldsymbol{u}^{n+1})$, we find that

$$\mathcal{D}_k \mathcal{P}\boldsymbol{u}^{n+1} + \nu \nabla p_S(\boldsymbol{u}^{n+1}) = \nu \Delta \boldsymbol{u}^{n+1} + \mathcal{P}\boldsymbol{f}^{n+1} - \mathcal{P}\mathcal{E}_k \boldsymbol{h}^{n+1}, \tag{35}$$

whence $\boldsymbol{u}^{n+1}$ satisfies

$$\mathcal{D}_k \boldsymbol{u}^{n+1} + \nabla \bar{p}^{n+1} = \nu \Delta \boldsymbol{u}^{n+1} + \boldsymbol{f}^{n+1} - \mathcal{E}_k \boldsymbol{h}^{n+1}, \tag{36}$$

where

$$\bar{p}^{n+1} = \nu p_S(\boldsymbol{u}^{n+1}) - \mathcal{R}(\mathcal{D}_k \boldsymbol{g}^{n+1}) + \mathcal{Q}\boldsymbol{f}^{n+1} - \mathcal{Q}\mathcal{E}_k \boldsymbol{h}^{n+1}. \tag{37}$$

Clearly (36) and (37) are kth-order accurate discretizations of (1) and (18) respectively.

Slip error. It remains to study the slip error in the boundary condition for $\boldsymbol{u}^{n+1}$. From (30) and (32) it follows that this error is given by

$$\boldsymbol{u}^{n+1} - \boldsymbol{g}^{n+1} = -\nabla(q^{n+1} - \mathcal{E}_m q^{n+1}) \quad \text{on } \Gamma. \tag{38}$$

Comparing (29) and (36) we find (after writing $\alpha_0 = \alpha_0^k$)

$$\left(\frac{\alpha_0}{\Delta t} - \nu\Delta\right) q^{n+1} = \bar{p}^{n+1}. \tag{39}$$

This equation together with (28) yields

$$\left(\frac{\alpha_0}{\Delta t} - \nu\Delta\right)(q^{n+1} - \mathcal{E}_m q^{n+1}) = \bar{p}^{n+1} - \mathcal{E}_m \bar{p}^{n+1}. \tag{40}$$

Since $\bar{p}^n$ consistently approximates the pressure p, the right-hand side of (40) is formally $\Delta t^m \partial_t^m p$ plus higher-order terms. Since $\boldsymbol{n} \cdot \nabla q^j = 0$ on Γ, the quantity

$$\hat{q} = q^{n+1} - \mathcal{E}_m q^{n+1} \tag{41}$$

also satisfies $\boldsymbol{n} \cdot \nabla \hat{q} = 0$, and we can infer from this boundary value problem that $\hat{q} = O(\Delta t^{m+1})$ formally. (Indeed, if one knew rigorously that the right-hand side takes values in a fixed interval $[-M, M]$ where $M = O(\Delta t^m)$, then it follows from the maximum principle that the solution to (40) takes values in $[-\hat{M}, \hat{M}]$ with $\hat{M} = M\Delta t/\alpha_0 = O(\Delta t^{m+1})$.) Presuming the boundary and the data are smooth, it is reasonable to expect that derivatives tangential to the boundary are of the same order $O(\Delta t^{m+1})$. Then the full gradient $\nabla \hat{q} = O(\Delta t^{m+1})$ on the boundary since the normal component is zero. We conclude that formally the slip error $\boldsymbol{u}^{n+1} - \boldsymbol{g}^{n+1}$ is $O(\Delta t^{m+1})$. This indicates that overall the order of accuracy for $\boldsymbol{u}^{n+1}$ is the minimum of k and $m+1$.

Approximate pressure. Though pressure is not computed explicitly in the scheme (29)–(32), an approximation $\hat{p}^{n+1}$ whose formal accuracy matches that of velocity can be computed at negligible further cost and without solving any further Poisson equations, by using (39) with (31). Namely we can set

$$\hat{p}^{n+1} = \bar{p}^{n+1} = \frac{\alpha_0}{\Delta t} q^{n+1} - \nu \nabla \cdot \boldsymbol{u}_*^{n+1}. \tag{42}$$

This expression involves computed quantities and by (37) it is a consistent kth-order approximation to the pressure corresponding to $\boldsymbol{u}^{n+1}$ from (18). Hence it should approximate the true pressure with order of accuracy $\min(k, m+1)$.

Remarks 1

1. It seems remarkable that Eq. (36) for the divergence-free velocity $\boldsymbol{u}^{n+1}$ is *fully implicit* with respect to Stokes pressure. Evidently this is due to the commutator formula (9). The same thing naturally happens for many different kinds of projection and splitting methods.
2. We emphasize that the final projection step (32) should *not* be regarded as a fractional step that solves $\partial_t u + \nabla p = 0$. (For an interesting perspective relating fractional step methods to block LU decompositions see [24].) The quantity q^{n+1} is *not* pressure times Δt, despite its role in enforcing the zero-divergence condition for $\boldsymbol{u}^{n+1}$. It is instructive to consider the error that occurs if one takes the approximate pressure to be

   $$\hat{p}^{n+1} = \frac{\alpha_0}{\Delta t} q^{n+1},$$

   a kind of approximation not uncommon in the literature. Supposing $m+1 \geqslant k$ for convenience, the error in this approximation is evidently $\nu \nabla \cdot \boldsymbol{u}_*^{n+1} + O(\Delta t^k)$. Put $\varepsilon = \sqrt{\nu \Delta t/\alpha_0}$. From (29), the quantity $w = \nu \nabla \cdot \boldsymbol{u}_*^{n+1}$ satisfies

   $$(1 - \varepsilon^2 \Delta) w = \varepsilon^2 \nabla \cdot (\boldsymbol{f}^{n+1} - \mathcal{E}_k \boldsymbol{h}^{n+1}) =: \varepsilon^2 a_1 \quad \text{in } \Omega, \tag{43}$$

   $$\boldsymbol{n} \cdot \nabla w = -\boldsymbol{n} \cdot \nabla \bar{p}^{n+1} =: a_2 \quad \text{on } \Gamma. \tag{44}$$

   Formally a_1 and a_2 are $O(1)$. We expect a boundary layer, whose leading-order behavior may be described by taking the boundary to be locally flat, a_1 and a_2 approximately constant, and presuming w depends only on the distance s to the boundary. Then the leading-order solution in the boundary layer is

   $$w \approx \varepsilon^2 a_1 - \varepsilon a_2 e^{-s/\varepsilon}. \tag{45}$$

Thus in this case one expects pressure error of order $O(\Delta t)$ in the interior, with a boundary layer error $O(\sqrt{\Delta t})$. And since $\partial_s w \approx a_2$ at $s = 0$, the pressure gradient would have error $O(1)$ at the boundary, as one can also see directly from (42).

3. The slip-correction incorporated into the boundary condition (30) is related to the second-order projection method of Kim and Moin [18] in the following way. Corresponding to (29) Kim and Moin use a Crank–Nicholson time discretization for viscous terms and Adams–Bashforth for convective terms. Instead of (30) the boundary condition imposed on the intermediate velocity is

$$\boldsymbol{u}_*^{n+1} = \boldsymbol{g}^{n+1} + \Delta t \nabla \phi^n, \tag{46}$$

where $\Delta t\,\phi^n$ in [18] corresponds to q^n here, which is a first-order extrapolation approximating q^{n+1}. In [18], this boundary condition is derived in Section 3 and appears two lines below (15). Eq. (4) of [18] is a discretized version of (32) with q^{n+1} replaced by $\Delta t\,\phi^{n+1}$. The potential ϕ^{n+1} is determined by solving a discrete Poisson equation to enforce $\nabla \cdot \boldsymbol{U}^{n+1} = 0$ discretely. This is done on a staggered grid "without the need for boundary conditions for ϕ^{n+1}." However, Eq. (9) of [18], written after "incorporation of the velocity boundary conditions," indicates that the discretization is based on the single derivative boundary condition

$$\Delta t\, \boldsymbol{n} \cdot \nabla \phi^{n+1} = \boldsymbol{n} \cdot (\boldsymbol{u}_*^{n+1} - \boldsymbol{g}^{n+1}). \tag{47}$$

(For a similar observation see [28]. To be precise, (9) is obtained using the replacement

$$\Delta t \frac{\phi^{n+1}(i,1,k) - \phi(i,0,k)}{x_2(1) - x_2(0)} = \hat{u}_2^{n+1}\left(i, \frac{1}{2}, k\right) - u_2^{n+1}\left(i, \frac{1}{2}, k\right),$$

which corresponds to (47).) If one follows the advice in [22] to modify (46) and explicitly enforce normal-component matching as in (33), the condition (47) becomes $\boldsymbol{n} \cdot \nabla \phi^{n+1} = 0$.

3.2. Pressure-approximation schemes

An alternative way to achieve third-order accuracy efficiently and with good stability involves approximating the pressure as determined by the formula (18). In particular, it was observed by Karniadakis et al. [17] (in a context involving Adams–Moulton/Adams–Bashforth discretizations for formulas equivalent to (1), (2), (20) and (21) but without the commutator formulae) that one can reduce the accuracy of approximating the curl–curl term in the boundary condition (21). For the linear Stokes equations, Leriche et al. [21] studied the numerical performance of a number of spectral collocation schemes from [17] based on backward differentiation. Their results indicated unconditional stability using third-order backward differentiation together with second-order extrapolation for the curl–curl boundary condition. For a smooth test problem they obtained third-order accuracy for velocity but less for pressure (as discussed below).

The following time-discretization scheme is close to ones studied in [17,21]: Let us use

$$\boldsymbol{h}_*^j = \boldsymbol{u}_*^j \cdot \nabla \boldsymbol{u}_*^j \tag{48}$$

in place of (27). Fix a pair of integers (k, m) with $k \geqslant m > 0$, and for convenience assume $m + 1 \geqslant k$. (Again $(k, m) = (3, 2)$ is particularly interesting.) We update the *intermediate* velocity $\boldsymbol{u}_*^n$ by successively determining $\bar{\boldsymbol{f}}, \bar{P}$ and $\boldsymbol{u}_*^{n+1}$ to satisfy

$$\bar{\boldsymbol{f}} = \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}_*^{n+1} - \frac{1}{\Delta t} \sum_{j \geqslant 1} \alpha_j^k \boldsymbol{u}_*^{n+1-j}, \tag{49}$$

$$\Delta \bar{P} = \nabla \cdot \bar{\boldsymbol{f}} \quad \text{in } \Omega, \tag{50}$$

$$\boldsymbol{n} \cdot \nabla \bar{P} = -\boldsymbol{n} \cdot \left(\nu \nabla \times \nabla \times \mathcal{E}_m \boldsymbol{u}_*^{n+1} + \frac{\alpha_0^k}{\Delta t} \boldsymbol{g}^{n+1} \right) + \boldsymbol{n} \cdot \bar{\boldsymbol{f}} \quad \text{on } \Gamma, \tag{51}$$

$$\frac{\alpha_0^k}{\Delta t} \boldsymbol{u}_*^{n+1} + \nabla \bar{P} = \nu \Delta \boldsymbol{u}_*^{n+1} + \bar{\boldsymbol{f}} \quad \text{in } \Omega, \quad \boldsymbol{u}_*^{n+1} = \boldsymbol{g}^{n+1} \quad \text{on } \Gamma. \tag{52}$$

We refer to this scheme as the (k, m) PA scheme (PA for pressure approximation). As before, we do not compute the divergence-free velocity $\boldsymbol{u}^{n+1} = \boldsymbol{u}_*^{n+1} - \nabla q^{n+1}$ as in (31) and (32) in each time step, but only as needed—for the final output, say. (Derivatives of $\boldsymbol{u}_*^{n+1}$ have weak boundary layers, as we shall see presently.)

Accuracy for velocity. We can study accuracy by applying $\mathcal{P}$ to (52). What $\bar{P}$ is does not matter, since $\mathcal{P}\nabla = 0$. Exactly as before, we get

$$\mathcal{D}_k \boldsymbol{u}^{n+1} + \nabla \bar{p}^{n+1} = \nu \Delta \boldsymbol{u}^{n+1} + \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}_*^{n+1}, \tag{53}$$

$$\bar{p}^{n+1} = \nu p_S(\boldsymbol{u}^{n+1}) - \mathcal{R}(\mathcal{D}_k \boldsymbol{g}^{n+1}) + \mathcal{Q}\boldsymbol{f}^{\,n+1} - \mathcal{Q}\mathcal{E}_k \boldsymbol{h}_*^{n+1}. \tag{54}$$

This yields kth-order accurate discretizations of (1) and (18) (anticipating that the difference between $\boldsymbol{h}^j$ and $\boldsymbol{h}_*^j$ does not matter, see below). But now the slip error at the boundary is only due to the decomposition (26):

$$\boldsymbol{u}^{n+1} = \boldsymbol{g}^{n+1} - \nabla q^{n+1} \quad \text{on } \Gamma. \tag{55}$$

If we recognize that

$$\mathcal{Q}\boldsymbol{u}_*^j = q^j + \mathcal{R}(\boldsymbol{g}^j), \tag{56}$$

since $\boldsymbol{u}^j + \nabla q^j = \boldsymbol{u}_*^j = \mathcal{P}\boldsymbol{u}_*^j + \nabla \mathcal{Q}\boldsymbol{u}_*^j = \boldsymbol{u}^j - \nabla \mathcal{R}(\boldsymbol{g}^j) + \nabla \mathcal{Q}\boldsymbol{u}_*^j$, and we write $\boldsymbol{F}^{n+1} = \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}_*^{n+1}$ for simplicity, we see that Eqs. (50) and (51) mean

$$\begin{aligned}\bar{P} &= \nu p_S(\mathcal{E}_m \boldsymbol{u}_*^{n+1}) - \frac{\alpha_0^k}{\Delta t}\mathcal{R}(\boldsymbol{g}^{n+1}) + \mathcal{Q}\boldsymbol{F}^{n+1} - \frac{1}{\Delta t}\sum_{j\geqslant 1}\alpha_j^k \mathcal{Q}\boldsymbol{u}_*^{n+1-j} \\ &= \nu p_S(\mathcal{E}_m \boldsymbol{u}^{n+1}) - \mathcal{R}(\mathcal{D}_k \boldsymbol{g}^{n+1}) + \mathcal{Q}\boldsymbol{F}^{n+1} - \frac{1}{\Delta t}\sum_{j\geqslant 1}\alpha_j^k q^{n+1-j} = \bar{p}^{n+1} - \nu p_S(\boldsymbol{u}^{n+1} - \mathcal{E}_m \boldsymbol{u}^{n+1}) - \frac{1}{\Delta t}\sum_{j\geqslant 1}\alpha_j^k q^{n+1-j}.\end{aligned} \tag{57}$$

Since (52) means

$$\mathcal{D}_k \boldsymbol{u}_*^{n+1} + \nabla \bar{P} = \nu \Delta \boldsymbol{u}_*^{n+1} + \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}_*^{n+1},$$

comparing with (53) using $\boldsymbol{u}_*^j = \boldsymbol{u}^j + \nabla q^j$ yields

$$\left(\frac{\alpha_0}{\Delta t} - \nu\Delta\right) q^{n+1} = \nu p_S(\boldsymbol{u}^{n+1} - \mathcal{E}_m \boldsymbol{u}^{n+1}), \tag{58}$$

with $\boldsymbol{n}\cdot\nabla q^{n+1} = 0$ on Γ. From this we see that q^{n+1} and ∇q^{n+1} are formally $O(\Delta t^{m+1})$, meaning that $\boldsymbol{u}^{n+1} - \boldsymbol{g}^{n+1} = O(\Delta t^{m+1})$ on Γ. Thus if $m+1 \geqslant k$, both quantities $\boldsymbol{u}^{n+1}$ and $\boldsymbol{u}_*^{n+1}$ are kth-order accurate approximations to velocity. The former has zero divergence and the latter has no slip error.

