Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2016-0234

Vol. **22**, No. 2, pp. 517-541
August 2017

Exponential Time Differencing Gauge Method for Incompressible Viscous Flows

Lili Ju[1] and Zhu Wang[1,*]

[1] *Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA.*

Received 4 December 2016; Accepted (in revised version) 5 February 2017

Abstract. In this paper, we study an exponential time differencing method for solving the gauge system of incompressible viscous flows governed by Stokes or Navier-Stokes equations. The momentum equation is decoupled from the kinematic equation at a discrete level and is then solved by exponential time stepping multistep schemes in our approach. We analyze the stability of the proposed method and rigorously prove that the first order exponential time differencing scheme is unconditionally stable for the Stokes problem. We also present a compact representation of the algorithm for problems on rectangular domains, which makes FFT-based solvers available for the resulting fully discretized system. Various numerical experiments in two and three dimensional spaces are carried out to demonstrate the accuracy and stability of the proposed method.

AMS subject classifications: 65M06, 65M22, 65Y20, 76D05, 76D07

Key words: Incompressible flows, Stokes equations, Navier-Stokes equations, gauge method, exponential time differencing.

1 Introduction

As a fundamental model of incompressible viscous flows, the time-dependent parabolic system of the velocity field $\mathbf{u}(t,\mathbf{x})=(u_1,\cdots,u_d)$ and the pressure $p(t,\mathbf{x})$,

$$\begin{cases} \mathbf{u}_t-\nu\Delta\mathbf{u}+\mathbf{F}(\mathbf{u})+\nabla p=\mathbf{f}, & \text{in } [0,T]\times\Omega, \\ \nabla\cdot\mathbf{u}=0, & \text{in } [0,T]\times\Omega \end{cases} \tag{1.1}$$

has wide applications in engineering and scientific problems. In the mathematical model, $\Omega\in\mathbb{R}^d$ is the domain, $\mathbf{f}=(f_1,\cdots,f_d)$ represents the body force, $\mathbf{F}=(F_1,\cdots,F_d)$ represents

*Corresponding author. *Email addresses:* ju@math.sc.edu (L. Ju), wangzhu@math.sc.edu (Z. Wang)

 ©2017 Global-Science Press

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

the nonlinear convection, and $\nu > 0$ denotes the kinematic viscosity of the fluids. When $\mathbf{F}(\mathbf{u})$ is zero, the system is Stokes; when $\mathbf{F}(\mathbf{u}) = (\mathbf{u}\cdot\nabla)\mathbf{u}$, the system is Navier-Stokes. Many numerical methods have been developed for solving the system (1.1) in order to simulate, predict and/or control the flows (see [10, 11, 14, 15, 25, 33] and references cited therein).

In the development of efficient time integration methods for the fluid system, special attentions have been drawn to deal with the incompressibility constraint. One of the most popular methods is the projection method (or the so-called fractional step methods), which was first developed in the late 1960s by Chorin and Temam independently [4, 32]. The basic idea is to decouple the velocity and pressure in a discrete setting so that one only needs to solve a sequence of elliptic equations. Thus, it would greatly reduce the computational complexity compared to the original fully coupled system. The existing projection methods are usually classified into three categories [11]: the pressure-correction methods, the velocity-correction methods, and the consistent splitting methods. Among them, the popular pressure-correction methods ignore or treat explicitly the pressure term in the first sub-step (i.e., treat viscous effect only) and then correct it in the second sub-step (i.e., treat incompressibility); the velocity-correction methods switch the roles of velocity and pressure terms as those in the pressure-correction method. In this approach, the viscous effect is ignored or treated explicitly in the first sub-step and then corrected in the second one; the consistent splitting methods first compute the velocity by treating the pressure explicitly, then update the pressure by using the weak form of a Poisson equation for the pressure. Although these approaches have been widely used, it is still difficult to develop high-order (in time) schemes for both the velocity and pressure. One of the main reasons is that the boundary condition for the pressure equation in projection methods is artificial, which limits the flexibility and accuracy of the projection methods, especially for the pressure approximation.

Another splitting approach is the gauge method [6–8, 29–31]. The method is based on the Hodge decomposition (or Helmholtz-Hodge decomposition), which states that a sufficient smooth, rapidly decaying vector field $\mathbf{m} = (m_1, \cdots, m_d)$ can be decomposed into the sum of a divergence-free term $\mathbf{u}$ (a solenoidal part) and the gradient of a scalar potential ϕ (an irrotational part), i.e.,

$$\mathbf{m}(t,\mathbf{x}) = \mathbf{u}(t,\mathbf{x}) + \nabla\phi(t,\mathbf{x}), \tag{1.2}$$

where $\nabla\cdot\mathbf{u} = 0$ and these two components are orthogonal. The gauge system is reformulated from (1.1), in which the velocity field $\mathbf{u}$ and the pressure p are replaced by the auxiliary filed $\mathbf{m}$ and the gauge variable ϕ. Based on the Hodge decomposition (1.2) and the boundary conditions of velocity $\mathbf{u}$, certain simple but consistent boundary conditions can be assigned for both $\mathbf{m}$ and ϕ. The resulting system consists of a second-order parabolic problem of $\mathbf{m}$ and a Poisson problem of ϕ that are weakly coupled through the boundary conditions. In order to fully decouple the auxiliary field from the gauge variable during simulations, an explicit extrapolation was used to generate an approximation of the boundary values of $\mathbf{m}$ at a current time step by using its approximations from previous

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

time steps [8]. This approach, plus an explicit treatment of the boundary conditions and an implicit time discretization such as the backward Euler or Crank-Nicolson method, has been proved to be unconditionally stable for the gauge formulation of Stokes system and be stable for the gauge formulation of Navier-Stokes when the CFL condition is satisfied [36]. Since the resulted discrete system is fully decoupled, the computational cost of simulations is reduced to solving a parabolic equation and a Poisson equation separately at each time step.

On the other hand, the exponential integrator based methods, including the exponential time differencing (ETD) [2, 3, 16–22, 34, 37] and the integrating factor approaches [27, 28], have been proven to be very effective for solving parabolic systems due to their good performance in preservation of the exponential behavior of the systems, numerical stability and high-order accuracy. In particular, the ETD method has been well analyzed for solving general semi-linear parabolic stiff equations. Therefore, in this paper, we synthesize the ETD method with the gauge formulation in a *consistent* way, which is explicit in nature with good numerical stability and accuracy, and also allows fast solvers to be developed on rectangular domains.

The remainder of the paper is organized as follows. In Section 2, we first present the gauge formulation of the incompressible viscous flow equation (1.1) and then propose a family of exponential time differencing multistep (ETDMs) schemes for solving the gauge system. The temporal stability of the ETD semi-discretized system is analyzed for the Stokes equation with a short discussion for the Navier-Stokes equation in Section 3. In Section 4, we discuss compact representations of the ETDMs schemes for problems on rectangular domains when the central finite difference is used for spatial discretization, which allows the use of FFT-based solvers for the fully discretized systems. Various numerical experiments are performed to demonstrate the accuracy and stability of the proposed methods in Section 5, and some concluding remarks are drawn in Section 6.

2 Exponential time differencing gauge method

2.1 Gauge formulation

We consider an incompressible viscous flow described by Eq. (1.1) together with a Dirichlet boundary condition $\mathbf{u}|_{\partial\Omega}=\mathbf{g}(t,\mathbf{x})$ and an initial condition $\mathbf{u}(0,\mathbf{x})=\mathbf{u}_0(\mathbf{x})$. The gauge method [8] introduces an auxiliary field $\mathbf{m}(t,\mathbf{x})$ and a gauge variable $\phi(t,\mathbf{x})$ satisfying the Hodge decomposition (1.2) in $\Omega\times[0,T]$. Instead of solving the original system (1.1), it substitutes $\mathbf{u}(t,\mathbf{x})=\mathbf{m}(t,\mathbf{x})-\nabla\phi(t,\mathbf{x})$ into (1.1) and leads to a gauge formulation of the system as

$$\mathbf{m}_t-\nu\Delta\mathbf{m}=\mathbf{f}-\mathbf{F}(\mathbf{u}),\quad \text{in } \Omega\times[0,T] \tag{2.1}$$

and

$$\Delta\phi=\nabla\cdot\mathbf{m},\quad \text{in } \Omega\times[0,T] \tag{2.2}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

with appropriate boundary conditions, where the independent variables t and $\mathbf{x}$ are suppressed. The pressure p is then determined by

$$\nabla p = \nabla(\phi_t - \nu\Delta\phi), \quad \text{in } \Omega\times[0,T]. \tag{2.3}$$

The initial conditions of $\mathbf{m}$ and ϕ are set to be

$$\mathbf{m}(0,\mathbf{x}) = \mathbf{u}_0, \qquad \phi(0,\mathbf{x}) = 0, \tag{2.4}$$

which are compatible with the Hodge decomposition (1.2) at the initial time $t=0$.

The main advantage of the gauge formulation is that it treats the incompressibility constraint and viscous term separately, while giving us the freedom to assign an unambiguous and consistent boundary conditions for $\mathbf{m}$ and ϕ. Two types of consistent boundary conditions are proposed in [6,8]. In this paper, we use the Neumann boundary condition, according to [6],

$$\frac{\partial\phi}{\partial\mathbf{n}} = 0, \quad \mathbf{m}\cdot\mathbf{n} = \mathbf{g}\cdot\mathbf{n}, \quad \mathbf{m}\cdot\boldsymbol{\tau} = \mathbf{g}\cdot\boldsymbol{\tau} + \frac{\partial\phi}{\partial\boldsymbol{\tau}}, \tag{2.5}$$

where $\mathbf{n}$ and $\boldsymbol{\tau}$ are the unit vectors in the normal and tangential directions, respectively. It is seen that $\mathbf{m}$ and ϕ are weakly coupled on the boundary. Obviously, if we can treat $\frac{\partial\phi}{\partial\boldsymbol{\tau}}$ and $\mathbf{F}(\mathbf{u})$ explicitly in (2.5) at each time step, the computational cost of the gauge system (2.1)-(2.2) together with the initial condition (2.4) and the boundary condition (2.5), reduces to the solution of heat equation and Poisson equation as done in [8].

2.2 Exponential time differencing multistep approximations

We now propose an exponential time differencing multistep method for numerical solution to the gauge system, (2.1) and (2.2), along with the boundary condition (2.5) over the time interval $[t_n,t_{n+1}]$. Let the length of time step size $\delta_t = t_{n+1}-t_n$ be uniform for all n. Suppose the domain is spatially partitioned by a mesh $\mathcal{T}_h$ with nodes $\{\mathbf{x}_i\}_{i=1}^n$, and denote by σ_I the set of interior nodes and by σ_B the set of boundary nodes. Denote by $\mathbf{m}_h$, ϕ_h and $\mathbf{u}_h$ the approximations of the auxiliary field $\mathbf{m}$, the gauge variable ϕ and the velocity field $\mathbf{u}$ on $\mathcal{T}_h$, respectively. Let $\vec{m}_{h,k}$ and $\vec{u}_{h,k}$ $(k=1,\cdots,d)$ be the vectors formed by the k-th components of $\mathbf{m}_h$ and $\mathbf{u}_h$ at all nodes of $\mathcal{T}_h$, respectively. Furthermore, we decompose $\vec{m}_{h,k}$ into two parts as $\vec{m}_{h,k} = \vec{m}_{h,k,I}\cup\vec{m}_{h,k,B}$, where $\vec{m}_{h,k,I}$ denotes the components of $\vec{m}_{h,k}$ at all interior nodes σ_I and $\vec{m}_{h,k,B}$ the components at all boundary nodes σ_B. Let $\vec{\mathbf{m}}_h$ be the vector form of $\mathbf{m}_h$. Similar notations will also be used for $\mathbf{f}$ and $\mathbf{F}$. Finally, let $\vec{\phi}_h$ represent the vector containing the values of gauge variable ϕ. In particular, we note that the gauge variable ϕ and $\mathbf{m}$ could be placed on the same grid, or different ones as done in a staggered or half-staggered grid.