The divergence $w^{n+1} = \nabla\cdot\boldsymbol{u}_*^{n+1} = \Delta q^{n+1}$ can be expected to have a weak boundary layer, however. This is because, by applying Δ and taking the normal derivative at the boundary in (58), we find

$$\left(\frac{\alpha_0}{\Delta t} - \nu\Delta\right) w^{n+1} = 0 \quad \text{in } \Omega, \tag{59}$$

$$\boldsymbol{n}\cdot\nabla w^{n+1} = \boldsymbol{n}\cdot\nabla\times\nabla\times(\boldsymbol{u}^{n+1} - \mathcal{E}_m\boldsymbol{u}^{n+1}) \quad \text{on } \Gamma. \tag{60}$$

A formal boundary layer analysis like that in remark 2 of the previous section yields (45) with $a_1 = 0$ and $a_2 = O(\Delta t^m)$, whence $w^{n+1} \approx O(\Delta t^m)\varepsilon e^{-s/\varepsilon}$ with $\varepsilon = \sqrt{\nu\Delta t/\alpha_0}$. Therefore we expect the maximum norm to be $O(\sqrt{\nu}\Delta t^{m+1/2})$ and the L^2 norm to be $O(\nu^{3/4}\Delta t^{m+3/4})$.

The presence of weak boundary layers in second derivatives of q^{n+1} prompts concern over the accuracy of approximation of $\boldsymbol{h}_*^j$, which replaces $\boldsymbol{h}^j$ in (36). One has the identity

$$\boldsymbol{u}_*^j\cdot\nabla\boldsymbol{u}_*^j - \boldsymbol{u}^j\cdot\nabla\boldsymbol{u}^j = (\nabla\times\boldsymbol{u}^j)\times(\nabla q^j) + \frac{1}{2}\nabla(|\boldsymbol{u}_*^j|^2 - |\boldsymbol{u}^j|^2). \tag{61}$$

On the right-hand side, the first term is $O(\Delta t^{m+1})$ and the second term is a gradient. Then one sees $\mathcal{P}\boldsymbol{h}_*^j = \mathcal{P}(\boldsymbol{u}^j\cdot\nabla\boldsymbol{u}^j) + O(\Delta t^{m+1})$ and since $m+1 \geqslant k$, indeed (36) with $\boldsymbol{h}_*^j$ for $\boldsymbol{h}^j$ is a consistent kth-order accurate approximation to (1) in this case.

Approximate pressure. If $k = m+1$, then the quantity $\bar{P}$ that appears in (50) and (51) is not a fully kth-order-accurate approximation to the pressure corresponding to (18). Using (58) in (57) yields

$$\bar{P} = \bar{p}^{n+1} - \mathcal{D}_k q^{n+1} + \nu w^{n+1}. \tag{62}$$

The error in the boundary layer should be dominated by the last term, being $O(\Delta t^{m+1/2})$ in max norm (the one relevant for boundary forces) and $O(\Delta t^{m+3/4})$ in L^2 for fixed ν. This is quite consistent with the numerical results reported in tables VIII and X of [21] for the pressure in the (3,2) HOS scheme of that paper, corresponding to $k = 3, m = 2$. Note that the error in ∇w^{n+1} and $\nabla\bar{P}$ should be $O(\Delta t^2)$ in max norm in this case, in fact.

For the scheme (49)–(52), the quantities q^j are not directly available. However, the pressure approximation

$$\hat{p}^{n+1} = \bar{p}^{n+1} - \mathcal{D}_k q^{n+1} = \bar{P} - \nu\nabla\cdot\boldsymbol{u}_*^{n+1} \tag{63}$$

is computable without solving a further Poisson equation. This should be a kth-order accurate approximation to the pressure corresponding to (18) for $\boldsymbol{u}^{n+1}$. For, formally $\mathcal{D}_k q^{n+1} = O(\Delta t^k)$ because the q^j are $O(\Delta t^{m+1})$. The expression (54) for $\bar{p}^{n+1}$ is evidently kth-order accurate except in the last term, where we have $\boldsymbol{h}_*^j$ instead of $\boldsymbol{u}^j\cdot\nabla\boldsymbol{u}^j$. But by the identity (61), the difference

$$\mathcal{Q}\boldsymbol{h}_*^j - \mathcal{Q}(\boldsymbol{u}^j\cdot\nabla\boldsymbol{u}^j) = O(\Delta t^k) + \frac{1}{2}\left(|\boldsymbol{u}_*^j|^2 - |\boldsymbol{u}^j|^2\right) = O(\Delta t^k). \tag{64}$$

Thus we expect (54), and hence (63), to be kth-order accurate.

Output. For output on time step N one probably wants to use the final velocity $\boldsymbol{u}_*^N$ to compute a divergence-free velocity $\boldsymbol{u}^N$ as in (31) and (32), and the pressure p^N that corresponds to $\boldsymbol{u}^N$ via (18) (i.e., (20) and (21)).

3.3. Pressure-update methods

Methods that update pressure approximations from previous time steps have been introduced by a number of authors, especially including van Kan [30], Bell et al. [5], Timmermans et al. [29], and most recently Ren et al. [25], whose scheme is formally third-order accurate. Here we describe a class of methods of this kind.

With basic notation as in the previous subsections, fix a pair of integers (k,m) with $k \geqslant m > 0$, and suppose $\boldsymbol{u}^j_*, q^j$ and P^j are known for $j \leqslant n$. With $\boldsymbol{u}^j$ given by (26), we determine $\boldsymbol{u}^{n+1}_*, q^{n+1}$ and P^{n+1} from the following.

$$\frac{1}{\Delta t}\left(\alpha_0^k \boldsymbol{u}_*^{n+1} + \sum_{j\geqslant 1} \alpha_j^k \boldsymbol{u}^{n+1-j}\right) + \nabla \mathcal{E}_m P^{n+1} = \nu \Delta \boldsymbol{u}_*^{n+1} + \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}^{n+1}, \tag{65}$$

$$\boldsymbol{u}_*^{n+1} = \boldsymbol{g}^{n+1} \quad \text{on } \Gamma, \tag{66}$$

$$\Delta q^{n+1} = \nabla \cdot \boldsymbol{u}_*^{n+1} \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\nabla q^{n+1} = 0 \quad \text{on } \Gamma. \tag{67}$$

$$P^{n+1} = \mathcal{E}_m P^{n+1} + \left(\frac{\alpha_0^k}{\Delta t} - \nu\Delta\right) q^{n+1} \tag{68}$$

The divergence-free velocity $\boldsymbol{u}^{n+1}$ is given by (32) as usual. We refer to this scheme as a (k,m) PU scheme. (PU for pressure update.) Timmermans et al. [29] introduced what is essentially a $(2,2)$ PU scheme. The scheme of [25] is a $(3,3)$ PU scheme.

Accuracy for velocity and pressure. Applying $\mathcal{P}$, exactly as in subsection 3.1 we find that $\boldsymbol{u}^{n+1}$ satisfies (36), with $\bar{p}^{n+1}$ given exactly by (37). Thus the discretization of the momentum equation is kth-order accurate. Subtracting (36) from (65), we find that

$$\left(\frac{\alpha_0}{\Delta t} - \nu\Delta\right) q^{n+1} = \bar{p}^{n+1} - \mathcal{E}_m P^{n+1}. \tag{69}$$

By (68) this means that

$$P^{n+1} = \bar{p}^{n+1}. \tag{70}$$

Thus by (37) this scheme provides a kth-order accurate approximation $\hat{p}^{n+1} = P^{n+1}$ to the pressure formula (18).

Slip error. It remains to consider the slip error $\boldsymbol{u}^{n+1} - \boldsymbol{g}^{n+1}$ on the boundary. By (70) it follows that the right-hand side of (69) is $\Delta t^m \partial_t^m p$ plus higher-order terms. Then formally $q^{n+1} = O(\Delta t^{m+1})$ and the same holds for tangential derivatives. By (32) it follows that the slip error $\boldsymbol{u}^{n+1} - \boldsymbol{g}^{n+1} = O(\Delta t^{m+1})$ on Γ, and this indicates kth-order accuracy overall for the scheme if $m+1 \geqslant k$.

4. Spatial discretization by C^0 finite elements

To obtain fully discrete schemes from the time-difference schemes above using C^0 finite elements, a key idea is to treat the Stokes pressure (or curl–curl boundary condition) by using the weak formulation in (19), as was done in [16]. But this is unnecessary for the 'pressureless' slip-corrected projection scheme of Section 3.1, whose discretization is fairly straightforward—we only have to describe how we handle the slip boundary condition.

We denote by $\langle f, g\rangle = \int_\Omega fg$ the inner product in $L^2(\Omega)$ and similarly $\langle f,g\rangle_\Gamma = \int_\Gamma fg$ for the inner product in $L^2(\Gamma)$. Given a discretization parameter $h > 0$, we let Y_h be a space of C^0 finite elements for approximating pressure and potentials, with $Y_h \subset H^1(\Omega)/\mathbb{R}$, the Sobolev space of functions with square-integrable gradients, modulo constants. Also let X_h be a space of C^0 finite elements for approximating the velocity field, with $X_h \subset H^1(\Omega,\mathbb{R}^N)$ having a nodal basis. Let $X_{0,h} = X_h \cap H^1_0(\Omega,\mathbb{R}^N)$ be the subspace of X_h consisting of vector fields that vanish on Γ.

The decomposition (26) into a divergence-free field and a gradient field with vanishing normal derivative at the boundary means that q^j is determined (up to constants) by requiring

$$\langle \nabla q^j, \nabla\psi\rangle = \langle \boldsymbol{u}^j_*, \nabla\psi\rangle - \langle \boldsymbol{n}\cdot\boldsymbol{u}^j_*, \psi\rangle_\Gamma \quad \forall \psi \in H^1(\Omega)/\mathbb{R}. \tag{71}$$

For consistency the integral $\int_\Gamma \boldsymbol{n}\cdot\boldsymbol{u}^j_*$ must vanish. Given a discrete field $\boldsymbol{u}^j_{*h} \in X_h$ that we desire to satisfy $\boldsymbol{n}\cdot\boldsymbol{u}^j_{*h} = \boldsymbol{n}\cdot\boldsymbol{g}^j$ on Γ, where $\boldsymbol{g}$ satisfies the consistency condition (14), we determine the corresponding discrete decomposition as follows. We find $q^j_h \in Y_h$ to satisfy

$$\langle \nabla q^j_h, \nabla\psi_h\rangle = \langle \boldsymbol{u}^j_{*h}, \nabla\psi\rangle - \langle \boldsymbol{n}\cdot\boldsymbol{g}^j, \psi\rangle_\Gamma \quad \forall \psi_h \in Y_h. \tag{72}$$

Then we write $\boldsymbol{U}^j_h = \boldsymbol{u}^j_{*h} - \nabla q^j_h$. This need not belong to X_h; the terms $\boldsymbol{u}^j_{*h}$ and ∇q^j_h can be handled separately throughout. For output, the vector field $\boldsymbol{U}^j_h$ may be L^2-projected into X_h.

Slip correction. To discretize the slip-correction schemes of Section 3.1, we suppose we have $\boldsymbol{u}^j_{*h}$ and q^j_h for all $j \leqslant n$, and write

$$\boldsymbol{H}^j_h = (\nabla\times\boldsymbol{u}^j_{*h})\times\boldsymbol{U}^j_h = (\nabla\times\boldsymbol{u}^j_{*h})\times(\boldsymbol{u}^j_{*h} - \nabla q^j_h). \tag{73}$$

The discrete momentum equations for determining $\boldsymbol{u}_{*h}^{n+1}$ are

$$\frac{\alpha_0^k}{\Delta t}\langle \boldsymbol{u}_{*h}^{n+1}, \boldsymbol{v}_h\rangle + \nu\langle \nabla \boldsymbol{u}_{*h}^{n+1}, \nabla \boldsymbol{v}_h\rangle = \langle \boldsymbol{F}_h^{n+1}, \boldsymbol{v}_h\rangle \quad \forall \boldsymbol{v}_h \in X_{0,h}, \tag{74}$$

where

$$\boldsymbol{F}_h^{n+1} = \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{H}_h^{n+1} - \frac{1}{\Delta t}\sum_{j\geqslant 1} \alpha_j^k \boldsymbol{u}_h^{n+1-j}.$$

These equations suffice to determine $\boldsymbol{u}_{*h}^{n+1} \in X_h$ once we specify discrete boundary conditions of the form

$$\boldsymbol{u}_{*h}^{n+1} = \boldsymbol{g}^{n+1} + \boldsymbol{r}_h^{n+1} \quad \text{on } \Gamma_h, \tag{75}$$

where Γ_h is the collection of grid nodes on Γ. Then we find $q_h^{n+1} \in Y_h$ by (72) and write $\boldsymbol{u}_h^{n+1} = \boldsymbol{u}_{*h}^{n+1} - \nabla q_h^{n+1}$ as above.

The terms $\boldsymbol{r}_h^{n+1}$ lie in the space of boundary values of functions in X_h and approximate $\nabla \mathcal{E}_m q_h^{n+1}$, which lies in ∇Y_h. For consistency with volume conservation, we require

$$\boldsymbol{n}\cdot \boldsymbol{r}_h^{n+1} = 0 \quad \text{on } \Gamma_h. \tag{76}$$

Consider a polygonal domain in 2D, whose boundary is the union of straight edges Γ^i. At corners where two edges meet, there are two independent normals and this forces $\boldsymbol{r}_h^{n+1} = 0$. Along each edge Γ^i, the tangential component $\boldsymbol{\tau}\cdot \boldsymbol{r}_h^{n+1}$ must be found in the space Z_h^i of tangential components of functions $\boldsymbol{v}_h \in X_h$ that satisfy $\boldsymbol{\tau}\cdot \boldsymbol{v}_h = 0$ at the endpoints of Γ^i. (When X_h is a space of Lagrange piecewise polynomial finite elements, the space Z_h^i is just a space of scalar Lagrange piecewise-polynomial elements on Γ^i that vanish at the two endpoints.) To determine $\boldsymbol{\tau}\cdot \boldsymbol{r}_h^{n+1}$, it is convenient to simply project the tangential derivative $\boldsymbol{\tau}\cdot \nabla \mathcal{E}_m q_h^{n+1}$ into Z_h^i using the inner product in $L^2(\Gamma^i)$.

This procedure will generalize naturally to 3D polyhedral domains. The condition (76) forces $\boldsymbol{r}_h^{n+1} = 0$ at corners where 3 faces meet. Two components vanish along edges where two faces meet, and the tangential component can be determined by L^2 projection of $\boldsymbol{\tau}\cdot \nabla \mathcal{E}_m q_h^{n+1}$ for each edge separately. Then the two tangential components on faces can be determined by L^2 projection separately for each face.

We mention an alternative (and more expensive) method of imposing boundary conditions that led to some stability problems in practice. Namely, in (75) we could simply take $\boldsymbol{r}_n^{n+1}$ to be the L^2 projection of $\nabla \mathcal{E}_m q_h^{n+1}$ into X_h.

Pressure-approximation schemes. Discretization of the pressure-approximation schemes from Section 3.2 by C^0 finite elements is based on the weak form Eq. (19) for pressure, as was used in [16,20]. With $\boldsymbol{h}_h^j = \boldsymbol{u}_{*h}^j \cdot \nabla \boldsymbol{u}_{*h}^j$ and writing

$$\bar{\boldsymbol{f}}_h = \boldsymbol{f}^{\,n+1} - \mathcal{E}_k \boldsymbol{h}_h^{n+1} - \frac{1}{\Delta t}\sum_{j\geqslant 1} \alpha_j \boldsymbol{u}_{*h}^{n+1-j}, \tag{77}$$

we require $\bar{P}_h \in Y_h$ to satisfy

$$\langle \nabla \bar{P}_h, \nabla \psi_h\rangle = \langle \nu \nabla \times \mathcal{E}_m \boldsymbol{u}_{*h}^{n+1}, \boldsymbol{n}\times \nabla \psi_h\rangle_\Gamma - \frac{\alpha_0}{\Delta t}\langle \boldsymbol{n}\cdot \boldsymbol{g}^{n+1}, \psi_h\rangle + \langle \bar{\boldsymbol{f}}_h, \nabla \psi_h\rangle \tag{78}$$

for all $\psi_h \in Y_h$. The momentum equations read

$$\frac{\alpha_0}{\Delta t}\langle \boldsymbol{u}_{*h}^{n+1}, \boldsymbol{v}_h\rangle + \langle \nabla \bar{P}_h, \boldsymbol{v}_h\rangle + \nu\langle \nabla \boldsymbol{u}_{*h}^{n+1}, \nabla \boldsymbol{v}_h\rangle = \langle \bar{\boldsymbol{f}}_h, \boldsymbol{v}_h\rangle \quad \forall \boldsymbol{v}_h \in X_{0,h}, \tag{79}$$

with the boundary conditions $\boldsymbol{u}_{*h}^{n+1} = \boldsymbol{g}^{n+1}$ on Γ_h.

5. Numerical tests in 1D

5.1. Single-mode Stokes flow in a periodic strip

To study stability and accuracy for the simplest kind of incompressible flows, we first consider the unsteady Stokes equations in the strip $-1 < x < 1, y \in \mathbb{R}$, with boundary conditions at $x = \pm 1$:

$$\partial_t \boldsymbol{u} + \nabla p_S(\boldsymbol{u}) = \Delta \boldsymbol{u} + \boldsymbol{f}, \quad \boldsymbol{u}|_{x=\pm 1} = \boldsymbol{g}, \quad \boldsymbol{u}|_{t=0} = \boldsymbol{u}_0. \tag{80}$$

We set the force and boundary velocity to zero:

$$\boldsymbol{f} = 0, \quad \boldsymbol{g} = 0, \tag{81}$$

and look for normal-mode solutions that we write in the form

$$\boldsymbol{u}(t,x,y) = e^{i\xi y - \sigma t}(u(x,\xi), iv(x,\xi)). \tag{82}$$

With

$$\mu = \sqrt{\xi^2 - \sigma}, \tag{83}$$

the equations reduce to the system

$$(\partial_x^2 - \mu^2)u = \partial_x p, \quad u|_{x=\pm 1} = 0, \tag{84}$$

$$(\partial_x^2 - \mu^2)v = \xi p, \quad v|_{x=\pm 1} = 0, \tag{85}$$

$$(\partial_x^2 - \xi^2)p = 0, \quad (\partial_x p - \xi \partial_x v)|_{x=\pm 1} = 0. \tag{86}$$

We see $p = c_1 \sinh \xi x + c_2 \cosh \xi x$. If for definiteness we take p as anti-symmetric,

$$p = \sinh \xi x, \tag{87}$$

we find

$$u(x) = A\left(\frac{\cosh \xi x}{\cosh \xi} - \frac{\cosh \mu x}{\cosh \mu}\right), \quad A = \frac{\xi \cosh \xi}{\xi^2 - \mu^2}, \tag{88}$$

$$v(x) = B\left(\frac{\sinh \xi x}{\sinh \xi} - \frac{\sinh \mu x}{\sinh \mu}\right), \quad B = \frac{\xi \sinh \xi}{\xi^2 - \mu^2}. \tag{89}$$

Now, we compute that at $x = \pm 1$,

$$\partial_x p - \xi \partial_x v = \xi \cosh \xi - \frac{\xi^2 \sinh \xi}{\xi^2 - \mu^2}\left(\frac{\xi \cosh \xi}{\sinh \xi} - \frac{\mu \cosh \mu}{\sinh \mu}\right).$$

Imposing the boundary condition $\partial_x p - \xi \partial_x v = 0$ at $x = \pm 1$ and simplifying leads to

$$\xi \coth \xi - \frac{\xi^2}{\xi^2 - \mu^2}(\mu \cot \mu - \xi \coth \xi) = 0 \tag{90}$$

which can be simplified to read $\mu \tanh \mu = \xi \tanh \xi$. This equation indeed implies $\partial_x u - \xi v = 0$ by a simple calculation using (88) and (89). Since ξ is real, it is not hard to see $\mu = i\hat{\mu}$ where $\hat{\mu}$ is real and

$$-\hat{\mu} \tan \hat{\mu} = \xi \tanh \xi. \tag{91}$$

In the numerical tests below, we will take $\xi = 1, \hat{\mu} = 2.883355658589349$, so that $\hat{\mu} \tan \hat{\mu} + \xi \tanh \xi \approx 0$ numerically.

5.2. Single-mode stability tests

To investigate the stability of fully discrete k-step schemes, we write the schemes with $\boldsymbol{f} = 0$ and $\boldsymbol{g} = 0$ as

$$A_0 \boldsymbol{u}^{n+1} + A_1 \boldsymbol{u}^n + \cdots + A_k \boldsymbol{u}^{n+1-k} = 0. \tag{92}$$

Looking for a normal-mode solution $\boldsymbol{u}^n = \kappa^n \boldsymbol{u}$, we require $(\sum_{j=0}^{k} A_j \kappa^{-j})\boldsymbol{u} = 0$ which is a polynomial eigenvalue problem for $z = \kappa^{-1}$. This can be rewritten as a generalized eigenvalue problem as usual – e.g., for a three-step scheme, $k = 2$ and we require

$$\begin{pmatrix} 0 & I & 0 \\ 0 & 0 & I \\ -A_3 & -A_2 & -A_1 \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ \kappa \boldsymbol{u} \\ \kappa^2 \boldsymbol{u} \end{pmatrix} = \kappa \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & A_0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ \kappa \boldsymbol{u} \\ \kappa^2 \boldsymbol{u} \end{pmatrix}. \tag{93}$$

The matrices A_j depend on Δt as well as the finite elements being used.

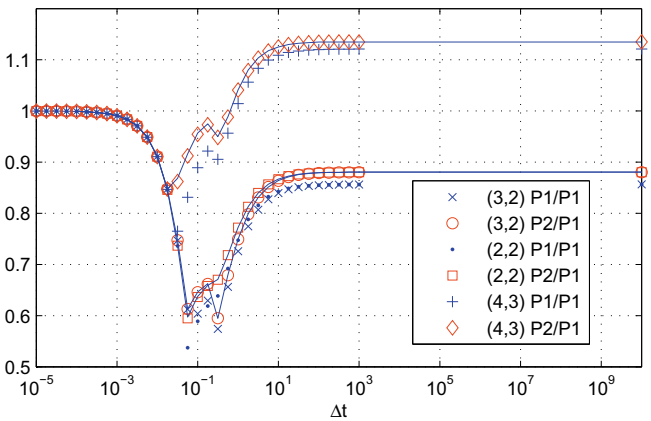

Fig. 1. Largest magnitude of eigenvalue vs. Δt. PA scheme. Thirty elements for each variable. Solid lines are theoretical results from Appendix A.

We calculated the eigenvalues of largest magnitude for the generalized eigenvalue problem (93) as a function of Δt for fixed $\xi = 1$, using the Matlab function `eigs`. The results are plotted in Figs. 1–3 and are discussed below. We used a range of time steps varying from large ($\Delta t = 10^{10}$) to small ($\Delta t = 10^{-5}$). The solid curves in Figs. 1 and 2 are determined by the space-continuous normal-mode theory of Appendix A.

5.2.1. PA and SC schemes with $m = 2$

The results for the (2,2) and (3,2) PA and SC schemes indicate that the eigenvalues always have magnitude less than 1. We found this result insensitive to spatial resolution, and it holds with various finite-element pairs for spatial approximation that were tested. This includes piecewise-polynomial approximations of equal order for both velocity and pressure, including piecewise-linear elements (P1/P1). Similar results were found also for piecewise quadratic (P2/P2) and quartic (P4/P4) elements. We will comment on the relation of these findings to the standard inf–sup stability condition in the Conclusions section below.

This suggests unconditional stability for the (3,2) PA and SC schemes, which involve reduced-order extrapolation of pressure or slip-correction terms. The result for the (3,2) PA finite-element scheme is consistent with the results reported by Leriche et al. [21], in a square 2D domain with spectral collocation in space, using a '(3,2) HOS scheme' that is equivalent to the (3,2) PA scheme at the time-discrete level for the Stokes equations.

5.2.2. PA and SC schemes with $m = 3$

We also tested (4,3) and (3,3) PA and SC schemes, which have unstable eigenmodes with $|\kappa| > 1$ when Δt large. Only the results for (4,3) schemes are shown, since the (3,3) results are quite similar. Again we found the results rather insensitive to spatial resolution and the type of finite-element discretization.

The unstable eigenmodes were found to appear smooth and turn out to fit rather well a theory of normal modes for a space-continuous version of (92) for which an explicit dispersion relation can be written that relates κ to Δt and ξ. See Appendix A for the details.

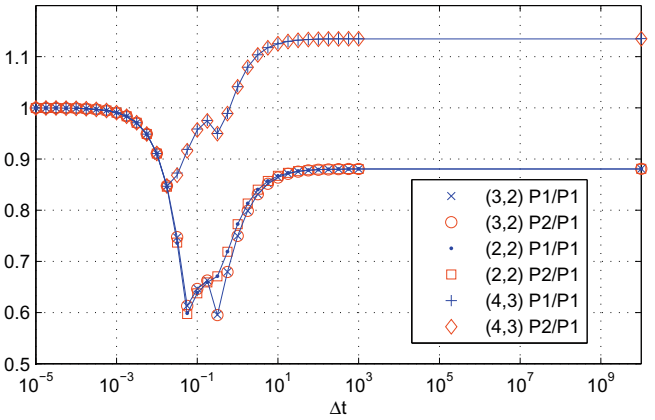

Fig. 2. Largest magnitude of eigenvalue vs. Δt. SC scheme. Thirty elements for each variable. Solid lines are theoretical results from Appendix A.

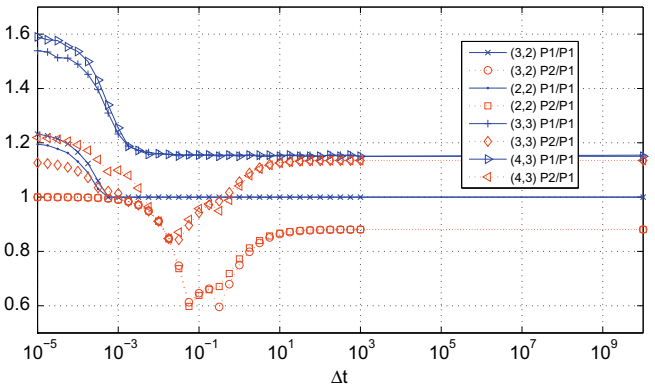

Fig. 3. Largest magnitude of eigenvalue vs. Δt. PU scheme. Thirty elements for each variable. Lines are interpolated to aid visualization.