After a spatial discretization of (2.1) by using a finite difference or finite volume method, the discretized auxiliary field $\mathbf{m}_h$ satisfies the following system of ordinary differential equations: for $k=1,\cdots,d$, and $t\in[t_n,t_{n+1}]$

$$\frac{d\vec{m}_{h,k,I}(t)}{dt} = \mathbf{L}_D\vec{m}_{h,k,I}(t) + \vec{w}(\vec{m}_{h,k,B}(t)) + \vec{f}_k(t) - \vec{F}_k(\mathbf{u}_h(t)), \tag{2.6}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

where $\mathbf{L}_D$ is the stiffness matrix associated with the scaled Laplace operator $\nu\Delta$ along with the Dirichlet boundary condition, and $\vec{w}(\cdot)$ is a vector-valued function of $\vec{m}_{h,k,B}$, resulted from the boundary condition. It is easy to integrate the system (2.6) from t_n to t_{n+1} and get

$$\vec{m}_{h,k,I}^{n+1} = e^{\delta_t \mathbf{L}_D}\vec{m}_{h,k,I}^{n} + \int_0^{\delta_t} e^{(\delta_t-\tau)\mathbf{L}_D}\Big[\vec{f}_k(t_n+\tau) - \vec{F}_k(\mathbf{u}_h(t_n+\tau)) + \vec{w}(\vec{m}_{h,k,B}(t_n+\tau))\Big]d\tau. \tag{2.7}$$

In order to compute $\{\vec{m}_{h,k,I}^{n+1}\}_{k=1}^{d}$, the last integral in (2.7) has to be approximated in some way because $\mathbf{u}_h(t)$ and $\{\vec{m}_{h,k,B}(t)\}_{k=1}^{d}$ are unknown *a priori* on the interval $(t_n, t_{n+1}]$.

For $k=1,\cdots,d$, we define

$$\vec{R}_k(t) = \vec{f}_k(t) - \vec{F}_k(\mathbf{u}_h(t)) + \vec{w}(\vec{m}_{h,k,B}(t)), \tag{2.8}$$

and employ the idea of multistep method to approximate the definite integration of $e^{(\delta_t-\tau)\mathbf{L}_D}\vec{R}_k(t_n+\tau)$. In spirit of the Adams-Bashforth method, to avoid nonlinear iterative solution process, we approximate $\vec{R}_k(t_n+\tau)$ by the Lagrange interpolating polynomial

$$\vec{P}_k^r(t_n+\tau) = \sum_{s=0}^{r}\eta_{r,s}(\tau)\vec{R}_k(t_{n-s}) \tag{2.9}$$

with

$$\eta_{r,s}(\tau) = \prod_{\substack{l=0\\ l\neq s}}^{r}\frac{\tau+l\delta_t}{(l-s)\delta_t}.$$

It has the property that $\vec{R}_k(t_n+\tau) \approx \vec{P}_k^r(\tau) + \mathcal{O}((\delta_t)^{r+1})$. By substituting $\vec{P}_k^r(t_n+\tau)$ into (2.7), we obtain

$$\vec{m}_{h,k,I}^{n+1} \approx e^{\delta_t \mathbf{L}_D}\vec{m}_{h,k,I}^{n} + \int_0^{\delta_t} e^{(\delta_t-\tau)\mathbf{L}_D}\vec{P}_k^r(t_n+\tau)\,d\tau. \tag{2.10}$$

The most popular first and second order polynomials are listed as follows:

$$r=0: \qquad \vec{P}_k^0(t_n+\tau) = \vec{R}_k(t_n),$$
$$r=1: \qquad \vec{P}_k^1(t_n+\tau) = \left(1+\frac{\tau}{\delta_t}\right)\vec{R}_k(t_n) - \frac{\tau}{\delta_t}\vec{R}_k(t_{n-1}).$$

Correspondingly, the first order ETD scheme (ETD1) and the second order ETD multistep scheme (ETDMs2) for $\vec{m}_{h,k,I}^{n+1}$ are the following:

$$\text{ETD1:} \qquad \vec{m}_{h,k,I}^{n+1} \approx e^{\delta_t \mathbf{L}_D}\vec{m}_{h,k,I}^{n} + \int_0^{\delta_t} e^{(\delta_t-\tau)\mathbf{L}_D}\vec{R}_k(t_n)\,d\tau \tag{2.11}$$

$$\text{ETDMs2:} \qquad \vec{m}_{h,k,I}^{n+1} \approx e^{\delta_t \mathbf{L}_D}\vec{m}_{h,k,I}^{n} + \int_0^{\delta_t} e^{(\delta_t-\tau)\mathbf{L}_D}\left((1+\frac{\tau}{\delta_t})\vec{R}_k(t_n) - \frac{\tau}{\delta_t}\vec{R}_k(t_{n-1})\right)d\tau. \tag{2.12}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Higher order schemes $(r \geq 2)$ can be derived in a similar manner.

Due to the boundary conditions (2.5), the boundary value of $\mathbf{m}$ at t_{n+1}, $\{\vec{m}_{h,k,B}^{n+1}\}_{k=1}^{d}$, is related to the unknown boundary information of ϕ at t_{n+1}. We first *temporarily* approximate it by

$$\vec{\mathbf{m}}_h^{n+1} = \mathbf{g}(t_{n+1}) + \mathbf{G}\vec{\phi}_h^n \tag{2.13}$$

at every boundary node $\mathbf{x}_i \in \sigma_B$ in the ETD1 scheme, and by

$$\vec{\mathbf{m}}_h^{n+1} = \mathbf{g}(t_{n+1}) + \mathbf{G}(2\vec{\phi}_h^n - \vec{\phi}_h^{n-1}) \tag{2.14}$$

in the ETDMs2 scheme, where $\mathbf{G}$ denotes a discrete version of the gradient operator.

The discretized gauge variable ϕ_h^{n+1} is next solved from the Poisson equation (2.2) with the associated zero Neumann boundary condition in (2.5) as

$$\mathbf{L}_N\vec{\phi}_h^{n+1} = \mathbf{D}\vec{\mathbf{m}}_h^{n+1}, \tag{2.15}$$

where $\mathbf{L}_N$ is the stiffness matrix associated with the discrete Laplacian operator Δ under zero Neumann boundary condition, and $\mathbf{D}$ denotes an appropriate discrete matrix for the divergence operator.

We then *correct* the approximate boundary values of the auxiliary field $\{\vec{m}_{h,k,B}^{n+1}\}_{k=1}^{d}$ by using the gradient information of ϕ_h at t_{n+1} as follows:

$$\vec{\mathbf{m}}_h^{n+1} = \mathbf{g}(t_{n+1}) + \mathbf{G}\vec{\phi}_h^{n+1} \tag{2.16}$$

at every boundary node $\mathbf{x}_i \in \sigma_B$. Since ϕ_h^{n+1} satisfies the zero Neumann conditions, only its tangential derivative needs to be evaluated at the boundary nodes σ_B.

Finally, if $\mathbf{F}(\mathbf{u})$ appears in (2.1) (such as in the Navier-Stokes equation), the approximate velocity filed $\mathbf{u}_h^{n+1}$ needs to be recovered from the Hodge decomposition (1.2) as

$$\vec{\mathbf{u}}_h^{n+1} = \vec{\mathbf{m}}_h^{n+1} - \mathbf{G}\vec{\phi}_h^{n+1} \tag{2.17}$$

on every interior node $\mathbf{x}_i \in \sigma_I$. The boundary value of $\mathbf{u}_h^{n+1}$ is explicitly given by $\mathbf{g}^{n+1}$ as stated by the boundary condition of the problem.

Remark 2.1. The integrals in (2.11) and (2.12) generally are computed by first performing exact integrations, and then evaluating some multiplications of matrix-exponentials with vectors. The later operations can be effectively implemented by the Krylov subspace method when the problem scale is large, see [9, 23] for details.

Remark 2.2. In order to enforce the consistency of the coupling of the auxiliary field $\mathbf{m}^{n+1}$ and the gauge variable ϕ^{n+1} in the discrete level, it could be theoretically better to repeat the process of (2.15) and (2.16) according the accuracy order of the scheme. However, from our numerical experiments in Section 5, we found that performing twice the process of (2.15) and (2.16) only yields a very slight improvement for both ETD1 and ETDMs2 schemes. Thus, we don't recommend such a process.

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

3 Temporal stability analysis

In this section, we focus on temporal stability analysis of the gauge system by using the ETD schemes for time integration. In particular, we mainly consider the gauge formulation of the Stokes equation (which is linear) without body force, i.e., the system consisting of (2.1) with vanished $\mathbf{f}$ and $\mathbf{F}(\mathbf{u})$, (2.2) and (2.5). We assume that $\mathbf{u}$ is zero on the boundary $\partial\Omega$.

When the time is restricted on the interval $[t_n,t_{n+1}]$, the semi-discretized gauge system corresponding to the ETD1 scheme reads as follows:

$$\begin{cases} \mathbf{m}_t-\nu\Delta\mathbf{m}=0, & \text{in } \Omega\times[t_n,t_{n+1}),\\ \mathbf{m}=\nabla\phi^n, & \text{on } \partial\Omega\times[t_n,t_{n+1}),\\ \mathbf{m}(t_n)=\mathbf{u}^n+\nabla\phi^n, & \text{in } \Omega \end{cases} \tag{3.1}$$

and

$$\begin{cases} \Delta\phi=\nabla\cdot\mathbf{m}, & \text{in } \Omega\times[t_n,t_{n+1}),\\ \frac{\partial\phi}{\partial\mathbf{n}}=0, & \text{on } \partial\Omega. \end{cases} \tag{3.2}$$

Without considering the spatial discretization, we analyze the stability of the ETD1 for time stepping. Let us first introduce a new variable $\widehat{\mathbf{u}}=\mathbf{m}-\nabla\phi^n$ in $[t_n,t_{n+1})$. Then it can be derived from (3.1) and (3.2) that

$$\begin{cases} \widehat{\mathbf{u}}_t-\nu\Delta\widehat{\mathbf{u}}=\nu\nabla(\Delta\phi^n), & \text{in } \Omega\times[t_n,t_{n+1}),\\ \widehat{\mathbf{u}}=0, & \text{on } \partial\Omega\times[t_n,t_{n+1}),\\ \widehat{\mathbf{u}}(t_n)=\mathbf{u}^n, & \text{in } \Omega \end{cases} \tag{3.3}$$

and the continuity of $\mathbf{m}$ at the time t_{n+1} indicates

$$\begin{aligned} \nabla\cdot\widehat{\mathbf{u}}^{n+1}&=\nabla\cdot\mathbf{m}^{n+1}-\Delta\phi^n\\ &=\nabla\cdot(\mathbf{u}^{n+1}+\nabla\phi^{n+1})-\Delta\phi^n\\ &=\Delta\phi^{n+1}-\Delta\phi^n \end{aligned} \tag{3.4}$$

due to the incompressibility condition $\nabla\cdot\mathbf{u}^{n+1}=0$.

Define the scaled (by ν) Dirichlet Laplace operator

$$L=\nu\Delta\colon H^2(\Omega)\cap H_0^1(\Omega)\to L^2(\Omega).$$

Denote the inner product $(\cdot,\cdot)$ in the L^2 sense as $(\mathbf{u},\mathbf{v})=\int_\Omega\mathbf{u}\cdot\mathbf{v}d\mathbf{x}$. Clearly L is self-adjoint, invertible and all its eigenvalues are negative. It is well known that L is the infinitesimal generator of a strongly continuous semigroup on $L^2(\Omega)$, which is denoted by e^{tL} with $t\geq 0$. Note all eigenvalues of e^{tL} are between 0 and 1.

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

The analytic solution of (3.3) is

$$\begin{aligned}\widehat{\mathbf{u}}^{n+1} &= e^{\delta_t L}\widehat{\mathbf{u}}^n + \nu\int_0^{\delta_t} e^{(\delta_t-\tau)L}\nabla(\Delta\phi^n)\,d\tau \\ &= e^{\delta_t L}\widehat{\mathbf{u}}^n - \nu L^{-1}(I-e^{\delta_t L})\nabla(\Delta\phi^n).\end{aligned} \tag{3.5}$$

Note that $e^{\delta_t L}$ and $I-e^{\delta_t L}$ are all self-adjoint and commutative with L.

Rearranging the above equation, we get

$$-(I-e^{\delta_t L})^{-1}L(\widehat{\mathbf{u}}^{n+1}-e^{\delta_t L}\widehat{\mathbf{u}}^n) = \nu\nabla(\Delta\phi^n). \tag{3.6}$$

Taking the inner product of both sides of (3.6) with $\widehat{\mathbf{u}}^{n+1}$ yields

$$\begin{aligned}&\left(-(I-e^{\delta_t L})^{-1}L(I-e^{\delta_t L})\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}\right) \\ &\quad+\left(-(I-e^{\delta_t L})^{-1}Le^{\delta_t L}(\widehat{\mathbf{u}}^{n+1}-\widehat{\mathbf{u}}^n),\widehat{\mathbf{u}}^{n+1}\right) = \nu\left(\nabla(\Delta\phi^n),\widehat{\mathbf{u}}^{n+1}\right).\end{aligned} \tag{3.7}$$

For the first term on the left-hand side of (3.7), we have

$$\left(-(I-e^{\delta_t L})^{-1}L(I-e^{\delta_t L})\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}\right) = -(L\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}). \tag{3.8}$$

Define a new operator $Q=-(I-e^{\delta_t L})^{-1}Le^{\delta_t L}$, and it easy to see that Q is a self-adjoint, positive definite operator. Then for the second term we get

$$\begin{aligned}&\left(Q(\widehat{\mathbf{u}}^{n+1}-\widehat{\mathbf{u}}^n),\widehat{\mathbf{u}}^{n+1}\right) \\ &=\left(Q\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}\right)-\left(Q\widehat{\mathbf{u}}^n,\widehat{\mathbf{u}}^{n+1}\right) \\ &=\frac{1}{2}\left(Q\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}\right)-\frac{1}{2}(Q\widehat{\mathbf{u}}^n,\widehat{\mathbf{u}}^n)+\frac{1}{2}\left(Q(\widehat{\mathbf{u}}^{n+1}-\widehat{\mathbf{u}}^n),\widehat{\mathbf{u}}^{n+1}-\widehat{\mathbf{u}}^n\right).\end{aligned} \tag{3.9}$$

For the term on the right-hand side of (3.7), we have

$$\begin{aligned}&\left(\nabla(\Delta\phi^n),\widehat{\mathbf{u}}^{n+1}\right) \\ &=-\int_\Omega \Delta\phi^n(\nabla\cdot\widehat{\mathbf{u}}^{n+1})\,d\mathbf{x} \qquad (\text{because } \widehat{\mathbf{u}}^{n+1}|_{\partial\Omega}=0) \\ &=-\int_\Omega \Delta\phi^n(\Delta\phi^{n+1}-\Delta\phi^n)\,d\mathbf{x} \qquad (\text{due to (3.4)}) \\ &=\frac{1}{2}\int_\Omega(\Delta\phi^{n+1}-\Delta\phi^n)^2\,d\mathbf{x}-\frac{1}{2}\int_\Omega(\Delta\phi^{n+1})^2\,d\mathbf{x}+\frac{1}{2}\int_\Omega(\Delta\phi^n)^2\,d\mathbf{x} \\ &=\frac{1}{2}\int_\Omega(\nabla\cdot\widehat{\mathbf{u}}^{n+1})^2\,d\mathbf{x}-\frac{1}{2}\|\Delta\phi^{n+1}\|^2+\frac{1}{2}\|\Delta\phi^n\|^2 \\ &=\frac{1}{2}\|\nabla\cdot\widehat{\mathbf{u}}^{n+1}\|^2-\frac{1}{2}\|\Delta\phi^{n+1}\|^2+\frac{1}{2}\|\Delta\phi^n\|^2.\end{aligned} \tag{3.10}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Since $\widehat{\mathbf{u}}|_{\partial\Omega}=0$, it holds that

$$\frac{\nu}{2}\|\nabla\cdot\widehat{\mathbf{u}}^{n+1}\|^2\leq\frac{\nu}{2}\|\nabla\widehat{\mathbf{u}}^{n+1}\|^2=-\frac{1}{2}(L\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}). \tag{3.11}$$

Substituting (3.8) - (3.11) into (3.7), we obtain

$$\begin{aligned}\left(Q\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}\right)-(Q\widehat{\mathbf{u}}^{n},\widehat{\mathbf{u}}^{n})+\left(Q(\widehat{\mathbf{u}}^{n+1}-\widehat{\mathbf{u}}^{n}),\widehat{\mathbf{u}}^{n+1}-\widehat{\mathbf{u}}^{n}\right)\\ -(L\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1})+\nu\|\Delta\phi^{n+1}\|^2-\nu\|\Delta\phi^{n}\|^2\leq 0.\end{aligned} \tag{3.12}$$

Because $\mathbf{m}^{n+1}$ is continuous at t^{n+1}, we have

$$\begin{aligned}0&=(\widehat{\mathbf{u}}^{n+1}-\nabla\phi^{n})-(\mathbf{u}^{n+1}-\nabla\phi^{n+1})\\ &=(\mathbf{u}^{n+1}-\widehat{\mathbf{u}}^{n+1})+(\nabla\phi^{n+1}-\nabla\phi^{n}).\end{aligned} \tag{3.13}$$

Also note that $Q\mathbf{u}^{n+1}$ and $\nabla\phi^{n+1}-\nabla\phi^{n}$ are orthogonal since

$$\begin{aligned}&\left(Q\mathbf{u}^{n+1},\nabla\phi^{n+1}-\nabla\phi^{n}\right)\\ &=\left(\mathbf{u}^{n+1},\nabla Q(\phi^{n+1}-\phi^{n})\right)\\ &=-\int_\Omega Q(\phi^{n+1}-\phi^{n})(\nabla\cdot\mathbf{u}^{n+1})d\mathbf{x}+\int_{\partial\Omega}(\mathbf{u}^{n+1}\cdot\mathbf{n})Q(\phi^{n+1}-\phi^{n})ds\\ &=0\qquad\text{(because }\mathbf{u}^{n+1}\text{ is divergence-free and }\mathbf{u}^{n+1}|_{\partial\Omega}=0).\end{aligned} \tag{3.14}$$

Taking the inner product of both sides of the equation (3.13) with $Q\mathbf{u}^{n+1}$ and using the orthogonality (3.14), we have

$$\begin{aligned}0&=\left(Q\mathbf{u}^{n+1},\mathbf{u}^{n+1}-\widehat{\mathbf{u}}^{n+1}\right)\\ &=\frac{1}{2}\left(Q\mathbf{u}^{n+1},\mathbf{u}^{n+1}\right)-\frac{1}{2}\left(Q\widehat{\mathbf{u}}^{n+1},\widehat{\mathbf{u}}^{n+1}\right)\\ &\quad+\frac{1}{2}\left(Q(\mathbf{u}^{n+1}-\widehat{\mathbf{u}}^{n+1}),\mathbf{u}^{n+1}-\widehat{\mathbf{u}}^{n+1}\right).\end{aligned} \tag{3.15}$$

Based on (3.12) and (3.15), the positivity of operators Q and $-L$, we have

$$\left(Q\mathbf{u}^{n+1},\mathbf{u}^{n+1}\right)-(Q\widehat{\mathbf{u}}^{n},\widehat{\mathbf{u}}^{n})+\nu\|\Delta\phi^{n+1}\|^2-\nu\|\Delta\phi^{n}\|^2\leq 0. \tag{3.16}$$

Because $\widehat{\mathbf{u}}^{n}=\mathbf{u}^{n}$, we have

$$\left(Q\mathbf{u}^{n+1},\mathbf{u}^{n+1}\right)+\nu\|\Delta\phi^{n+1}\|^2\leq(Q\mathbf{u}^{n},\mathbf{u}^{n})+\nu\|\Delta\phi^{n}\|^2. \tag{3.17}$$

Thus we obtain the following result.

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Theorem 3.1. *The ETD1 time stepping scheme for the gauge system of the Stokes problem with zero Dirichlet boundary conditions is unconditionally stable.*

The theoretical analysis of ETDMs2 or even higher-order ETD gauge algorithms would involve analyzing much more complicated operators, which needs further deeper study.

Remark 3.1. For the Navier-Stokes equation, or more general nonlinear flows where $\mathbf{F}(\mathbf{u}) \neq 0$, the stability of the ETD method could be restricted by the CFL condition resulted from the nonlinear term $\mathbf{F}(\mathbf{u})$ as discussed in [36]; in particular, in the case of very small viscosity ν, it is usually required that $\delta_t \leq h/|\mathbf{u}(\mathbf{x})|$ where h denotes the spatial grid size.

4 Fast solver on rectangular domains

In this section, we consider problems in a rectangular domain and develop a compact formulation for the ETD multistep method and the corresponding FFT-based fast implementation. The essentials of the compact representation include the splitting of spatial dimensions, diagonalization of the Laplacian, and analytic evaluations of exponential parts in both the linear term and the approximation of nonlinear and boundary terms. We will present the algorithm in the context of a two-dimensional domain, while its extension to the three dimensional space is straightforward.