The theory indicates that for this single-mode problem, the (4,3) and (3,3) time-discrete schemes are stable for time steps Δt less than a critical value Δt_c independent of wave number ξ. Our numerical results suggest that this holds independent of spatial resolution. This means that these schemes in 1D *do not* appear to be subject to a stability restriction of diffusive type like $\Delta t \leqslant Ch^2$, which becomes more restrictive as the grid is refined.

For the PA schemes this finding is not consistent with the corresponding results of [21] for 2D square domains, where instability for all time steps was found for (4,3) and (3,3) HOS schemes. It is possible that instability for these schemes is associated with presence of corners, so we performed numerical tests in a 2D domain with smooth boundary (a ring domain) that are described in the next section.

5.2.3. PU schemes

The results for pressure-update (PU) schemes are reported in Fig. 3 and have a different character. Of course, one has to augment Eq. (93) with pressure variables for PU schemes. The (2,2) and (3,2) PU schemes appear unconditionally stable only for the P2/P1 velocity/pressure finite-element pair, which satisfies the standard inf–sup condition. With P1/P1 elements, these schemes are unstable for small Δt, and almost neutrally stable ($|\kappa_{\max}| \approx 1$) for larger Δt, with neutral modes dominated by high-frequency oscillations in the pressure.

The (3,3) and (4,3) PU schemes are always unstable with P1/P1 elements. With P2/P1 elements, however, these schemes exhibit a *window* of stability, with instability for both small Δt and for large Δt. The lower threshhold for stability appears to get smaller as the spatial grid is refined, in a way we did not analyze. (The (3,3) PU scheme was described and tested using finite differences by Ren et al. [25].)

5.3. Single-mode accuracy tests

We checked the accuracy of various finite-element schemes using an explicitly specified smooth solution

$$(u, v, p) = g(t)e^{iky}(u(x), iv(x), p(x)), \quad g(t) = \cos(t), \tag{94}$$

where $u(x)$, $v(x)$, and $p(x)$ are given by (88), (89) and (87), respectively. The computational domain for x is $[0.1, 0.9]$. The forcing functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are determined so that the Stokes Eq. (80) hold.

Temporal accuracy. In Tables 2 and 3, we take time steps $\Delta t = 0.02/2^k$ for $k = 0$ to 4 and integrate to $T = 2$ to do a temporal accuracy check. We use P5 finite elements for both velocity and pressure. We refine the grid when reducing the time step so that $\Delta t/h$ remains constant ($= 1$), to make spatial errors less than temporal errors. The main quantity tabulated in all tables is $-\log_{10}E$, where E is the quantity listed in the left-hand column. (This indicates the number of essentially correct digits in the approximation.) In parentheses we also list the local convergence rate α for E. In Tables 3 and 2 this is determined from the formula

$$\alpha = \frac{\log_{10}(E_{k-1}/E_k)}{\log_{10}(\Delta t_{k-1}/\Delta t_k)} = \frac{\log_{10}E_{k-1} - \log_{10}E_k}{0.3010\ldots}. \tag{95}$$

(Note $\log_{10}2 \approx 0.3010$. Values of α in the first column of the tables are based on values of E for a larger time step not shown.) We only show results for the divergence-free approximate velocity $\boldsymbol{u}_h = (u_h, v_h)$. We use p_h to denote the pressure obtained from solving a discrete version of (20) and (21), and will use $\hat{p}_h$ to denote the approximate pressure obtained without solving any further Poisson equations – i.e., $\hat{p}_h$ is based on (42) for SC schemes or (63) for PA schemes. Since discretization of these formulas yields discontinuities across element boundaries due to the divergence term, we L^2-project the result into the finite-element pressure space to facilitate computation of gradient errors. (In a number of cases we checked that one gets essentially the same results by computing errors elementwise without projecting.)

The results of the temporal accuracy check all show third-order convergence in time consistent with the formal analysis, except for $\nabla\hat{p}_h$ in the SC scheme at the finest resolution.

Spatial accuracy. Tables 4–7 contain results of tests of spatial accuracy using uniform grids with element size $h = 0.02/2^k$ for $k = 0, 1, 2, 3, 4$, with time step $\Delta t = h^{3/2}$ to minimize temporal error, and integrate to time $T = 1$. We use P2/P1 finite

Table 2
Temporal accuracy of the (3,2) PA scheme in 1D. $-\log_{10}E$ (and local order α) vs. Δt. P5/P5 FE, $T = 2, \Delta t = h$.

$E \setminus \Delta t$	0.01	0.005	0.0025	0.00125
$\|u - u_h\|_{L^\infty}$	7.2 (2.95)	8.1 (2.97)	9 (2.98)	9.9 (2.99)
$\|\nabla(u - u_h)\|_{L^\infty}$	6.08 (2.95)	6.98 (2.97)	7.87 (2.98)	8.78 (2.99)
$\|v - v_h\|_{L^\infty}$	6.08 (2.95)	6.98 (2.97)	7.87 (2.98)	8.78 (2.99)
$\|\nabla(v - v_h)\|_{L^\infty}$	5.19 (2.95)	6.08 (2.97)	6.98 (2.98)	7.88 (2.98)
$\|p - p_h\|_{L^\infty}$	4.77 (2.95)	5.67 (2.97)	6.56 (2.98)	7.49 (3.08)
$\|\nabla(p - p_h)\|_{L^\infty}$	5.19 (2.95)	6.08 (2.97)	6.98 (2.98)	7.88 (2.98)
$\|p - \hat{p}_h\|_{L^\infty}$	4.77 (2.95)	5.67 (2.97)	6.56 (2.98)	7.46 (2.98)
$\|\nabla(p - \hat{p}_h)\|_{L^\infty}$	5.19 (2.95)	6.09 (2.97)	6.98 (2.98)	7.72 (2.43)

Table 3
Temporal accuracy of the (3, 2) SC scheme in 1D. $-\log_{10}E$ (and local order α) vs. Δt. P5/P5 FE, $T = 2, \Delta t = h$.

$E \setminus \Delta t$	0.01	0.005	0.0025	0.00125
$\|u - u_h\|_{L^\infty}$	7.67 (2.87)	8.55 (2.92)	9.44 (2.95)	10.3 (2.98)
$\|\nabla(u - u_h)\|_{L^\infty}$	6.69 (2.89)	7.57 (2.93)	8.46 (2.95)	9.35 (2.96)
$\|v - v_h\|_{L^\infty}$	6.69 (2.9)	7.57 (2.94)	8.46 (2.96)	9.36 (2.97)
$\|\nabla(v - v_h)\|_{L^\infty}$	5.92 (2.92)	6.81 (2.95)	7.7 (2.97)	8.6 (2.97)
$\|p - p_h\|_{L^\infty}$	5.58 (2.95)	6.48 (2.97)	7.38 (2.98)	8.51 (3.78)
$\|\nabla(p - p_h)\|_{L^\infty}$	5.92 (2.92)	6.81 (2.95)	7.7 (2.97)	8.59 (2.97)
$\|p - \hat{p}_h\|_{L^\infty}$	5.59 (2.95)	6.48 (2.97)	7.38 (2.97)	8.25 (2.92)
$\|\nabla(p - \hat{p}_h)\|_{L^\infty}$	5.91 (2.91)	6.8 (2.93)	7.68 (2.92)	8.06 (1.28)

Table 4
Spatial accuracy of the (3, 2) PA scheme in 1D. $-\log_{10}E$ (and local order α) vs. h. P2/P1 FE, $T = 1, \Delta t = h^{1.5}$.

$E \setminus h$	0.8/80	0.8/160	0.8/320	0.8/640
$\|u - u_h\|_{L^\infty}$	7.74 (3.16)	8.68 (3.11)	9.61 (3.08)	10.5 (3.05)
$\|\nabla(u - u_h)\|_{L^\infty}$	4.94 (2.01)	5.54 (2)	6.14 (2)	6.75 (2)
$\|v - v_h\|_{L^\infty}$	6.98 (3.29)	7.97 (3.27)	8.94 (3.25)	9.91 (3.21)
$\|\nabla(v - v_h)\|_{L^\infty}$	4.49 (2.04)	5.1 (2.01)	5.7 (2)	6.31 (2)
$\|p - p_h\|_{L^\infty}$	3.88 (2.02)	4.49 (2)	5.09 (2)	5.69 (2)
$\|\nabla(p - p_h)\|_{L^\infty}$	2.66 (1.02)	2.97 (1.01)	3.27 (1)	3.57 (1)
$\|p - \hat{p}_h\|_{L^\infty}$	5.44 (2.75)	5.98 (1.82)	6.56 (1.9)	7.15 (1.97)
$\|\nabla(p - \hat{p}_h)\|_{L^\infty}$	2.67 (0.999)	2.97 (0.998)	3.27 (1)	3.57 (1)

Table 5
Spatial accuracy of the (3, 2) SC scheme in 1D. $-\log_{10}E$ (and local order α) vs. h. P2/P1 FE, $T = 1, \Delta t = h^{1.5}$.

$E \setminus h$	0.8/80	0.8/160	0.8/320	0.8/640
$\|u - u_h\|_{L^\infty}$	4 (1.68)	4.51 (1.69)	5.02 (1.7)	5.54 (1.71)
$\|\nabla(u - u_h)\|_{L^\infty}$	1.49 (0.671)	1.7 (0.683)	1.91 (0.696)	2.12 (0.705)
$\|v - v_h\|_{L^\infty}$	6.71 (2.74)	7.54 (2.74)	8.36 (2.75)	9.18 (2.72)
$\|\nabla(v - v_h)\|_{L^\infty}$	4.05 (1.78)	4.58 (1.78)	5.12 (1.78)	5.65 (1.77)
$\|p - p_h\|_{L^\infty}$	3.88 (1.99)	4.48 (1.99)	5.08 (2)	5.68 (2)
$\|\nabla(p - p_h)\|_{L^\infty}$	2.66 (1.02)	2.97 (1.01)	3.27 (1.01)	3.57 (1)
$\|p - \hat{p}_h\|_{L^\infty}$	5.35 (1.96)	5.95 (1.97)	6.55 (1.99)	7.15 (1.99)
$\|\nabla(p - \hat{p}_h)\|_{L^\infty}$	2.67 (0.998)	2.97 (0.997)	3.27 (0.999)	3.57 (0.999)

Table 6
Spatial accuracy of the (3, 2) PA scheme in 1D. $-\log_{10}E$ (and local order α) vs. h. P2/P2 FE, $T = 1, \Delta t = h^{1.5}$.

$E \setminus h$	0.8/80	0.8/160	0.8/320	0.8/640
$\|u - u_h\|_{L^\infty}$	7.75 (3.15)	8.68 (3.11)	9.61 (3.08)	10.5 (3.05)
$\|\nabla(u - u_h)\|_{L^\infty}$	4.94 (2.01)	5.54 (2)	6.14 (2)	6.75 (2)
$\|v - v_h\|_{L^\infty}$	6.98 (3.29)	7.96 (3.27)	8.94 (3.25)	9.91 (3.21)
$\|\nabla(v - v_h)\|_{L^\infty}$	4.49 (2.04)	5.1 (2.01)	5.7 (2)	6.31 (2)
$\|p - p_h\|_{L^\infty}$	3.89 (2.02)	4.49 (2.01)	5.1 (2)	5.7 (2)
$\|\nabla(p - p_h)\|_{L^\infty}$	4.26 (2.02)	4.87 (2.01)	5.47 (2)	6.07 (2)
$\|p - \hat{p}_h\|_{L^\infty}$	4.7 (2.16)	5.32 (2.06)	5.93 (2.02)	6.53 (2.01)
$\|\nabla(p - \hat{p}_h)\|_{L^\infty}$	2.01 (1)	2.31 (1)	2.61 (1)	2.92 (1)

Table 7
Spatial accuracy of the (3, 2) SC scheme in 1D. $-\log_{10}E$ (and local order α) vs h. P2/P2 FE, $T = 1$, $\Delta t = h^{1.5}$.

$E \setminus h$	0.8/80	0.8/160	0.8/320	0.8/640
$\|u - u_h\|_{L^\infty}$	5.31 (1.94)	5.89 (1.94)	6.48 (1.95)	7.07 (1.96)
$\|\nabla(u - u_h)\|_{L^\infty}$	2.6 (0.938)	2.89 (0.94)	3.17 (0.948)	3.46 (0.954)
$\|v - v_h\|_{L^\infty}$	7.41 (3)	8.31 (3)	9.21 (3)	10.1 (3)
$\|\nabla(v - v_h)\|_{L^\infty}$	4.5 (2)	5.1 (2)	5.7 (2)	6.31 (2)
$\|p - p_h\|_{L^\infty}$	3.89 (1.99)	4.49 (1.99)	5.09 (2)	5.69 (2)
$\|\nabla(p - p_h)\|_{L^\infty}$	4.29 (1.98)	4.88 (1.99)	5.48 (1.99)	6.08 (1.99)
$\|p - \hat{p}_h\|_{L^\infty}$	3.7 (1.13)	4.05 (1.16)	4.41 (1.18)	4.77 (1.2)
$\|\nabla(p - \hat{p}_h)\|_{L^\infty}$	0.997 (0.119)	1.04 (0.154)	1.1 (0.178)	1.15 (0.193)

elements respectively for velocity and pressure in Tables 4 and 5, and use P2 elements for both velocity and pressure in Tables 6 and 7.

For the (3,2) PA scheme with P2/P1 elements, all errors in Table 4 exhibit the same convergence rate as interpolation. For the (3,2) SC scheme, however, the velocity errors in Table 5 exhibit fine-scale oscillations near the boundary, particularly the horizontal velocity, and the order of convergence is degraded. See the error plots in Fig. 4.