Assume the domain $\Omega=[x_b,x_e]\times[y_b,y_e]$. Although various numerical methods can be used for the spatial discretization of the incompressible viscous flow model (1.1), we here choose the well-known second order accurate central finite difference approximations due to its simplicity. Suppose the domain is partitioned into $N_x \times N_y$ equally spaced subdomains. The corresponding grid sizes in the horizontal and vertical directions are $h_x=(x_e-x_b)/N_x$ and $h_y=(y_e-y_b)/N_y$, and coordinates of the grid points are (x_i,y_j), where $x_i=x_b+ih_x$, $y_j=y_b+jh_y$, $i=0,\cdots,N_x$ and $j=0,\cdots,N_y$. In Eq. (2.15), the divergence information of $\mathbf{m}$ is needed for computing the gauge variable ϕ. If $\mathbf{m}$ and ϕ are defined at the same grid, to evaluate $\nabla\cdot\mathbf{m}$ at a boundary node, one has to use one-sided finite differences. It would result in larger approximation errors on the boundary than the interior region. To this end, we employ a half-staggered grid, which allows us to use the central finite difference method for approximating all the differentiations. A schematic of the grid is shown in Fig. 1.

4.1 Spatial discretizations of differential operators

We first introduce some notations related to the spatial discretization of differential operators. Let $\mathbf{m}_{h,k,i,j}=\mathbf{m}_{h,k}(x_i,y_j)$, the Laplacian $\mathbf{L}_D\vec{\mathbf{m}}_h$ at each interior node $(x_i,y_j)\in\sigma_I$ is defined as

$$\begin{aligned}(\mathbf{L}_D\vec{m}_{h,k,I})_{i,j}=&h_x^{-2}(m_{h,k,i+1,j}-2m_{h,k,i,j}+m_{h,k,i-1,j})\\&+h_y^{-2}(m_{h,k,i,j+1}-2m_{h,k,i,j}+m_{h,k,i,j-1})\end{aligned}\tag{4.1}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

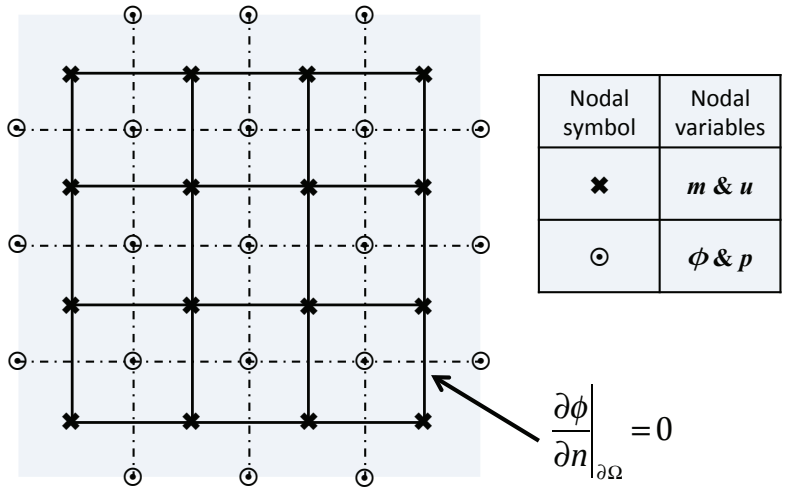


Figure 1: A schematic of the half-staggered grid ($N_x=N_y=3$) for the spatial discretization.

for $k=1,2$. The divergence of $\mathbf{m}$ needs to be computed at the centers of the grid, $(x_{i+1/2},y_{j+1/2})$, where

$$\begin{aligned}(\mathbf{D}\vec{\mathbf{m}}_h)_{i+\frac{1}{2},j+\frac{1}{2}}=&\frac{1}{2}h_x^{-1}(m_{h,1,i+1,j+1}-m_{h,1,i,j+1}+m_{h,1,i+1,j}-m_{h,1,i,j})\\&+\frac{1}{2}h_y^{-1}(m_{h,2,i,j+1}-m_{h,2,i,j}+m_{h,2,i+1,j+1}-m_{h,2,i+1,j}).\end{aligned}\tag{4.2}$$

Given $\phi_{h,i,j}=\phi_h(x_{i+1/2},y_{j+1/2})$, the Laplacian $\mathbf{L}_N\vec{\phi}_h$ at an interior node $(x_i,y_j)\in\sigma_I$ is defined as

$$(\mathbf{L}_N\vec{\phi}_h)_{i,j}=h_x^{-2}(\phi_{h,i+1,j}-2\phi_{h,i,j}+\phi_{h,i-1,j})+h_y^{-2}(\phi_{h,i,j+1}-2\phi_{h,i,j}+\phi_{h,i,j-1}).\tag{4.3}$$

As for boundary nodes, we shall use the zero Neumann boundary condition of ϕ. For instance, when $i=0$ and $0<j<N_y-1$,

$$(\mathbf{L}_N\vec{\phi}_h)_{i,j}=2h_x^{-2}(\phi_{h,i+1,j}-\phi_{h,i,j})+h_y^{-2}(\phi_{h,i,j+1}-2\phi_{h,i,j}+\phi_{h,i,j-1}).$$

When $i=j=0$, we have

$$(\mathbf{L}_N\vec{\phi}_h)_{i,j}=2h_x^{-2}(\phi_{h,i+1,j}-\phi_{h,i,j})+2h_y^{-2}(\phi_{h,i,j+1}-\phi_{h,i,j}).$$

The gradient of ϕ_h, $\mathbf{G}\vec{\phi}_h$, is defined at all the nodes of $\mathcal{T}_h$, which has two components:

$$\begin{cases}\left(\frac{\partial\vec{\phi}_h}{\partial x}\right)_{i,j}=\frac{1}{2}h_x^{-1}(\phi_{h,i,j-1}-\phi_{h,i-1,j-1}+\phi_{h,i,j}-\phi_{h,i-1,j}),\\ \left(\frac{\partial\vec{\phi}_h}{\partial y}\right)_{i,j}=\frac{1}{2}h_y^{-1}(\phi_{h,i,j}-\phi_{h,i,j-1}+\phi_{h,i-1,j}-\phi_{h,i-1,j-1}).\end{cases}\tag{4.4}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Let us write the unknowns $\mathbf{m}_h$ and ϕ_h in their original array form according to the space dimension, denoted by $\mathbf{M}_{h,k,I}=(\mathbf{m}_{h,k,I,i,j})$ and $\mathbf{\Phi}_h=(\phi_{i,j})$. The second order spatial differential operators $\frac{\partial^2}{\partial x^2}$ and $\frac{\partial^2}{\partial y^2}$ also can be written in the matrix form. We define $\mathbf{A}=h_x^{-2}\mathbb{G}_{N_x}$ and $\mathbf{B}=h_y^{-2}\mathbb{G}_{N_y}$ where

$$\mathbb{G}_N=\begin{pmatrix} -2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ & & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 & -2 & 1 \\ 0 & \cdots & 0 & 0 & 1 & -2 \end{pmatrix}_{(N-1)\times(N-1)}.$$

Two special operators $\textcircled{x}$ and $\textcircled{y}$ are defined as in [37] such that

$$(\mathbf{A}\textcircled{x}\mathbf{M}_{h,k,I})_{i,j}=\sum_{l=1}^{N_x-1}(\mathbf{A})_{i,l}(\mathbf{M}_{h,k,I})_{l,j},(\mathbf{B}\textcircled{y}\mathbf{M}_{h,k,I})_{i,j}=\sum_{l=1}^{N_y-1}(\mathbf{B})_{j,l}(\mathbf{M}_{h,k,I})_{i,l}$$

and these two operators are commutative. After enforcing the Dirichlet boundary condition, the discrete 2D Laplacian operator in (2.6) satisfies

$$\mathbf{L}_D\mathbf{M}_{h,k,I}=\nu\left[(\mathbf{A}\textcircled{x}+\mathbf{B}\textcircled{y})\mathbf{M}_{h,k,I}+\mathbf{M}^x_{h,k,B}+\mathbf{M}^y_{h,k,B}\right] \tag{4.5}$$

for $k=1,2$, where

$$\mathbf{M}^x_{h,k,B}=h_x^{-2}\begin{pmatrix} m_k(t,x_0,y_1) & m_k(t,x_0,y_2) & \cdots & m_k(t,x_0,y_{N_y-1}) \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \\ m_k(t,x_{N_x},y_1) & m_k(t,x_{N_x},y_2) & \cdots & m_k(t,x_{N_x},y_{N_y-1}) \end{pmatrix} \tag{4.6}$$

and

$$\mathbf{M}^y_{h,k,B}=h_y^{-2}\begin{pmatrix} m_k(t,x_1,y_0) & 0 & \cdots & 0 & m_k(t,x_1,y_{N_y}) \\ m_k(t,x_2,y_0) & 0 & \cdots & 0 & m_k(t,x_2,y_{N_y}) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ m_k(t,x_{N_x-1},y_0) & 0 & \cdots & 0 & m_k(t,x_{N_x-1},y_{N_y}) \end{pmatrix}. \tag{4.7}$$

As for the Laplacian with zero Neumann boundary condition in (2.15), $\frac{\partial^2}{\partial x^2}$ and $\frac{\partial^2}{\partial y^2}$ are discretized as the following: $\widehat{\mathbf{A}}=h_x^{-2}\widehat{\mathbb{G}}_{N_x}$ and $\widehat{\mathbf{B}}=h_y^{-2}\widehat{\mathbb{G}}_{N_y}$, where

$$\widehat{\mathbb{G}}_N=\begin{pmatrix} -1 & 1 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ & & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 & -2 & 1 \\ 0 & \cdots & 0 & 0 & 1 & -1 \end{pmatrix}_{(N+1)\times(N+1)}.$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Then

$$\mathbf{L}_N\mathbf{\Phi}_h=\left(\widehat{\mathbf{A}}\textcircled{x}+\widehat{\mathbf{B}}\textcircled{y}\right)\mathbf{\Phi}_h. \tag{4.8}$$

4.2 Main components of the ETDMs method

The ETDMs method has three major components: I) to evolve the heat equation (2.1) of the auxiliary variable by (2.11) in ETD1 or (2.12) in ETDMs2; II) to solve the Poisson equation (2.15) with zero Neumann boundary condition for the gauge variables; and III) to update the boundary values of the auxiliary variable. Next, we elaborate these components. In particular, based on the eigen-decomposition of the discrete Laplace operator in x and y directions separately, we develop a compact representation of the algorithm, which provides a fast solver to the gauge system.