Results using P2 elements for all variables are in Tables 6 and 7. Note that while the pressure gradient ∇p_h exhibits second-order accuracy in space (the same order as interpolation error for P2 elements), the pressure p_h itself exhibits only the same second-order accuracy, which is one order less than interpolation error. This may be due to a breakdown of the typical duality argument for optimal approximation in the elliptic problem (10) that determines the Stokes pressure. The sharp estimate from [19] in (22) indicates that the Stokes pressure gradient is a second-order operator on the velocity field, and consequently the pressure is a first-order operator. The second-order accuracy in space for pressure with P2 elements is consistent with this similarity to velocity gradients. We might then expect the pressure gradient to exhibit first-order accuracy (like second derivatives of velocity) instead of the observed second-order accuracy.

For the (3,2) PA scheme with P2/P2 elements, we remark that the error $p - \hat{p}_h$ is smaller than the error $p - p_h$. But the former is dominated by grid-scale oscillations, resulting in gradient errors of similar magnitude.

The spatial convergence rates for the (3,2) SC scheme with the P2/P2 finite-element pair are less than optimal for horizontal velocity and especially its gradient. The error $\boldsymbol{u} - \boldsymbol{u}_h$ appears to be dominated by high-frequency grid-scale oscillations near the boundary. The pressure error $p - \hat{p}_h$ based on the approximation in (42) is also dominated by oscillations in this case. Note that only the vertical velocity v_h is corrected by the projection step in this 1D problem—the horizontal velocity is not affected by slip-correction.

6. Numerical tests in 2D

In this section, we report the results of tests on the stability, accuracy and benchmark performance of the PA and SC finite-element schemes described in Section 3 for several basic 2D examples. All the finite-element computations are performed using Lagrange piecewise polynomial isoparametric C^0 finite elements of equal order for both velocity and pressure [27,11]. For benchmark problems including driven cavity flow, flow over a backward-facing step and flow past a cylinder, for the most part we use piecewise quadratic (P2) elements.

Our stability and accuracy tests all use the same explicitly specified smooth solution, given by

$$u_{\text{ex}} = \cos(t)\cos^2(\pi x/2)\sin(\pi y), \tag{96}$$

$$v_{\text{ex}} = -\cos(t)\sin(\pi x)\cos^2(\pi y/2), \tag{97}$$

$$p_{\text{ex}} = \cos(t)\cos(\pi x/2)\sin(\pi y/2). \tag{98}$$

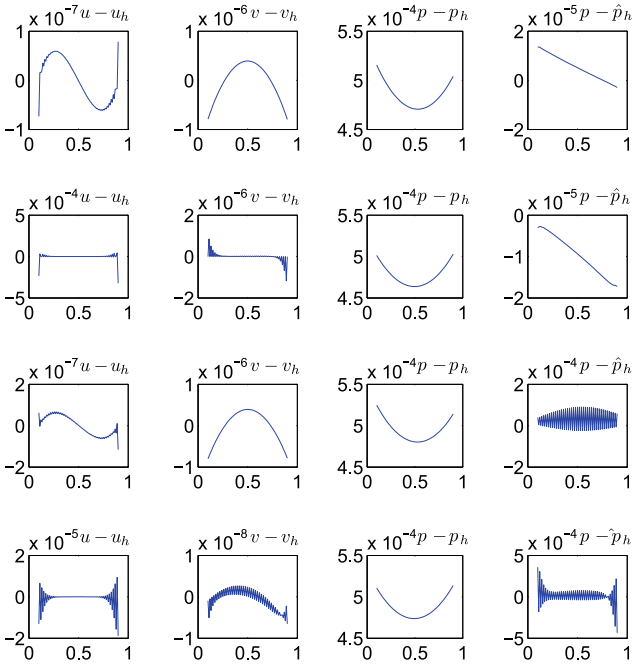

Fig. 4. From top to bottom: error plots for (3,2) PA-P2/P1, (3,2) SC-P2/P1, (3,2) PA-P2/P2, and (3,2) SC-P2/P2 schemes. Forty elements for each variable. $\Delta t = h^{1.5} = 0.0028$, $T = 1$.

We take $\nu = 1$, and use forcing $\boldsymbol{f}$ and boundary values $\boldsymbol{g}$ for velocity determined as necessary to yield the same exact solution in each case.

The finite-element package we have implemented is in some sense an upgraded version of iFEM due to Long Chen (GNU Licensed, see http://math.uci.edu/~chenlong/iFEM.html). 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 [8,1]). In particular, local refinement and coarsening can be done fairly easily. For our purposes, we have extended it to isoparametric Lagrange elements up to P5. The finite-element mesh is generated using DistMesh of Persson and Strang [23]. The contour plots on unstructured meshs are generated by the MATLAB routine `tricontour.m` due to Engwirda [10].

6.1. Stability

We checked the nonlinear stability of several PA and SC finite element schemes by integrating the full NSE to time $T = 10,000$. The domain (a square with hole) and mesh used are shown in the left half of Fig. 5. If any element has an edge on the circle, it is an isoparametric element. We used 324 P2 finite elements (dof = 740) for each variable.

With (3,2) PA and SC schemes, the computations were observed to remain stable to time $T = 10,000$, for values of Δt as large as 8 with errors no larger than $O(1)$. Similar results were obtained for these schemes with P1 (piecewise linear) and with P4 finite elements.

The (3,3) and (4,3) PA and SC schemes in a square with hole were observed to have a time-step restriction for stability of possibly diffusive type. See Table 8.

For (3,3) and (4,3) PA and SC schemes in the annulus, we calculated the eigenvalues of largest magnitude using the Matlab function `eigs`, adapting the formalism of Section 5.2 to the fully discrete scheme in 2D. The largest magnitude was observed to be less that 1 for Δt less than a critical value Δt_c that depends weakly on the mesh size, in the way reported in Table 9. These results suggest that the phenomenon observed for these schemes in 1D, namely stability for small time steps independent of mesh size, may indeed hold also for smooth 2D domains.

6.2. Temporal accuracy

We perform temporal accuracy checks in two different domains: an annulus, and a square with a hole. One has smooth boundary, and the other has corners. We use P4 isoparametric finite elements. The coarsest meshes used are pictured in Fig. 5. These meshes were used with $\Delta t = 0.04$ for Tables 10–12. When Δt is reduced by half, one triangle breaks into 4 triangles.

Third-order schemes. We only show results for the (3,2) SC scheme, since results for the PA scheme are similar. See Tables 10 and 11. The pressure gradient shows slightly degraded accuracy in the square with hole. In Fig. 6 we show a mesh plot of the pressure error in the square with hole for the (3,2) PA and SC schemes. One sees steep gradients near the corners, where the formal analysis of Section 3 evidently breaks down. Max-norm errors are strongly affected by behavior in the corner, and for this reason we tabulate L^2 norms for pressure error and its gradient in Tables 10 and 11.

Second-order schemes. For purposes of comparison, in Table 12 we provide results of an accuracy check for a (2,1) SC scheme, which is formally second-order accurate in time. We are not aware of previously published results using such a finite-element method, which is based on a time discretization close to the original Kim–Moin finite-difference scheme described for a staggered grid in [18]. Comparison of Tables 11 and 12 indicates that for this smooth test problem the third-order schemes are substantially more accurate than the second-order $(2,1)$ SC scheme, with essentially the same cost. Note that here we have nonhomogeneous boundary conditions and are using P4 elements with an unstructured grid in a domain with corners and a hole.

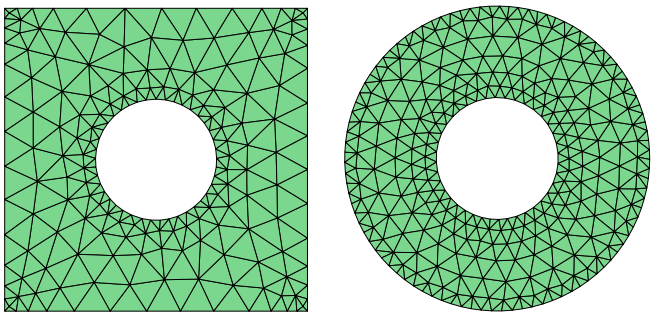

Fig. 5. Finite-element meshes for stability and temporal accuracy checks. Square with hole: $\{(x,y)|r^2 \leqslant x^2+y^2, |x|, |y| \leqslant R\}$. Annulus: $\{(x,y)|r^2 \leqslant x^2+y^2 \leqslant R^2\}$. $(r,R) = (0.2, 0.5)$.

Table 8
Largest time step for linear stability of PA schemes in a square with hole. N = number of refinements from grid in Fig. 5.

Δt_c \ N	0	1	2
(3,3) P1/P1	0.0448	0.0155	0.00067
(3,3) P2/P2	0.0002685	0.0000733	0.0000182
(4,3) P1/P1	0.0485	0.0168	0.00074
(4,3) P2/P2	0.0002909	0.0000805	0.0000200

Table 9
Largest time step for linear stability of PA schemes in annulus. N = number of refinements from grid in Fig. 5.

Δt_c \ N	0	1	2
(3,3) P1/P1	0.022	0.014	0.010
(3,3) P2/P2	0.007	0.005	0.005
(4,3) P1/P1	0.024	0.017	0.014
(4,3) P2/P2	0.011	0.011	0.010

Table 10
Temporal accuracy, $(3,2)$ SC scheme in annulus. $-\log_{10}E$ (and local order α) vs. Δt. P4 isoparametric FE, $T = 2$.

$E \setminus \Delta t$	0.02	0.01	0.005
$\|u - u_h\|_{L^\infty}$	5.76 (2.58)	6.66 (2.99)	7.57 (3.04)
$\|\nabla(u - u_h)\|_{L^\infty}$	4.06 (2.93)	5.01 (3.13)	5.94 (3.1)
$\|p - p_h\|_{L^2}$	4.18 (2.5)	5.06 (2.93)	5.96 (2.98)
$\|\nabla(p - p_h)\|_{L^2}$	3.68 (2.52)	4.56 (2.93)	5.45 (2.96)
$\|p - \hat{p}_h\|_{L^2}$	4.17 (2.52)	5.06 (2.94)	5.96 (2.98)
$\|\nabla(p - \hat{p}_h)\|_{L^2}$	3.63 (2.8)	4.53 (2.99)	5.4 (2.92)

Table 11
Temporal accuracy, $(3,2)$ SC scheme in square with hole. $-\log_{10}E$ (and local order α) vs. Δt. P4 isoparametric FE, $T = 2$.

$E \setminus \Delta t$	0.02	0.01	0.005
$\|u - u_h\|_{L^\infty}$	5.74 (3.64)	6.64 (2.99)	7.56 (3.04)
$\|\nabla(u - u_h)\|_{L^\infty}$	3.87 (3.44)	4.87 (3.31)	5.83 (3.2)
$\|p - p_h\|_{L^2}$	4.17 (3.71)	5.05 (2.94)	5.96 (3)
$\|\nabla(p - p_h)\|_{L^2}$	3.53 (3.58)	4.38 (2.82)	5.21 (2.77)
$\|p - \hat{p}_h\|_{L^2}$	4.16 (3.71)	5.05 (2.95)	5.96 (3)
$\|\nabla(p - \hat{p}_h)\|_{L^2}$	3.38 (3.5)	4.29 (3.03)	5.14 (2.82)

Table 12
Temporal accuracy, $(2,1)$ SC scheme in square with hole. $-\log_{10}E$ (and local order α) vs Δt. P4 isoparametric FE, $T = 2$.

$E \setminus \Delta t$	0.02	0.01	0.005
$\|u - u_h\|_{L^\infty}$	3.61 (1.9)	4.21 (1.99)	4.82 (2.04)
$\|\nabla(u - u_h)\|_{L^\infty}$	2.09 (1.92)	2.69 (1.99)	3.3 (2.03)
$\|p - p_h\|_{L^2}$	2.02 (1.91)	2.61 (1.98)	3.22 (2.02)
$\|\nabla(p - p_h)\|_{L^2}$	1.41 (1.84)	1.97 (1.87)	2.52 (1.84)
$\|p - \hat{p}_h\|_{L^2}$	2.02 (1.91)	2.61 (1.98)	3.22 (2.02)
$\|\nabla(p - \hat{p}_h)\|_{L^2}$	1.41 (1.84)	1.97 (1.87)	2.52 (1.83)

Fourth-order scheme in a smooth domain. We also provide results of an accuracy check for a (4,3) PA scheme in an annulus (which has smooth boundary) in Table 13. For stability reasons the largest time step we take is $\Delta t = 0.01$, smaller than that used in the previous tables. The coarsest mesh used is that in Fig. 5, used when $\Delta t = 0.01$. The results demonstrate better absolute accuracy than the (3,2) SC scheme in Table 10 for comparable step sizes, but they fall short of fourth-order accuracy, especially for $\nabla \hat{p}_h$, whose absolute accuracy is rather good but does not improve with refinement in this test. It appears that the error in $\hat{p}_h$ in Table 13 exhibits fine-scale oscillations like in Fig. 4 for the (3,2) PA scheme with P2/P2 elements. The error in p_h and ∇p_h decreases steadily, however.

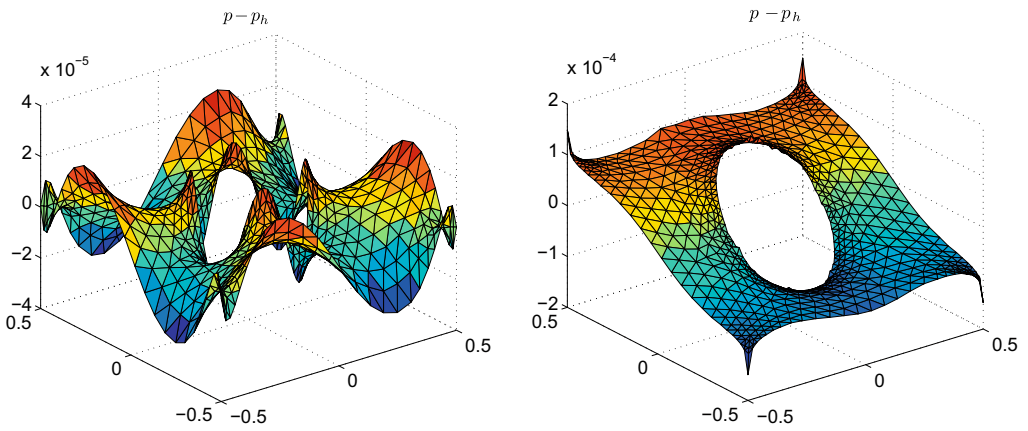

Fig. 6. Pressure error in the square with hole for the (3,2) PA (left) and SC (right) scheme. $\Delta t = 0.02$. 1296 P4 elements for each variable. $T = 2$. Only values at the vertices of triangles are used in the plots.

Table 13
Temporal accuracy, (4,3) PA scheme in annulus. $-\log_{10}E$ (and local order α) vs Δt. P5 isoparametric finite elements are used. $T = 2$.

$E \setminus \Delta t$	0.01	0.005	0.0025
$\|u - u_h\|_{L^\infty}$	8.42	9.52 (3.68)	10.6 (3.67)
$\|\nabla(u - u_h)\|_{L^\infty}$	6.97	8.03 (3.52)	9.07 (3.45)
$\|p - p_h\|_{L^2}$	7.59	8.71 (3.72)	9.85 (3.79)
$\|\nabla(p - p_h)\|_{L^2}$	6.39	7.56 (3.89)	8.66 (3.64)
$\|p - \hat{p}_h\|_{L^2}$	7.57	8.7 (3.74)	9.84 (3.81)
$\|\nabla(p - \hat{p}_h)\|_{L^2}$	6.12	6.58 (1.54)	5.83 (−2.5)

6.3. Benchmark tests with finite elements

In this subsection, we test our schemes on the benchmark problems of driven cavity flow (with $\nu = 1/1000$), flow over a backward-facing step (with $\nu = 1/100$ and $\nu = 1/600$) and flow past a cylinder (with $\nu = 1/1000$). We use P1 finite-element or P2 and P4 isoparametric finite-element discretization. To save space, we only show results for the $(3,2)$ SC and (3,2) PA schemes. We emphasize the (3,2) SC scheme since the performance of the (3,2) PA scheme is always at least as good.

For the driven cavity flow, we compute the flow in the domain $[0,1] \times [0,1]$ and start from rest, impulsively imposing horizontal velocity $u = 1$ on the top boundary for $t > 0$. Following [7], we plot the contours of vorticity with values $[-5, -4, -3, -2, -1, -0.5, 0, 0.5, 1, 2, 3]$ and the contours of pressure with values [0.3, 0.17, 0.12, 0.11, 0.09, 0.07, 0.05, 0.02, 0, −0.002]. the pressure is set to be zero at (0.5,0.5) which is the center of the cavity. (For the SC scheme, we report the pressure computed from the final velocity field using (19), with the convective form of the nonlinearity and not the rotational form which yields a different pressure.) The computational mesh used is a 64×64 stretched rectangular grid subdivided into triangles, one refinement of the grid shown in [20]. We refer to computational results of [18,7] for comparison. Although we use a rather coarse mesh, the vorticity and pressure contour plots agree quite well with [7].

For the backward-facing step, we compute the flow in the domain

$$\Omega = [0, L] \times [-0.5, 0.5] \setminus [0, 0.5] \times [-0.5, 0]$$

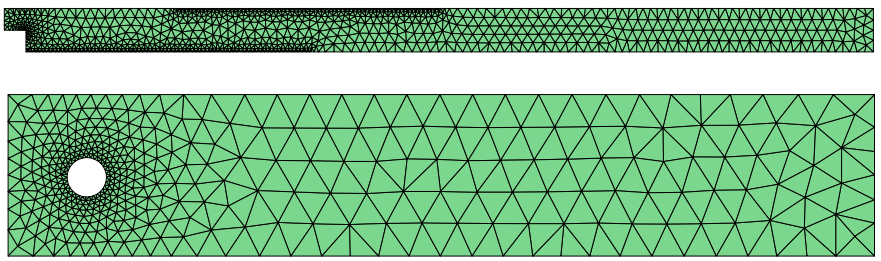

Fig. 7. Mesh used in backward-facing step flow computation when $\nu = 1/600$ and in flow past a cylinder calculation when $\nu = 1/1000$.

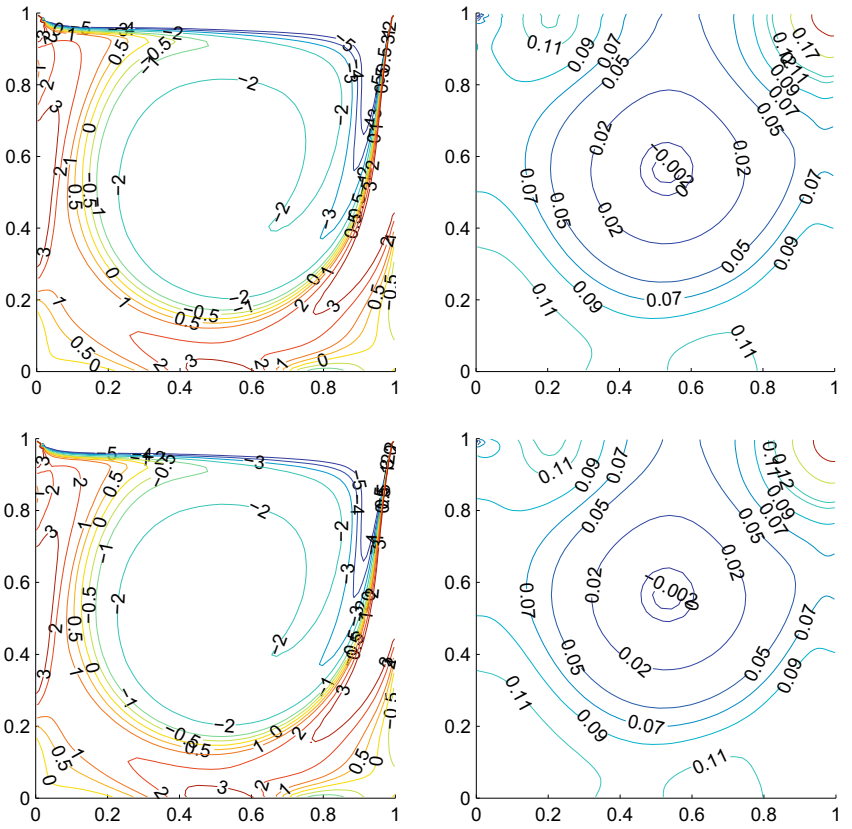

Fig. 8. Driven cavity, $\nu = 1/1000$. P1/P1 with 8192 P1 elements (dof = 4225) for each variable. $h_{\min} = 0.00594$, $h_{\max} = 0.0397$, $\Delta t = 0.006$, $T = 50$. Top: (3,2) SC scheme. Bottom: (3,2) PA scheme. From left to right: vorticity contour plots, pressure contour plots.

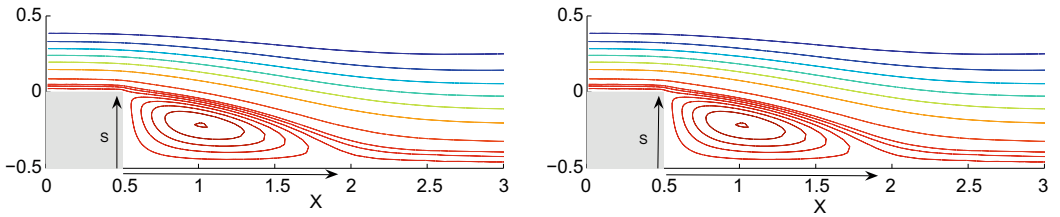

Fig. 9. Backward-facing step. $\nu = 1/100$. P1/P1 with 6640 P1 elements (dof = 3487) for each variable. $h_{\min} = 0.00783$, $h_{\max} = 0.116$, $\Delta t = 0.006$, $T = 20$. $X/S = 2.84$. Left: (3,2) SC scheme. Right: (3,2) PA scheme.

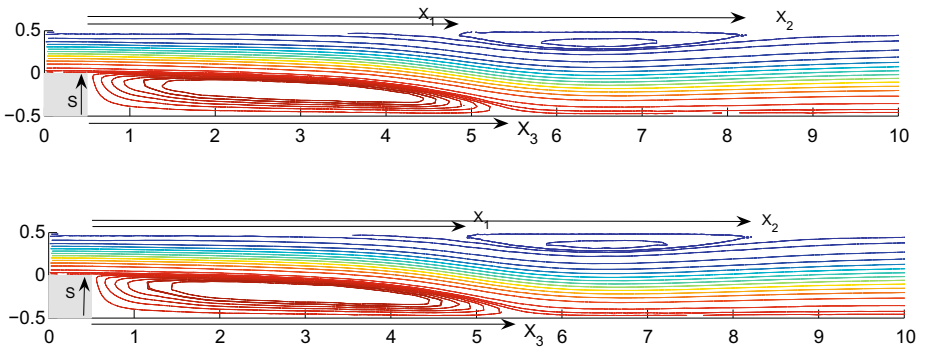

Fig. 10. Backward-facing step. $\nu = 1/600$. 1700 P2 elements (dof = 3925) for each variable. $h_{\min} = 0.0186$, $h_{\max} = 0.334$, $\Delta t = 0.003$, $T = 120$. Top: (3,2) SC scheme, $X_1/S = 8.86$, $X_2/S = 15.5$, $X_3/S = 9.9$. Bottom: (3,2) PA scheme, $X_1/S = 8.86$, $X_2/S = 15.55$, $X_3/S = 9.9$.

with no-slip boundary conditions everywhere except at the inflow boundary $x = 0$ and the outflow boundary $x = L$. We take $L = 8$ when $\nu = 1/100$, and take $L = 20$ when $\nu = 1/600$. But we will only show results near the step. We start from rest and gradually increase the boundary velocity (u, v) to $(12y(1-2y), 0)$ at the inflow boundary and $(-3y^2 + 3/4, 0)$ at the outflow boundary, with no net influx at each time. The time-dependent function we used for gradually increasing velocity is $(1 - \cos(\pi t))/2$ on $[0, 1]$. So, when t is large, the mean inflow velocity is 1 which leads to Re=$1/\nu$ when we use twice the step size as reference length. The computational mesh for $\nu = 1/600$ is shown in Fig. 7. Once the velocity field is obtained, we calculate the stream function and then show its contour plot. Once again, we obtain results that agree rather well with [2,18] using a rather coarse mesh.

For the flow past a cylinder, we follow the setup in [15]. Then the domain is $[0, 2.2] \times [0, 0.41] \backslash \{(x-0.2)^2 + (y-0.2)^2 \leqslant 0.05^2\}$. (Note that the hole is slightly off-center.) The time-dependent inflow and outflow profile (see Figs. 8–10)

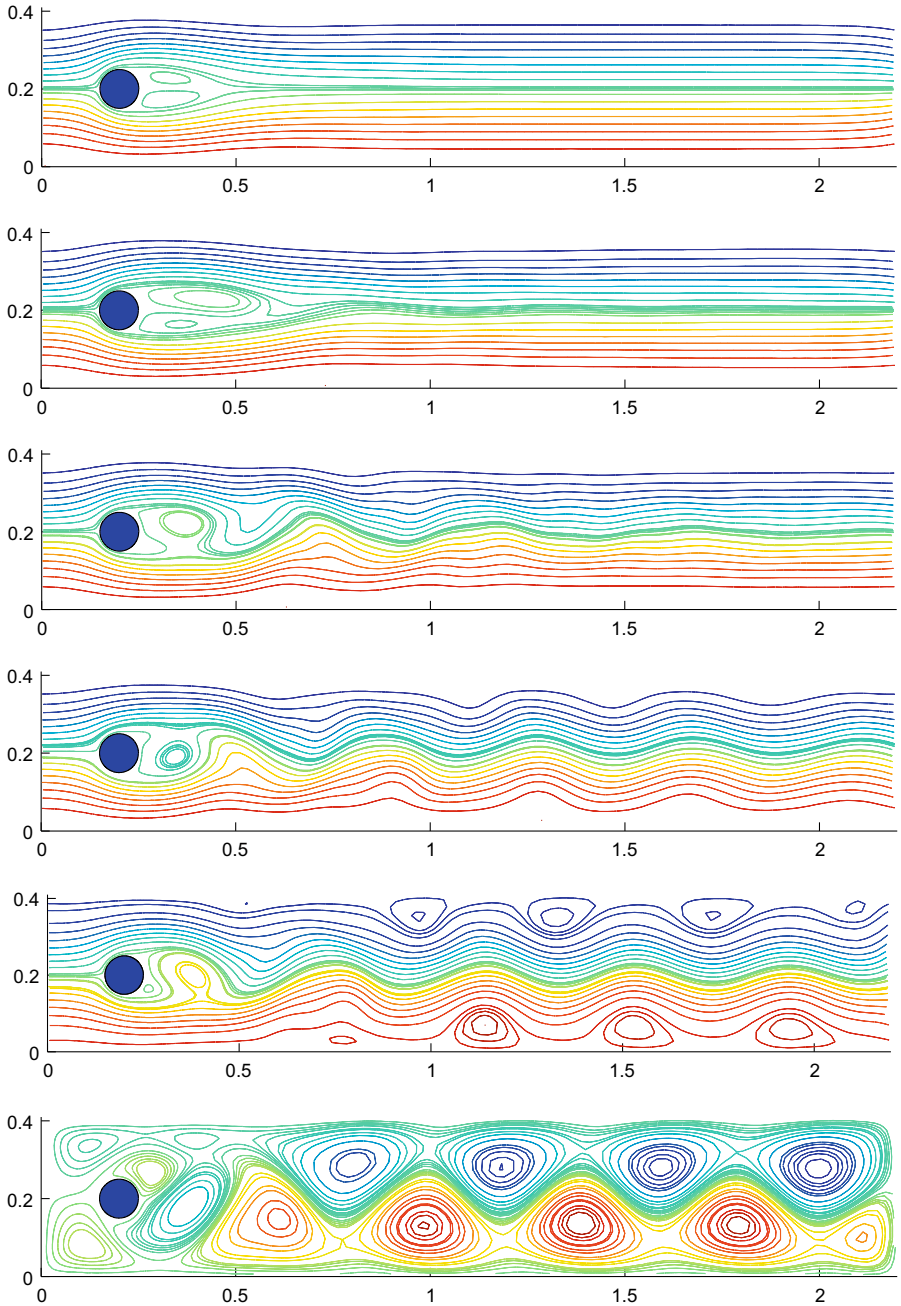

Fig. 11. Flow past a cylinder. $\nu = 1/1000$. 763 isoparametric P4 elements (dof = 6322) for each variable. The velocity at $t = [2, 4, 5, 6, 7, 8]$. $\Delta t = 0.0004$. $h_{\min} = 0.00822$. $h_{\max} = 0.117$. (3,2) SC scheme.

$$\boldsymbol{u}(t,0,y) = \boldsymbol{u}(t,2.2,y) = 0.41^{-2}\sin(\pi t/8)(6y(0.41-y),0) \tag{99}$$

is prescribed. v is chosen to be $1/1000$. 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. 7 and the contour plot of the stream function at t=[2,4,5,6,7,8] is shown in Fig. 11.