Component I With the spatial discretization and the dimension splitting (4.5) discussed in Section 4.1, the equation (2.6) can be rewritten to be the following system: for $k=1,2$,

$$\frac{d\mathbf{M}_{h,k,I}}{dt}=\nu(\mathbf{A}\textcircled{x}+\mathbf{B}\textcircled{y})\mathbf{M}_{h,k,I}+\mathcal{F}_k(t,\mathbf{U}_h)+\mathcal{W}(\mathbf{M}_{h,k,B}), \tag{4.9}$$

where $\mathcal{F}_k(t,\mathbf{U}_h)=(\mathbf{f}_k(t)-F_k(\mathbf{U}_h))_{(N_x-1)\times(N_y-1)}$ comes from the forcing source and nonlinear terms, and $\mathcal{W}(\mathbf{M}_{h,k,B})=\nu(\mathbf{M}^x_{h,k,B}+\mathbf{M}^y_{h,k,B})_{(N_x-1)\times(N_y-1)}$ from the enforcement of the boundary condition. Note that $\mathbf{A}$ and $\mathbf{B}$ are diagonalizable (see [26]) and have the following eigenvalue decompositions:

$$\mathbf{A}=\mathbf{P}_x\widetilde{\mathbf{D}}_x{\mathbf{P}_x}^{-1},\quad \mathbf{B}=\mathbf{P}_y\widetilde{\mathbf{D}}_y{\mathbf{P}_y}^{-1}, \tag{4.10}$$

where $\widetilde{\mathbf{D}}_x$ and $\widetilde{\mathbf{D}}_y$ are diagonal matrices whose diagonal entries are the eigenvalues of $\mathbf{A}$ and $\mathbf{B}$ respectively, and $\mathbf{P}_x$ and $\mathbf{P}_y$ are the corresponding eigenvector matrices. In particular, we have

$$\begin{aligned}
d^x_i&=-\frac{4D}{h_x^2}\sin^2\left(\frac{\pi i}{2N_x}\right),\quad (\mathbf{P}_x)_{i,j}=\sin\left(\frac{\pi ij}{N_x}\right),\quad i,j=1,\cdots,N_x-1,\\
d^y_i&=-\frac{4D}{h_y^2}\sin^2\left(\frac{\pi i}{2N_y}\right),\quad (\mathbf{P}_y)_{i,j}=\sin\left(\frac{\pi ij}{N_y}\right),\quad i,j=1,\cdots,N_y-1.
\end{aligned}$$

Applying $\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}$ on both sides of Eq. (4.9) and denoting $\mathbf{V}_{h,k,I}=\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathbf{M}_{h,k,I}$, we have

$$\frac{d\mathbf{V}_{h,k,I}}{dt}=\mathbf{H}\odot\mathbf{V}_{h,k,I}+\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathcal{F}_k(t,\mathbf{U}_h)+\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathcal{W}(\mathbf{M}_{h,k,B}), \tag{4.11}$$

where $\mathbf{H}=(h_{i,j})_{(N_x-1)\times(N_y-1)}$ with the element $h_{i,j}=\nu(d^x_i+d^y_j)$ and the operation "$\odot$" represents element by element multiplication between two arrays of the same size. It

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

is easy to show that $h_{i,j}<0$. We also remark that the operations $\mathbf{P}_x\textcircled{x}$ and $\mathbf{P}_y\textcircled{y}$ can be implemented efficiently by fast Fourier transform (FFT), and $\mathbf{P}_x^{-1}\textcircled{x}$ and $\mathbf{P}_y^{-1}\textcircled{y}$ by inverse FFT, see [21].

Define another operation "(e^*)" to be the element by element exponentials of an array. Given the information at the beginning of the time interval $[t_n,t_{n+1}]$, $\mathbf{V}_{h,k,I}^n=\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathbf{M}_{h,k,I}^n$, Eq. (4.11) can be solved by multiplying $(e^*)^{-\mathbf{H}t}$ and integrating the system from t_n to t_{n+1}

$$\begin{aligned}\mathbf{V}_{h,k,I}^{n+1}=&e^{\delta_t\mathbf{H}}\odot\mathbf{V}_{h,k,I}^n\\&+\int_0^{\delta_t}(e^*)^{(\delta_t-\tau)\mathbf{H}}\odot(\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathcal{F}_k(t_n+\tau,\mathbf{U}_h(t_n+\tau)))d\tau\\&+\int_0^{\delta_t}(e^*)^{(\delta_t-\tau)\mathbf{H}}\odot(\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathcal{W}(\mathbf{M}_{h,k,B}(t_n+\tau)))d\tau,\end{aligned}\tag{4.12}$$

which yields

$$\begin{aligned}\mathbf{M}_{h,k,I}^{n+1}=&\mathbf{P}_y\textcircled{y}\mathbf{P}_x\textcircled{x}\Big[(e^*)^{\delta_t\mathbf{H}}\odot(\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\mathbf{M}_{h,k,I}^n)+\int_0^{\delta_t}(e^*)^{(\delta_t-\tau)\mathbf{H}}\odot\\&\Big(\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}\big(\mathcal{F}(t_n+\tau,\mathbf{U}_h(t_n+\tau))+\mathcal{W}(\mathbf{M}_{h,k,B}(t_n+\tau))\big)\Big)d\tau\Big].\end{aligned}\tag{4.13}$$

To shorten the presentation, we define

$$\mathbf{Q}_k^R=\int_0^{\delta_t}(e^*)^{(\delta_t-\tau)\mathbf{H}}\odot\mathbf{R}_k(t_n+\tau,\mathbf{U}_h(t_n+\tau),\mathbf{M}_{h,k,B}(t_n+\tau))\,d\tau,$$

where

$$\mathbf{R}_k(t,\mathbf{x},\mathbf{U}_h(t),\mathbf{M}_{h,k,B}(t))=\mathbf{P}_y^{-1}\textcircled{y}\mathbf{P}_x^{-1}\textcircled{x}[\mathcal{F}_k(t,\mathbf{U}_h(t))+\mathcal{W}(\mathbf{M}_{h,k,B}(t))].\tag{4.14}$$

Then Eq. (4.13) can be written as

$$\mathbf{M}_{h,k,I}^{n+1}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\,\Big[(e^*)^{\delta_t\mathbf{H}}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{M}_{h,k,I}^n)+\mathbf{Q}_k^{\mathbf{R}}\Big].\tag{4.15}$$

As presented in Section 2.2, $\mathbf{R}_k(t_n+\tau,\mathbf{U}_h(t_n+\tau),\mathbf{M}_{h,k,B}(t_n+\tau))$ can be approximated by a polynomial $P_k^r(\tau)$ of degree r on the interval $\tau\in[0,\delta_t]$, for example,

$$\begin{aligned}&P_k^0(\tau)=\mathbf{R}_k(t_n,\mathbf{U}_h^n,\mathbf{M}_{h,k,B}^n),\\&P_k^1(\tau)=\Big(1+\frac{\tau}{\delta_t}\Big)\mathbf{R}_k(t_n,\mathbf{U}_h^n,\mathbf{M}_{h,k,B}^n)-\frac{\tau}{\delta_t}\mathbf{R}_k(t_{n-1},\mathbf{U}_h^{n-1},\mathbf{M}_{h,k,B}^{n-1}).\end{aligned}\tag{4.16}$$

Then we have

$$\mathbf{Q}_k^{R,(r)}=\int_0^{\delta_t}(e^*)^{(\delta_t-\tau)\mathbf{H}}\odot P_k^r(\tau)\,d\tau,\tag{4.17}$$

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

which is an approximation of $\mathbf{Q}_k^R$ with a temporal accuracy of order $r+1$. Next, $\mathbf{M}_{h,k,I}^{n+1}$ given by (4.15) is computed in the compact form as

$$\mathbf{M}_{h,k,I}^{n+1}=\mathbf{P}_y \textcircled{y}\, \mathbf{P}_x \textcircled{x} \left[(e^*)^{\delta_t \mathbf{H}} \odot (\mathbf{P}_y^{-1} \textcircled{y}\, \mathbf{P}_x^{-1} \textcircled{x}\, \mathbf{M}_{h,k,I}^n)+\mathbf{Q}_k^{R,(r)}\right]. \tag{4.18}$$

One can precompute the integration of $(e^*)^{(\delta_t-\tau)\mathbf{H}} \odot (\tau/\delta_t)^s$ in (4.17), named by $\phi^s(\delta_t)$, whose elements are

$$\phi_{i,j}^s(\delta_t)=\int_0^{\delta_t} \left(\frac{\tau}{\delta_t}\right)^s e^{(\delta_t-\tau)h_{i,j}}\, d\tau.$$

The integrals can be evaluated analytically, for example, $\phi_{i,j}^s(\delta_t)$ for $s=0,1$ are listed in Table 1. Furthermore, when the time step size is fixed in a simulation, $\phi^s(\delta_t)$ only needs to be computed once and can be used repeatedly.

Table 1: The list of $\phi_{i,j}^s(\delta_t)$ for $s=0,1$.

$\phi_{i,j}^0(\delta_t)=-\dfrac{1}{h_{i,j}}\left(1-e^{\delta_t h_{i,j}}\right)$ if $h_{i,j}\neq 0$,	$\phi_{i,j}^0(\delta_t)=\delta_t$ if $h_{i,j}=0$,
$\phi_{i,j}^1(\delta_t)=-\dfrac{1}{h_{ij}}\left(1-\dfrac{\phi_{i,j}^0(\delta_t)}{\delta_t}\right)$ if $h_{i,j}\neq 0$,	$\phi_{i,j}^1(\delta_t)=\dfrac{\delta_t}{2}$ if $h_{i,j}=0$.

In case of $r=0$ (i.e., the ETD1 scheme (2.11)),

$$\mathbf{Q}_k^{r,(0)}=\mathbf{R}_k(t_n,\mathbf{U}_h^n,\mathbf{M}_{h,k,B}^n)\odot \mathbf{S}_{0,0}(\delta_t), \tag{4.19}$$

where $\mathbf{S}_{0,0}(\delta_t)=\phi_{i,j}^0(\delta_t)$; in case of $r=1$ (i.e., the ETDMs2 scheme (2.12));

$$\mathbf{Q}_k^{R,(1)}=\mathbf{R}_k(t_n,\mathbf{U}_h^n,\mathbf{M}_{h,k,B}^n)\odot \mathbf{S}_{1,0}(\delta_t)-\mathbf{R}_k(t_{n-1},\mathbf{U}_h^{n-1},\mathbf{M}_{h,k,B}^{n-1})\odot \mathbf{S}_{1,1}(\delta_t), \tag{4.20}$$

where $\mathbf{S}_{1,0}(\delta_t)=\phi_{i,j}^0(\delta_t)+\phi_{i,j}^1(\delta_t)$, $\mathbf{S}_{1,1}(\delta_t)=\phi_{i,j}^1(\delta_t)$. We also temporarily update the boundary values of $\mathbf{M}_{h,k}$, $\mathbf{M}_{h,k,B}^{n+1}$ by (2.13) when $r=0$ and by (2.14) when $r=1$.

Component II The Poisson equation (2.15) with zero Neumann boundary condition for the gauge variable can be expressed in the compact form as

$$\left(\widehat{\mathbf{A}} \textcircled{x} + \widehat{\mathbf{B}} \textcircled{y}\right) \mathbf{\Phi}_h^{n+1}=\mathbf{D}\mathbf{M}_h^{n+1}, \tag{4.21}$$

where $\mathbf{D}\mathbf{M}_h^{n+1}$ is calculated by $\mathbf{M}_h^{n+1}$ obtained from *Component I* and the divergence $\mathbf{D}$ defined in (4.2). A fast Poisson solver based on FFT [21] can be used to solve (4.21).

Component III The boundary values of the auxiliary field $\mathbf{m}_{h,k,B}^{n+1}$, for $k=1,\cdots,d$, are then *corrected* by using $\mathbf{\Phi}_h^{n+1}$ computed from *Component II* and Eq. (2.16).

When nonlinear terms appear in a system such as Navier-Stokes equations, after the above steps, $\mathbf{U}_h^{n+1}$ will be recovered by $\mathbf{U}_h^{n+1}=\mathbf{M}_h^{n+1}-\mathbf{G}\mathbf{\Phi}_h^{n+1}$, where the gradient $\mathbf{G}$ is defined in (4.4).

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

5 Numerical experiments

In this section, we perform several numerical experiments on testing the ETD Gauge method for solving Stokes and Navier-Stokes problems in rectangular domains. Four prototype problems are considered: the first two, where the exact solutions are known, are used to illustrate the rates of convergence of the proposed methods in time and space; the last two are driven-cavity flow problems in two and three dimensional spaces, respectively. Furthermore, the driven-cavity flow problems are closely related to the application of flows over and within cavities in many engineering problems, ranging from the small cavities due to gaps in the body work of vehicles to the larger scale flows in urban street canyons. We apply the fast solver developed in the preceding section for all the tests.

5.1 Stokes equation

Consider the Stokes equation (that is, Eq. (1.1) with $F(\mathbf{u})=0$) on a square domain $\Omega=[0,1]^2$ over the time interval $[0,1]$. The exact solution for the velocity is given by

$$\mathbf{u}(t,x,y)=\pi\sin t(\sin 2\pi y\sin^2\pi x,-\sin 2\pi x\sin^2\pi y) \tag{5.1}$$

and for the pressure is

$$p(x,y,t)=\sin t\cos\pi x\sin\pi y. \tag{5.2}$$

The source term $\mathbf{f}=\mathbf{u}_t-\nu\Delta\mathbf{u}+\nabla p$ is calculated correspondingly from the exact solution. The ending time for all simulations is set to be $T=1$. The zero Dirichlet boundary condition is used for the velocity field.

Taking the viscosity coefficient $\nu=1$, we first test the spatial convergence of the ETD gauge method by using the ETDMs2 scheme with a fixed small time step size; then we test the respective temporal convergence of the ETD1 and ETDMs2 schemes by fixing a small spatial grid size and varying the time step size. The approximation errors of the velocity, divergence and pressure measured in the L_∞ norm and associated convergence rates are reported in Table 2. It is observed that, for velocity and pressure in all the test cases, the approximation errors decay at the expected optimal rates; and they quickly reach the approximation error plateaus caused by either the fixed spatial grid sizes or the time step sizes. In particular, the spatial accuracies of the velocity and the pressure are both of second order; the temporal accuracy order is one for the ETD1 scheme and two for the ETDMs2 scheme respectively. On the other hand, the errors of the divergence-free condition decrease quadratically along the refinement of the spatial grids, but they are almost independent of the time step sizes for both ETD schemes.

Next taking a small viscosity coefficient $\nu=10^{-3}$, we conduct the same type of tests. The errors and corresponding rates of convergence are reported in Table 3. The ETD schemes perform similar to that in the preceding case: for the velocity and pressure, the optimal approximation errors are achieved; for the divergence of velocity, the approximation errors converge quadratically along the spatial grid refinement when a small fixed

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Table 2: Numerical errors and convergence rates of the ETD gauge method for the 2D Stokes flow with the viscosity $\nu=1$ and the exact solution (5.1) and (5.2).

(N_x, N_y, K)	$\|\mathbf{u}-\mathbf{u}_h\|_\infty$	Rate	$\|\nabla\cdot\mathbf{u}_h\|_\infty$	Rate	$\|p-p_h\|_\infty$	Rate
Spatial convergence						
(16, 16, 1024)	3.387e-02	–	7.151e-03	–	3.537e-02	–
(32, 32, 1024)	8.421e-03	2.01	2.030e-03	1.82	1.182e-02	1.58
(64, 64, 1024)	2.103e-03	2.00	5.422e-04	1.90	3.708e-03	1.67
(128, 128, 1024)	5.264e-04	2.00	1.392e-04	1.96	1.116e-03	1.73
(256, 256, 1024)	1.324e-04	1.99	3.524e-05	1.98	3.265e-04	1.77
(512, 512, 1024)	3.390e-05	1.97	8.860e-06	1.99	9.360e-05	1.80
Temporal convergence of ETD1						
(1024,1024, 16)	8.265e-02	–	8.787e-04	–	1.344e-01	–
(1024,1024, 32)	3.510e-02	1.24	3.093e-04	1.50	6.457e-02	1.06
(1024,1024, 64)	1.560e-02	1.17	1.417e-04	1.13	2.644e-02	1.29
(1024,1024, 128)	7.258e-03	1.10	7.496e-05	0.92	1.032e-02	1.36
(1024,1024, 256)	3.485e-03	1.06	3.836e-05	0.97	4.103e-03	1.33
(1024,1024, 512)	1.703e-03	1.03	1.997e-05	0.94	1.833e-03	1.16
Temporal convergence of ETDMs2						
(1024,1024, 16)	6.809e-03	–	3.547e-03	–	9.553e-02	–
(1024,1024, 32)	1.462e-03	2.22	8.583e-04	2.05	1.098e-02	3.12
(1024,1024, 64)	3.291e-04	2.15	1.006e-05	6.41	6.220e-04	4.14
(1024,1024, 128)	8.239e-05	2.00	2.142e-06	2.23	1.193e-04	2.38
(1024,1024, 256)	2.596e-05	1.67	2.081e-06	0.04	2.902e-05	2.04
(1024,1024, 512)	1.255e-05	1.05	2.067e-06	0.01	2.683e-05	0.11

time step size is used; and it reaches the error plateau quickly in the temporal refinement tests where a small fixed spatial grid size is considered.

These results also numerically demonstrate that the gauge approach-based ETD1 and ETDMs2 schemes are both unconditionally stable for the linear Stokes equation. With the same spatial grid size and time step size, the larger the viscosity ν is, the smaller the errors are.

The same test problem with $\nu=1$ was used in the review article [11]: for the standard projection method, a numerical boundary layer appears in the pressure approximation; for the improved, rotational forms of the project method, large spikes of the pressure approximation error occur at the four corners of the domain. We also note that several revised projection approaches have been considered later to reduce the splitting error and alleviate the boundary layer [1,12,13,24]. On the other hand, the proposed ETD schemes of the gauge formulation yields the optimal L_∞ errors for the pressure approximations as shown in Table 2, which indicates our schemes do not introduce any artificial, inconsistent boundary condition on the pressure. To this end, we plot in Fig. 2 the pressure

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Table 3: Numerical errors and convergence rates of the ETD gauge method for the 2D Stokes flow with the viscosity $\nu=10^{-3}$ and the exact solution (5.1) and (5.2).

(N_x, N_y, K)	$\|\mathbf{u}-\mathbf{u}_h\|_\infty$	Rate	$\|\nabla\cdot\mathbf{u}_h\|_\infty$	Rate	$\|p-p_h\|_\infty$	Rate
Spatial convergence						
(16, 16, 1024)	3.008e-02	–	4.292e-01	–	1.233e-02	–
(32, 32, 1024)	6.914e-03	2.12	1.859e-01	1.21	3.222e-03	1.94
(64, 64, 1024)	1.693e-03	2.03	6.242e-02	1.57	8.753e-04	1.88
(128, 128, 1024)	4.234e-04	2.00	1.802e-02	1.79	2.416e-04	1.86
(256, 256, 1024)	1.057e-04	2.00	4.835e-03	1.90	6.676e-05	1.86
(512, 512, 1024)	2.645e-05	2.00	1.250e-03	1.95	1.835e-05	1.86
Temporal convergence of ETD1						
(1024,1024, 16)	1.305e-01	–	8.686e-01	–	3.218e-02	–
(1024,1024, 32)	6.448e-02	1.02	5.138e-01	0.76	1.683e-02	0.93
(1024,1024, 64)	3.094e-02	1.06	2.645e-01	0.96	8.219e-03	1.03
(1024,1024, 128)	1.449e-02	1.09	1.312e-01	1.01	3.944e-03	1.06
(1024,1024, 256)	6.634e-03	1.13	6.399e-02	1.04	1.889e-03	1.06
(1024,1024, 512)	2.995e-03	1.15	3.087e-02	1.05	9.096e-04	1.05
Temporal convergence of ETDMs2						
(1024,1024, 16)	5.741e-03	–	2.924e-02	–	1.381e-03	–
(1024,1024, 32)	1.299e-03	2.14	1.625e-03	4.17	3.449e-04	2.00
(1024,1024, 64)	2.956e-04	2.14	7.209e-04	1.17	8.624e-05	2.00
(1024,1024, 128)	6.645e-05	2.15	3.677e-04	0.97	2.163e-05	2.00
(1024,1024, 256)	1.628e-05	2.03	3.095e-04	0.25	5.490e-06	1.98
(1024,1024, 512)	6.873e-06	1.24	3.020e-04	0.03	4.919e-06	0.16

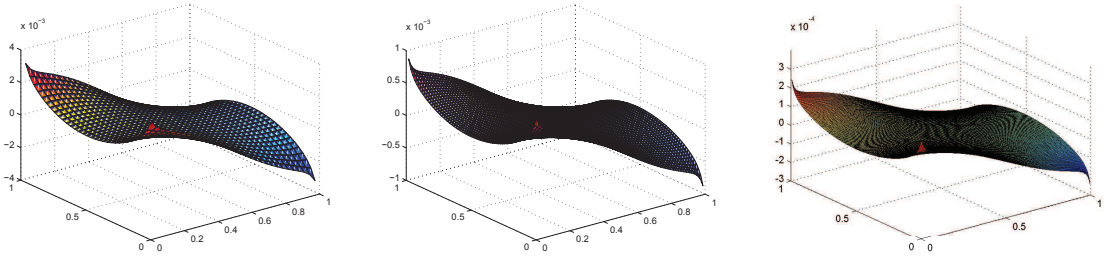


Figure 2: Pressure approximation errors of the ETDMs2 scheme for the 2D Stokes flow with the viscosity $\nu=10^{-3}$ and the exact solution (5.1) and (5.2) when $h_x=h_y=\delta_t=\frac{1}{32}$, $h_x=h_y=\delta_t=\frac{1}{64}$ and $h_x=h_y=\delta_t=\frac{1}{128}$ (from left to right).

approximation errors when the ETDMs2 scheme is used. It is seen that the error profiles keep the same shape as the spatial grid size and time step size uniformly decrease, while the magnitudes decay quadratically (no any boundary layer is observed).

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

5.2 Navier-Stokes equation

The other important model of incompressible viscous flows is the Navier-Stokes equation (that is, $F(\mathbf{u})=(\mathbf{u}\cdot\nabla)\mathbf{u}$ in (1.1)). The performance of the ETD gauge method is studied through three test problems. Among them, the first two are two dimensional and the last one is three dimensional.

Case I The purpose of this experiment is to numerically illustrate the optimal convergence and accuracy of the ETD gauge method when the system is nonlinear. We consider the 2D Navier-Stokes flow on a square box $\Omega=[0,1]^2$ with the exact solution same as (5.1) for the velocity and (5.2) for the pressure. The source term $\mathbf{f}=\mathbf{u}_t-\nu\Delta\mathbf{u}+\mathbf{u}\cdot\nabla\mathbf{u}+\nabla p$ is again determined from the exact solutions and the viscosity is selected to be $\nu=1$. The simulation ending time is still $T=1$.

The approximation errors in the L_∞ norm and rates of convergence are shown in Table 4. The same numerical behavior as that in the Stokes examples is also observed in this test case. It implies the ETD schemes are stable for the Navier-Stokes equation when the convection is not dominated.

Table 4: Numerical errors and convergence rates of the ETD gauge method for the 2D Navier-Stokes flow with the viscosity $\nu=1$ and the exact solution (5.1) and (5.2).