For comparison with [15] we also calculate the drag and lift coefficients, denoted by $c_d(t)$ and $c_l(t)$, which are the x and y components of the quantity

$$\frac{2}{LU_{\max}^2}\int_S v\partial_n\boldsymbol{u} - p\,\boldsymbol{n}, \tag{100}$$

where S is the surface of the cylinder. Since our goal is to test the scheme, we faithfully calculate these quantities by surface integration, instead of transforming them into volume integrals, which is known to be more accurate. We also calculate when the maxima of c_d and c_l occur, and compute the pressure difference between the front and the back of the cylinder

$$\Delta p(t) = p(t,0.15,0.2) - p(t,0.25,0.2). \tag{101}$$

Since we use rotational form for the nonlinear term in the SC calculations, the pressure we obtained is different from the standard pressure by $\frac{1}{2}|\boldsymbol{u}|^2$. But because $\boldsymbol{u}$ vanishes on the cylinder surface, we have used our pressure directly. As we have mentioned, we do not need to solve an extra Poisson equation to obtain this pressure.

The results for both the (3,2) SC and PA schemes with P4 isoparametric elements are shown in Fig. 12. Agreement with the reference results of [15] appears good, given that we use a grid roughly comparable to the coarsest grid (level 1) used in [15].

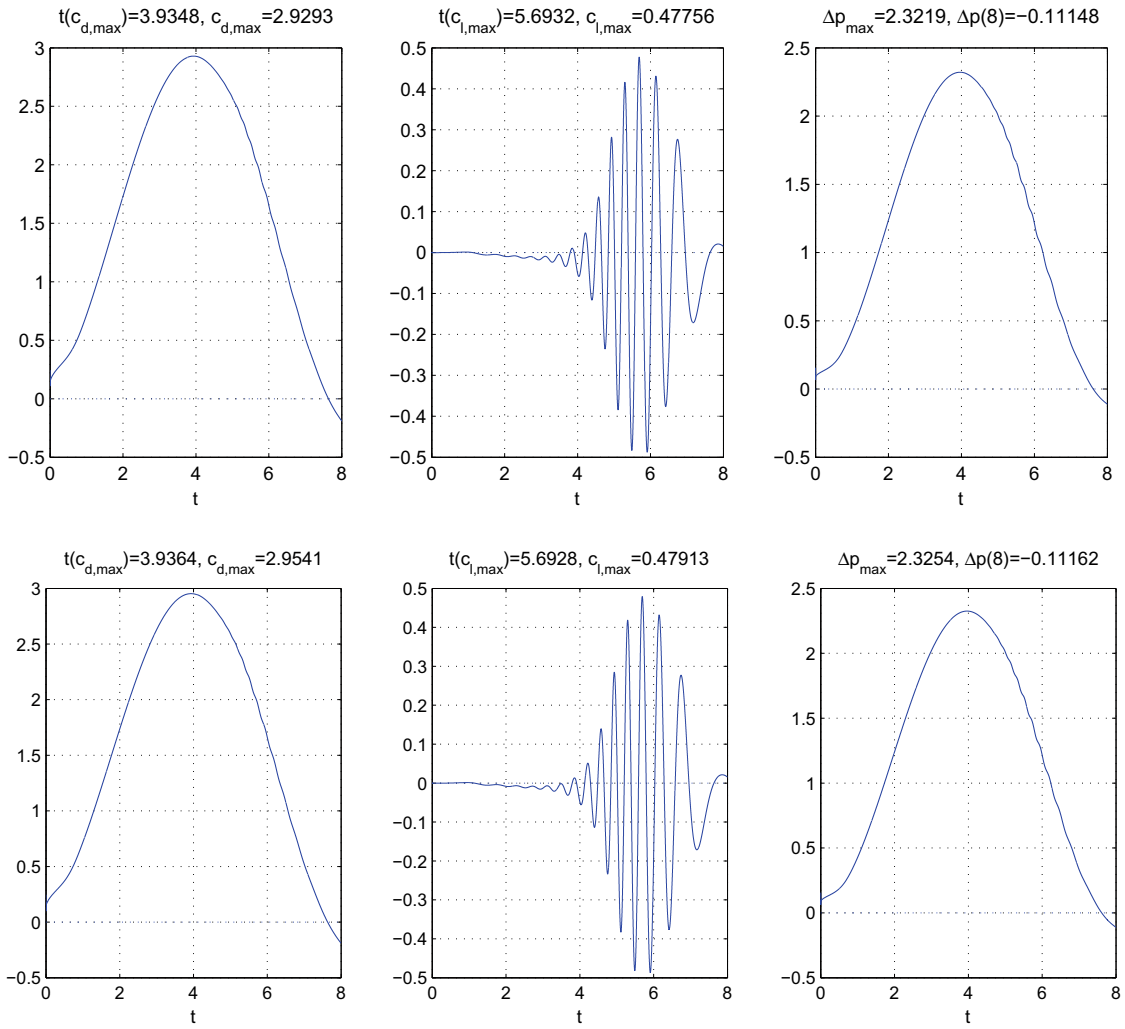

Fig. 12. From left to right: the drag and lift coefficients c_d, c_l and pressure difference between front and back of the cylinder Δp for flow past a cylinder with $v = 1/1000$. Mesh is in Fig. 7. $\Delta t = 0.0004$. 763 isoparametric P4 elements (dof = 6322) for each variable. Top: (3,2) SC scheme. Bottom: (3,2) PA scheme.

If we follow [15] and use $[2.95092, 0.47795, -0.1116]$ as reference values for the maxima of c_d, c_l and $\Delta p(8)$, the relative errors of those quantities are

$$[0.73\%, 0.08\%, 0.11\%] \quad \text{and} \quad [0.11\%, 0.25\%, 0.02\%]$$

for the SC and PA schemes, respectively. We have also used P2 elements instead of P4, with the same time step $\Delta t = 0.0004$, but with one global refinement of the mesh in Fig. 7, so that the number of degrees of freedom remains the same. Then the relative errors in the maxima of c_d, c_l and $\Delta p(8)$ change to

$$[3.19\%, 2.10\%, 0.14\%] \quad \text{and} \quad [0.33\%, 1.65\%, 0.40\%]$$

for the SC and PA schemes respectively. We mention that if we increase Δt from 0.0004 to 0.0005, and keep the other parameters the same as in Fig. 11, the solution blows up around $t = 3$ (after about 6000 time steps).

7. Conclusions

The well-posed formula (18) that expresses pressure in terms of current velocity and forcing fields, via the Laplace–Leray commutator, has enabled us to study in a rather simple way the formal accuracy of time-difference schemes for incompressible viscous flow. We used the commutator formula in (9) and the concept of Stokes pressure to explain the accuracy of existing pressure-approximation and pressure-update projection methods, to devise improved approximations for computation of pressure, and to derive new higher-order slip-corrected projection methods. The slip-correction methods are closely related to the original Kim–Moin scheme from [18] which was devised for a staggered finite-difference grid. At the space-continuous level, the Kim–Moin scheme corresponds to a (2,1) SC scheme here with second-order Crank–Nicolson time differencing and first-order extrapolation for slip correction.

Stability. Our numerical tests indicate that, with no more cost than traditional 2nd-order methods, one can achieve third-order accuracy in time for both velocity and pressure, using (3,2) PA or SC schemes that retain stability with large time steps at low Reynolds number. In tests in smooth domains it appears one may even achieve fourth-order accuracy using (4,3) PA or SC schemes for the Stokes equations with a stability restriction on Δt that appears independent of the spatial grid size h. In domains with corners, however, (4,3) and (3,3) schemes appear subject to a diffusive time-step restriction with the spatial discretizations that we tested.

In general it is not clear just how the implicit treatment of viscous terms enhances stability, but the effect naturally diminishes when viscosity becomes sufficiently small. Our tests on benchmarks involve Reynolds numbers in the hundreds, and here we do encounter practical time step restrictions for stability of (3,2) PA and SC schemes. For these tests we find that we need $U_{\max}\Delta t/h_{\min} \approx O(1)$ where $h_{\min}$ is the size of the smallest edge in the mesh and each edge contains $k+1$ grid points for P_k elements. This appears roughly consistent with a CFL constraint based on the explicit treatment of convection terms.

Accuracy. An interesting fact seen in our formal accuracy analysis is that the projected (divergence-free) velocity satisfies a discretized momentum equation that is fully implicit as regards the Stokes pressure (the viscous part of total pressure). This is a consequence of the commutator formula in (9) and holds for many different projection and time-splitting methods. It does not mean that we need to solve a coupled system for velocity and pressure, however.

The formal analysis indicates that, as with any of the known projection methods, weak boundary layers usually remain in higher gradients of velocity and pressure. In the numerical tests for (3,2) PA and SC schemes, there are some indications of degraded accuracy in such quantities, especially near corners in the domain. Improved understanding and handling of corners would be desirable. Future work is also needed to understand better the impact of spatial discretization on stability and error, especially near boundaries.

Finite elements and the inf–sup condition. A potentially important finding in this paper is that for PA and SC schemes we have observed good stability and accuracy in tests and benchmark problems using simple Lagrange finite elements of equal order for velocity and pressure. This suggests that there is far more flexibility in choosing methods for space discretization in schemes of this type than is traditionally possible for incompressible flow problems using finite elements without stabilization techniques (as in [3,4], for example). For three-dimensional problems, for example, where complex mesh geometry may demand a simple approach, a (2,2) or (2,1) PA scheme with piecewise linear elements (P1/P1) could be considered. (Our numerical tests for a smooth test problem have indicated a substantial improvement in temporal accuracy with third-order schemes, however.) Alternatively, for situations that demand high accuracy, high-degree elements might be used without regard to velocity-pressure compatibility.

Many traditional finite-element discretizations of mixed formulations of the Stokes equations with divergence constraint require the spaces for velocity and pressure approximations to satisfy the inf–sup (Ladyzhenskaya–Babuška–Brezzi) condition: there should exist $c > 0$ independent of the discretization parameter h such that

$$\inf_{q_h \in Y_h} \sup_{\boldsymbol{v}_h \in X_h} \frac{\langle \nabla \cdot \boldsymbol{v}_h, q_h \rangle}{\|\nabla \boldsymbol{v}_h\| \|q_h\|} \geqslant c > 0. \tag{102}$$

The role of this condition is to ensure stability and accuracy for the pressure as determined by the mixed formulation. But as is well known, Lagrange finite elements of equal order for velocity and pressure fail to satisfy this condition.

As we emphasized in Section 2, however, the pressure is necessarily determined by formula (18), for strong solutions of the Navier–Stokes boundary value problem (1)–(3). This should mean that whenever one can compute an accurate velocity, one can compute an accurate pressure from the weak form (19) or the Poisson BVP (20) and (21). The inf–sup condition (102) should play no role in this. (See also [20] for more discussion, where stability of a C^1 finite-element scheme for steady-state Stokes equations is proved, by using simply the Lax–Milgram lemma instead of the inf–sup condition.) But while our numerical tests are suggestive, we have no real reason why PA and SC schemes appear stable irrespective of the inf–sup condition, while PU schemes do not. Clearly, much regarding the role of the inf–sup condition with regard to stability and accuracy of projection methods remains to be explained.

Comparisons. The new slip-correction schemes appear to have stability properties similar to corresponding pressure-approximation schemes. In our implementations, however, higher-order PA schemes seem to be somewhat more robust than their SC counterparts in terms of spatial accuracy and accuracy near boundaries and corners. Pressure-update (PU) schemes are somewhat simpler to describe and to code, but appear far less stable with finite-element pairs that violate the inf–sup condition.

Finally, we remark that certainly one can consider schemes that combine the pressure-approximation, slip-correction, and pressure-update strategies as we have discussed. This may involve additional cost per time step, but it allows one to use lower-order extrapolation while retaining accuracy in the overall scheme. Whether such combinations might yield gains in stability or mitigate deficiencies in individual strategies remains to be seen.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. DMS 06-04420 and 09-05723 (RLP) and DMS 10-11738 (JGL), and partially supported by the Center for Nonlinear Analysis (CNA) under National Science Foundation Grant Nos. 0405343 and 0635983. JL thanks L. Chen for the finite-element package iFEM based on which the finite-element calculations in this paper are done.

Appendix A. Time-discrete normal modes in a strip

Here we describe how to determine normal-mode solutions for the time-discrete schemes of Section 3.1 (slip-correction) and Section 3.2 (pressure-approximation), in the strip $-1 \leqslant x \leqslant 1$ as in Section 5.1. From this analysis we will get a stability condition for the time step that is consistent with the numerical observations from Figs. 1 and 2, namely that the (4,3) and (3,3) PA and SC schemes are stable for time steps below a critical value independent of spatial grid size.