(N_x, N_y, K)	$\|\mathbf{u}-\mathbf{u}_h\|_\infty$	Rate	$\|\nabla\cdot\mathbf{u}_h\|_\infty$	Rate	$\|p-p_h\|_\infty$	Rate
Spatial convergence						
(16, 16, 1024)	3.387e-02	–	7.152e-03	–	3.639e-02	–
(32, 32, 1024)	8.421e-03	2.01	2.031e-03	1.82	1.202e-02	1.60
(64, 64, 1024)	2.103e-03	2.00	5.422e-04	1.91	3.752e-03	1.68
(128, 128, 1024)	5.264e-04	2.00	1.392e-04	1.96	1.126e-03	1.74
(256, 256, 1024)	1.324e-04	1.99	3.524e-05	1.98	3.279e-04	1.78
(512, 512, 1024)	3.390e-05	1.97	8.860e-06	1.99	9.460e-05	1.79
Temporal convergence of ETD1						
(1024,1024, 16)	8.262e-02	–	2.297e-03	–	4.308e-01	–
(1024,1024, 32)	3.509e-02	1.24	3.665e-04	2.65	1.814e-01	1.25
(1024,1024, 64)	1.560e-02	1.17	1.495e-04	1.29	8.016e-02	1.18
(1024,1024, 128)	7.256e-03	1.10	8.034e-05	0.90	3.726e-02	1.11
(1024,1024, 256)	3.484e-03	1.06	4.047e-05	0.99	1.792e-02	1.06
(1024,1024, 512)	1.703e-03	1.03	2.058e-05	0.98	8.780e-03	1.03
Temporal convergence of ETDMs2						
(1024,1024, 16)	6.807e-03	–	4.235e-03	–	1.122e-01	–
(1024,1024, 32)	1.461e-03	2.22	9.019e-04	2.23	1.217e-02	3.20
(1024,1024, 64)	3.290e-04	2.15	1.012e-05	6.48	1.642e-03	2.89
(1024,1024, 128)	8.237e-05	2.00	2.143e-06	2.24	3.797e-04	2.11
(1024,1024, 256)	2.596e-05	1.67	2.081e-06	0.04	9.061e-05	2.07
(1024,1024, 512)	1.254e-05	1.05	2.063e-06	0.01	3.240e-05	1.48

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Case II We apply the proposed ETD gauge method to the 2D lid-driven cavity flow problems. The fluid is contained in a square domain $\Omega=[0,1]^2$ and is initially at rest. The flow is then driven by a moving lid whose speed is unity. The cavity flow satisfies Dirichlet boundary conditions, in particular, its velocity has the tangential component $u=1$ and vertical component $v=0$ on the top side of the box; and the velocity is zero on all the other sides. It is known that the flow structure inside the cavity is governed by the Reynolds number, $\mathrm{Re}=1/\nu$. We test the flows with different Reynolds number $\mathrm{Re}=200$, 1000, 2500, and 5000 by using the ETDMs2 scheme. The flows were simulated until steady-states are achieved. For $\mathrm{Re}=200$, a grid with $h_x=h_y=1/128$ was used and final time T=20; for Re=1000, $h_x=h_y=1/128$ and T=40; for Re=2500, $h_x=h_y=1/256$ and $T=60$; for $\mathrm{Re}=5000$, $h_x=h_y=1/512$ and $T=80$. Since the viscosity is small in these tests, we enforce the CFL condition (see Remark 3.1) by taking the time step size δ_t equal or slightly smaller than the spatial grid size. It is observed that the flow reaches the steady state at the final time T in all cases. The corresponding vorticity and stream function are computed, and their contour plots together with that of the horizontal component of the velocity are shown in Fig. 3.

When $\mathrm{Re}=200$, the flow exhibits a large primary vortex with a secondary vortex in the right bottom corner. As Re increases to 1000, another secondary vortex appears at the left bottom corner in addition to the primary vortex and the secondary vortex at the right bottom corner. For the cases of $\mathrm{Re}=2500$ and 5000, a third vortex emerges in the upper left corner. Furthermore, by comparing the simulation results with those obtained on the grids with a respective double resolutions, we find the outputs to be very consistent. We remark that these numerical results also match very well the observations reported in the literature, e.g. [5].

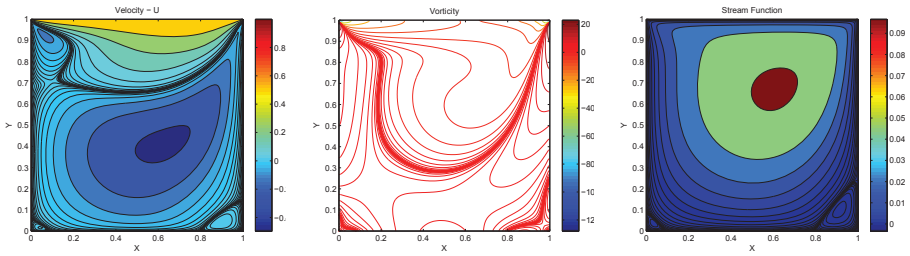
Case III Next, we consider the 3D cavity flow in a unit cube driven by a moving lid. The fluid is at rest initially and starts moving while the top lid is dragging at a constant unit speed in x-direction. The velocity satisfies the zero Dirichlet boundary condition on all the other sides. The evolution of the flows was again simulated by using the ETDMs2 scheme.

Varying the Reynolds number, with $\mathrm{Re}=200$ or 1000, we plot the contours of vorticity and the velocity vector field on the middle slice planes $x=\frac{1}{2}$, $y=\frac{1}{2}$ and $z=\frac{1}{2}$ respectively in Fig. 4. The horizontal velocity of the fluid at a centerline, which is the line through the center of the cube and parallel to the z-axis, is plotted in Fig. 5. The results match those in literature such as [35].

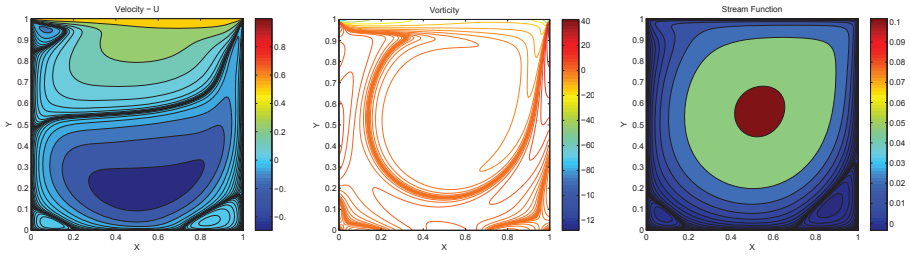
6 Conclusions

The gauge formulation of the incompressible viscous flows applies the Hodge decomposition at the PDE level and decomposes the original model into a parabolic problem of the auxiliary field $\mathbf{m}$ and a Poisson problem of the gauge variable ϕ. These two prob-

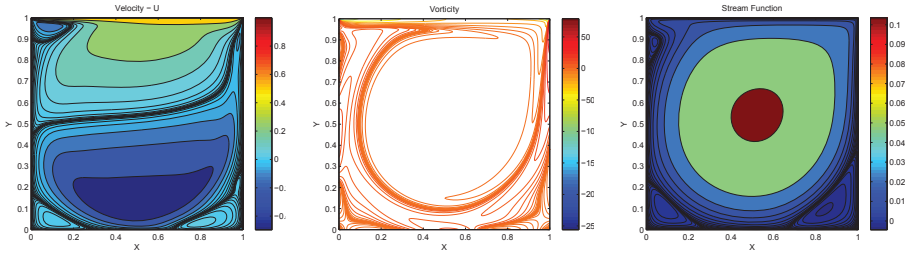
https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press



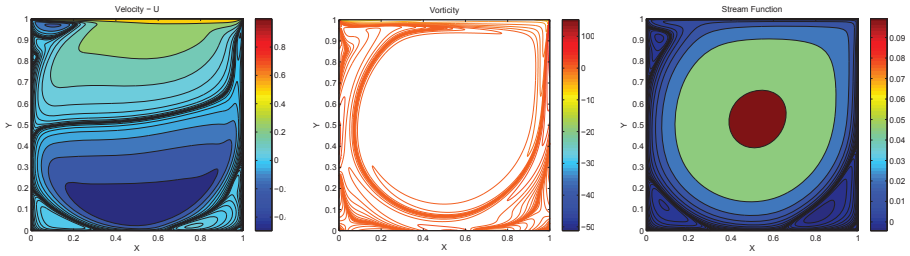
(a) Re$=200$, $T=20$, $h_x=h_y=\delta_t=\frac{1}{128}$



(b) Re$=1000$, $T=40$, $h_x=h_y=\delta_t=\frac{1}{128}$



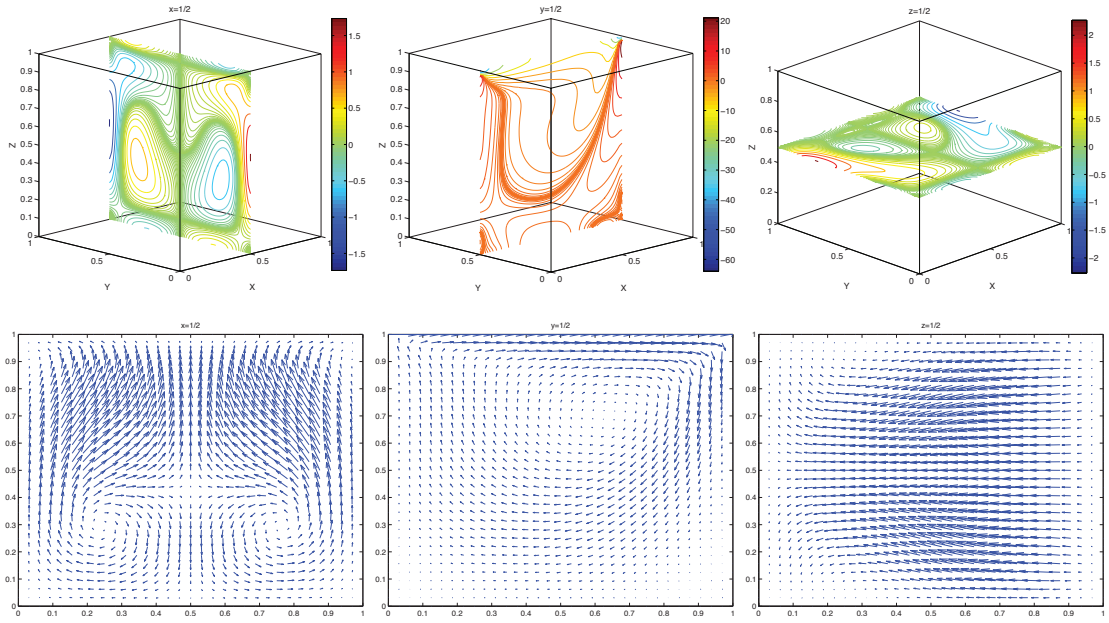
(c) Re$=2500$, $T=60$, $h_x=h_y=\frac{9}{8}\delta_t=\frac{1}{256}$