With zero forcing and boundary velocity and neglecting nonlinear terms, we look first for solutions of the time-discrete SC scheme (29)–(31) with the form

$$\boldsymbol{u}^n = \kappa^n \boldsymbol{u}(x,y) = \kappa^n e^{i\xi y}(u(x), iv(x))$$

with corresponding notation for $\boldsymbol{u}_*^n$ and q^n. We will find it convenient to define

$$\mathcal{D}_k(z) = \sum_{j\geqslant 0} \alpha_j^k z^j, \quad \mathcal{E}_m(z) = \sum_{j\geqslant 1} \beta_j^m z^j.$$

We fix $v = 1$. The SC scheme requires

$$\frac{1}{\Delta t}\left(\alpha_0^k \boldsymbol{u}_* + \sum_{j\geqslant 1} \alpha_j^k \boldsymbol{u}\right) = \Delta \boldsymbol{u}_* \quad \text{in } \Omega, \quad \boldsymbol{u}_* = \mathcal{E}_m(\kappa^{-1})\nabla q \quad \text{on } \Gamma, \tag{103}$$

$$\boldsymbol{u}_* = \boldsymbol{u} + \nabla q, \quad \nabla \cdot \boldsymbol{u} = 0 \quad \text{in}\Omega, \quad \boldsymbol{n}\cdot\nabla q^{n+1} = 0 \quad \text{on } \Gamma. \tag{104}$$

We write separate equations for $\boldsymbol{u}$ and q as in Section 3.1 by applying the projection $\mathcal{P}$ and using the commutator relation $(\Delta\mathcal{P} - \mathcal{P}\Delta)\boldsymbol{u}_* = \nabla p$, where $p = p_S(\boldsymbol{u})$ is the Stokes pressure. The equations for $\boldsymbol{u}$ and q correspond to (36) and (39) respectively, and take the form

$$(\mathcal{D}_k(\kappa^{-1}) - \Delta)\boldsymbol{u} + \nabla p = 0, \quad \nabla\cdot\boldsymbol{u} = 0 \quad \text{in } \Omega, \tag{105}$$

$$\Delta p = 0 \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\nabla p = \boldsymbol{n}\cdot(\Delta - \nabla\nabla\cdot)\boldsymbol{u} \quad \text{on } \Gamma, \tag{106}$$

$$\boldsymbol{u} + (1 - \mathcal{E}_m(\kappa^{-1}))\nabla q = 0 \quad \text{on } \Gamma, \tag{107}$$

$$\left(\frac{\alpha_0^k}{\Delta t} - \Delta\right) q = p \quad \text{in } \Omega, \quad \boldsymbol{n}\cdot\nabla q = 0 \quad \text{on } \Gamma. \tag{108}$$

With y-dependence proportional to $e^{i\xi y}$ in the strip $-1 \leqslant x \leqslant 1$, and with

$$\tilde{\mu}(z) = \sqrt{\xi^2 + \mathcal{D}_k(z)}, \quad \lambda = \sqrt{\xi^2 + \frac{\alpha_0^k}{\Delta t}}, \tag{109}$$

and $z = \kappa^{-1}$, the equations become

$$(\tilde{\mu}^2 - \partial_x^2)\begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} \partial_x \\ \xi \end{pmatrix} p = 0, \quad \partial_x u - \xi v = 0 \quad (-1 < x < 1), \tag{110}$$

$$(\xi^2 - \partial_x^2)p = 0 \quad (-1 < x < 1), \quad \partial_x p = \xi \partial_x v \quad (x = \pm 1), \tag{111}$$

$$u = 0, \quad v + (1 - \mathcal{E}_m(\kappa^{-1}))\xi q = 0 \quad (x = \pm 1), \tag{112}$$

$$(\lambda^2 - \partial_x^2)q = p \quad (-1 < x < 1), \quad \partial_x q = 0 \quad (x = \pm 1). \tag{113}$$

We can separately study modes for which pressure, for example, has even or odd symmetry. Looking first for the latter, we find that solutions with divergence-free velocity tangent to the boundary take the form

$$p(x) = A \sinh \xi x \tag{114}$$

$$u(x) = B\xi\left(\frac{\cosh \xi x}{\cosh \xi} - \frac{\cosh \tilde{\mu} x}{\cosh \tilde{\mu}}\right), \tag{115}$$

$$v(x) = B\left(\frac{\xi \sinh \xi x}{\cosh \xi} - \frac{\tilde{\mu} \sinh \tilde{\mu} x}{\cosh \tilde{\mu}}\right), \tag{116}$$

$$q(x) = C\left(\frac{\sinh \xi x}{\xi \cosh \xi} - \frac{\sinh \lambda x}{\lambda \cosh \lambda}\right). \tag{117}$$

On the boundaries $x = \pm 1$, these forms satisfy $u = 0$ and $\partial_x q = 0$. The pressure boundary condition reads $\partial_x p = \xi \partial_x v = \partial_x^2 u$, so it will automatically hold once the first component of (110a) is enforced. Imposing (110a) and the (113a) for q, we find that A, B and C are related by the linear equations

$$A + \frac{\tilde{\mu}^2 - \xi^2}{\cosh \xi} B = 0, \tag{118}$$

$$A - \frac{\lambda^2 - \xi^2}{\xi \cosh \xi} C = 0, \tag{119}$$

so that $(\tilde{\mu}^2 - \xi^2)\xi B + (\lambda^2 - \xi^2)C = 0$. Imposing the boundary condition (112b) for v and using this relation for B and C leads to the equation that determines the growth factor κ, which we write in the form

$$F_o(z, \Delta t, \xi) = 0, \quad z = \kappa^{-1}, \tag{120}$$

with F_o defined via

$$\frac{F_o(z, \Delta t, \xi)}{\cosh \tilde{\mu}} = \xi \tanh \xi - \tilde{\mu} \tanh \tilde{\mu} - \mathcal{S}(z)\left(\xi \tanh \xi - \frac{\xi^2}{\lambda} \tanh \lambda\right), \tag{121}$$

$$\mathcal{S}(z) = \frac{\Delta t}{\alpha_0^k} \mathcal{D}_k(z)(1 - \mathcal{E}_m(z)). \tag{122}$$

The factor $\cosh \tilde{\mu}$ is introduced in order to remove the poles arising from $\tanh \tilde{\mu}$. This makes F_o an entire (i.e., globally analytic) function of z.

For the mode with pressure having even symmetry, $p = \cosh \xi x$, similar calculations lead to the dispersion relation

$$F_e(z, \Delta t, \xi) = 0, \tag{123}$$

$$\frac{F_e(z, \Delta t, \xi)}{\tilde{\mu} \sinh \tilde{\mu}} = \xi \coth \xi - \tilde{\mu} \coth \tilde{\mu} - \mathcal{S}(z)\left(\xi \coth \xi - \frac{\xi^2}{\lambda} \coth \lambda\right). \tag{124}$$

Again, F_e is an entire function of z.

It turns out that exactly the same single-mode dispersion relations govern the time-discrete PA scheme.

Numerical study via winding number. To count all unstable odd modes (and similarly for even modes), for given values of Δt and ξ, we want to count all solutions of (120) for which $|z| < 1$. From basic complex function theory, the number of solutions of (120) satisfying $|z| < R$, counting multiplicity, is the winding number around zero of the closed curve $F_o(R\gamma, \Delta t, \xi)$, where γ parametrizes the unit circle:

$$\gamma(\theta) = e^{i\theta}, \quad 0 \leqslant \theta \leqslant 2\pi. \tag{125}$$

If there are no zeros satisfying $|z| = R$, then the winding number is an integer, the total change in the complex argument divided by 2π. It turns out there is always a trivial zero at $z = 1$, since $D_k(1) = 0$ and $\tilde{\mu} = \xi$. It is convenient to remove this zero and scale amplitudes, by defining

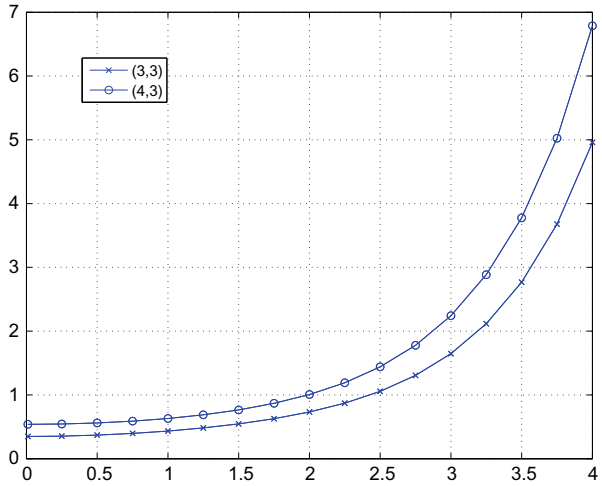

Fig. 13. Largest stable time step Δt_c vs. ξ according to normal-mode theory.

$$\tilde{F}(z) = \frac{F_o(z, \Delta t, \xi)}{1 - z}, \quad f(z) = \frac{\tilde{F}(z)}{(1 + |\tilde{F}(z)|)^{7/8}}.$$

(The parameter $\frac{7}{8}$ is just a convenient number close to 1.) Thus, the quantity

$$N(R) = \frac{1}{2\pi} \int_0^{2\pi} d\Big(\arg f(Re^{i\theta})\Big), \tag{126}$$

the winding number of $f(R\gamma)$ around zero, is the number of zeros of F_o satisfying $|z| < R$, excluding $z = 1$ (once, in case of higher multiplicity).

We compute the winding number $N(R)$ numerically using a simple adaptive stepping algorithm. We evaluate $f_j = f(R \exp i\theta_j)$ for $0 \approx \theta_0 < \theta_1 < \cdots < \theta_n = \theta_0 + 2\pi$, and accumulate $\sum_{j=1}^{n} \arg(f_j/f_{j-1})/2\pi \approx N$. The values θ_j are determined successively by doubling or halving the step $\theta_{j+1} - \theta_j$ to keep $|1 - f_{j+1}/f_j|$ within specified tolerances. This controls the relative change in $\arg f$ and $|f|$, and works well even when there is a zero of f very close to the circle $|z| = R$.

Using bisection in R for $N(R)$, we can then find rather accurately (to a tolerance $\tau = 0.0005$ with no difficulty) a value R for which the winding number satisfies $N(R) = 0 < N(R + \tau)$. Then $1/R$ is approximately the largest magnitude of any amplification factor κ for a mode with odd pressure in the space-continuous scheme. We perform the corresponding calculation for even pressure modes and take the max. The results provide the solid curves in Figs. 1 and 2.

Recall that in Figs. 1 and 2, the results for the (3,3) and (4,3) schemes indicate that for $\xi = 1$ there is a largest stable time step $\Delta t_c > 0$ independent of spatial resolution. The present normal-mode theory suggests that this remains true uniformly for all wave numbers ξ. We compute Δt_c as a function of ξ by fixing $R = 1$ and using bisection in Δt to find the largest Δt where $N = 0$. The results are plotted in Fig. 13, and indicate that Δt_c increases with ξ but remains strictly positive in the limit $\xi \to 0$. (For $\xi = 0.01, \Delta t_c$ is 0.539 for the (4,3) scheme and 0.349 for the (3,3) scheme.) For the (4,3) scheme with $\xi = 1$ the critical time step $\Delta t_c = 0.63$ which compares well with the results in Figs. 1 and 2.

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] T. Barth, P. Bochev, M. Gunzburger, J. Shadid, A taxonomy of consistently stabilized finite element methods for the Stokes problem, SIAM J. Sci. Comput. 25 (2004) 1585–1607.
[4] 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.
[5] J.B. Bell, P. Colella, H. Glaz, A second-order projection method for the incompressible Navier–Stokes equations, J. Comp. Phys. 85 (1989) 257–283.
[6] D.L. Brown, R. Cortez, M.L. Minion, Accurate projection methods for the incompressible Navier–Stokes equations, J. Comp. Phys. 168 (2001) 464–499.
[7] O. Botella, R. Peyret, Benchmark spectral results on the lid-driven cavity flow, Comput. Fluids 27 (1998) 421–433.
[8] L. Chen, C.-S. Zhang, A coarsening algorithm and multilevel preconditioners on adaptive grids by newest vertex bisection, submitted for publication.
[9] W. E, J.-G. Liu, Gauge method for viscous incompressible flows, Commun. Math. Sci. 1 (2003) 317–332.
[10] D. Engwirda (2006), Contours for Triangular Grids, MATLAB Central File Exchange, Retrieved September 15, 2008. (%3chttp://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=10408%3e).
[11] M.S. Gockenbach, Understanding and Implementing the Finite Element Method, SIAM, Philadelphia, 2006.
[12] P.M. Gresho, R.L. Sani, Incompressible Flow and the Finite Element Method, vol. 2, Wiley, 2000.
[13] 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.

[14] W.D. Henshaw, N.A. Petersson, A split-step scheme for the incompressible Navier–Stokes equations, in: Numerical Simulations of Incompressible Flows (Half Moon Bay, CA, 2001), 108–125, World Science Publishing, River Edge, NJ, 2003.
[15] V. John, Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder, Int. J. Numer. Methods Fluids 44 (2004) 777–788.
[16] H. Johnston, J.-G. Liu, Accurate, stable and efficient Navier–Stokes solvers based on explicit treatment of the pressure term, J. Comp. Phys. 199 (2004) 221–259.
[17] 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.
[18] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, J. Comp. Phys. 59 (1985) 308–323.
[19] 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.
[20] J.-G. Liu, J. Liu, R. Pego, Error estimates for finite-element Navier–Stokes solvers without standard inf–sup conditions, Chin. Ann. Math. Ser. B 30 (6) (2009) 743–768.
[21] 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.
[22] S.A. Orszag, M. Israeli, M. Deville, Boundary conditions for incompressible flows, J. Sci. Comput. 1 (1986) 75–111.
[23] P.-O. Persson, G. Strang, A simple mesh generator in MATLAB, SIAM Rev. 46 (2004) 329–345.
[24] J.B. Perot, An analysis of the fractional step method, J. Comp. Phys. 108 (1993) 51–58.
[25] Y. Ren, Y. Jiang, M. Liu, H. Zhang, A class of fully third-order accurate projection methods for solving the incompressible Navier–Stokes equations, Acta Mech. Sin. 21 (2005) 542–549.
[26] 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.
[27] R. Scott, Finite Element Techniques for Curved Boundaries, Thesis, Massachusetts Institute of Technology, Cambridge, 1973.
[28] J.C. Strikwerda, Y.S. Lee, The accuracy of the fractional step method, SIAM J. Numer. Anal. 37 (1999) 37–47.
[29] 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.
[30] J. van Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM J. Sci. Stat. Comput. 7 (1986) 870–891.