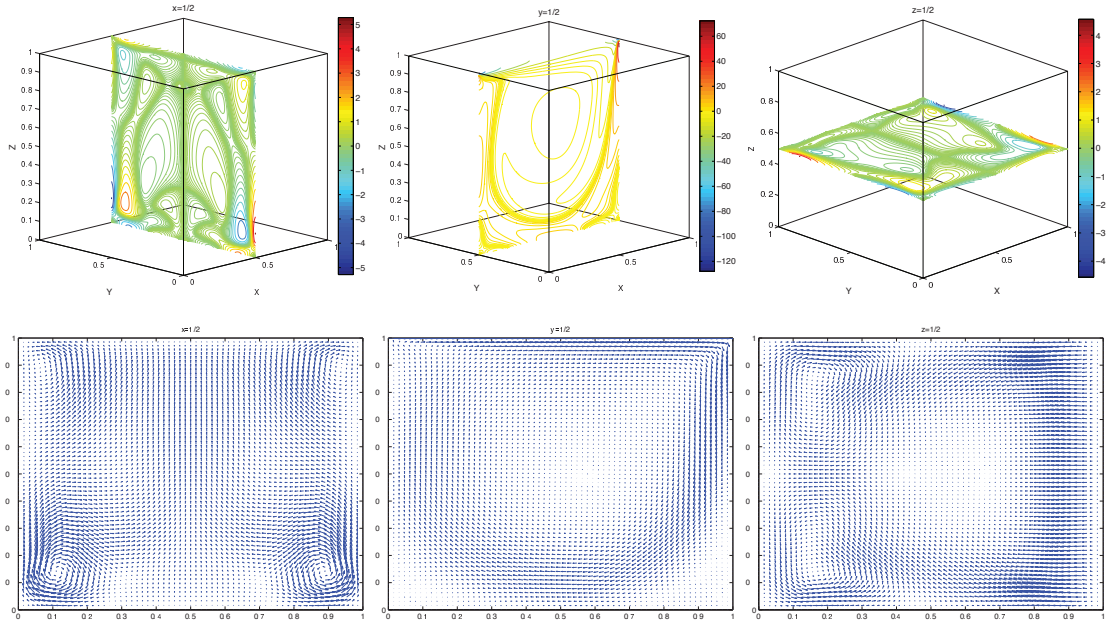
(d) Re$=5000$, $T=80$, $h_x=h_y=\frac{9}{8}\delta_t=\frac{1}{512}$

Figure 3: Lid-driven cavity flows at various Reynolds numbers in a 2D unit square: contour plots of the horizontal component of the velocity (left), the vorticity (middle) and the stream function (right) at the time T obtained by using the ETDMs2 gauge scheme.

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press



(a) Re$=200$, $T=20$, $h_x=h_y=h_z=\delta_t=1/128$



(b) Re$=1000$, $T=40$, $h_x=h_y=h_z=\delta_t=1/128$

Figure 4: Lid-driven cavity flows at Re$=200$ and 1000 respectively in a 3D unit box: the vorticity contours (first and third rows) and the velocity vector fields (second and fourth rows) on the middle-planes $x=\frac{1}{2}$ (left), $y=\frac{1}{2}$ (middle) and $z=\frac{1}{2}$ (right) at time T, respectively.

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

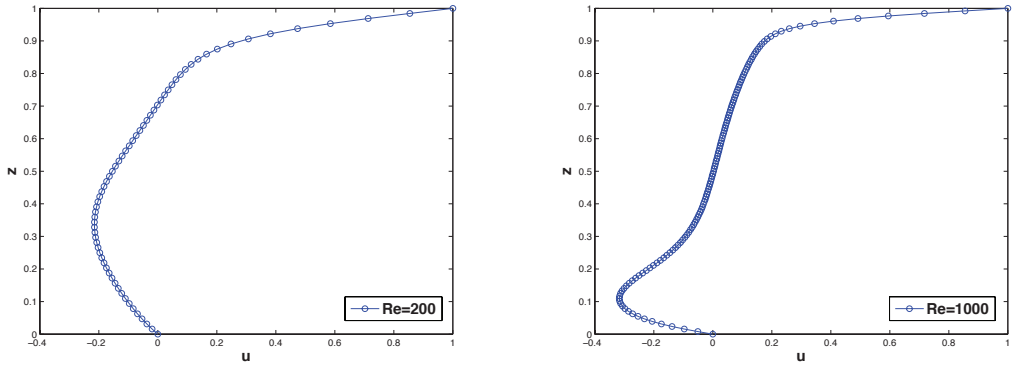


Figure 5: Horizontal velocity component u in the 3D centerline parallel to the z-axis: Re$=200$, $T=20$ (left) and Re$=1000$, $T=40$ (right).

lems are weakly coupled through the consistent boundary conditions. In this paper, we develop the ETD multistep methods for the gauge system. At each time step, it treats the boundary of $\mathbf{m}$ explicitly. After solving $\mathbf{m}$ and ϕ successively, the boundary values of $\mathbf{m}$ are updated so that the Hodge decomposition is satisfied at the discrete level. With a half-staggered grid for the spatial discretization, the ETD schemes are numerically consistent. Therefore, no artificial boundary layer appears in the pressure approximation and the associated error in the L_∞ norm converges optimally.

The proposed first-order ETD method is proven to be unconditionally stable for the Stokes system. A compact representation of the ETD methods is obtained for problems on rectangular domains, which makes FFT-based solvers available for the resulting fully discrete problem, hence, is computationally efficient. Various numerical experiments are carried out to demonstrate the accuracy and stability of the proposed method. Although we only present first order and second order ETD methods, higher order ETD schemes can be derived in the same manner. Furthermore, the theoretical stability analysis of ETDMs2 or higher order ETD methods is still under our investigation.

Acknowledgments

The first author is grateful to the research support from the U.S. National Science Foundation through the grant DMS-1521965 and the U.S. Department of Energy through the grants DE-SC0008087-ER65393 and DE-SC0016540. The second author is grateful to the research support from the U.S. National Science Foundation through the grant DMS-1522672 and the U.S. Department of Energy through the grant DE-SC0016540.

References

[1] D. L. BROWN, R. CORTEZ AND M. L. MINION, Accurate projection methods for the incom-

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

pressible Navier–Stokes equations, *J. Comput. Phys.*, **168**(2), pp. 464–499, 2001.
[2] M. P. CALVO, C. PALENCIA, A class of explicit multistep exponential integrators for semilinear problems, *Numer. Math.*, **102**, pp. 367–381, 2006.
[3] S. COX AND P. MATTHEWS, Exponential time differencing for stiff systems, *J. Comput. Phys.*, **176**, pp. 430–455, 2002.
[4] A. J. CHORIN, Numerical solution of the Navier-Stokes equations, *Math. Comp.*, **22**, pp. 745–762, 1968.
[5] E. ERTURK, Comparison of wide and compact fourth-order formulations of the Navier-Stokes equations, *Inter. J. Numer. Meth. Fluids*, **60**, pp. 992–1010, 2009.
[6] W. E AND J.-G. LIU, Finite difference schemes for incompressible flows in the velocity-impulse density formulation, *J. Comput. Phys.*, **130**, pp. 67–76, 1997.
[7] W. E AND J.-G. LIU, Gauge finite element method for incompressible flows, *Int. J. Num. Meth. Fluids*, **34**, pp. 701–710, 2000.
[8] W. E AND J.-G. LIU, Gauge method for viscous incompressible flows, *Comm. Math. Sci.*, **1**, pp. 317–332, 2003.
[9] E. GALLOPOULOS AND Y. SAAD, Efficient solution of parabolic equations by Krylov approximation methods, *SIAM J. Sci. Stat. Comput.*, **13**, pp. 1236–1264, 1992.
[10] G. GOLUB, L.-C. HUANG, H. SIMON, AND W.-P. TANG, A fast Poisson solver for the finite difference solution of the incompressible Navier-Stokes equations, *SIAM J. Sci. Comput.*, **19**, pp. 1606–1624, 1998.
[11] J. L. GUERMOND, P. MINEV AND J. SHEN, An overview of projection methods for incompressible flows, *Comput. Methods Appl. Mech. Engrg.*, **195**, pp. 6011–6045, 2006.
[12] J. L. GUERMOND AND A. SALGADO, A splitting method for incompressible flows with variable density based on a pressure Poisson equation, *J. Comput. Phys.*, **228**(8), pp. 2834–2846, 2009.
[13] J. L. GUERMOND AND A. SALGADO, Error analysis of a fractional time-stepping technique for incompressible flows with variable density, *SIAM J. Numer. Anal.*, **49**(3), pp. 917–944, 2011.
[14] M. D. GUNZBURGER, *Perspectives in Flow Control and Optimization*, Society for Industrial and Applied Mathematics, 2003.
[15] M. D. GUNZBURGER, *Finite Element Methods for Viscous Incompressible Flows: a guide to theory, practice, and algorithms*, Academic Press, INC., 1989.
[16] M. HOCHBRUCKY AND C. LUBICH, On Krylov subspace approximations to the matrix exponential operator, *SIAM J. Numer. Anal.*, **34**, pp. 1911–1925, 1997.
[17] M. HOCHBRUCK, C. LUBICH AND H. SELHOFER, Exponential integrators for large systems of differential equations, *SIAM J. Sci. Comput.*, **19**, pp. 1552–1574, 1998.
[18] M. HOCHBRUCK AND A. OSTERMANN, Exponential integrators, *Acta Numerica*, **19**, pp. 209–286, 2010.
[19] M. HOCHBRUCK AND A. OSTERMANN, Explicit exponential Runge-Kutta methods for semilinear parabolic problems, *SIAM J. Numer. Anal.*, **43**, pp. 1069–1090, 2005.
[20] M. HOCHBRUCK AND A. OSTERMANN, Exponential multistep methods of Adams-type, *BIT Numer. Math.*, **51**, pp. 889–908, 2011.
[21] L. JU, J. ZHANG, L. ZHU AND Q. DU, Fast explicit integration factor methods for semilinear parabolic equations, *J. Sci. Comput.*, **62**, pp. 431–455, 2015.
[22] S. KROGSTAD, Generalized integrating factor methods for stiff PDEs, *J. Comput. Phys.*, **203**, pp. 72–88, 2005.
[23] A. K. KASSAM AND L. N. TREFETHEN, Fourth-order time stepping for stiff PDEs, *SIAM J.*

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press

Sci. Comput., **26**, pp. 1214–1233, 2005.
[24] L. G. REBHOLZ AND M. XIAO, On reducing the splitting error in Yosida methods for the Navier–Stokes equations with grad-div stabilization, *Comput. Methods Appl. Mech. Engrg.*, **294**, pp. 259–277, 2015.
[25] W. LAYTON, *Introduction to the numerical analysis of incompressible viscous flows*, Society for Industrial and Applied Mathematics, 2008.
[26] C. V. LOAN, *Computational Frameworks for the Fast Fourier Transform*, Society for Industrial and Applied Mathematics, 1992.
[27] Q. NIE, F. WAN, Y.-T. ZHANG AND X. LIU, Compact integration factor methods in high spatial dimensions, *J. Comput. Phys.*, **227**, pp. 5238–5255, 2008.
[28] Q. NIE, Y.-T. ZHANG AND R. ZHAO, Efficient semi-implicit schemes for stiff systems, *J. Comput. Phys.*, **214**, pp. 521–537, 2006.
[29] R. NOCHETTO AND J.-H. PYO, Error estimates for semi-discrete gauge methods for the Navier-Stokes equations, *Math. Comput.*, **74**, pp. 521–542, 2005.
[30] R. NOCHETTO AND J.-H. PYO, The Gauge-Uzawa finite element method. Part I: the Navier-Stokes equations, *SIAM J. Numer. Anal.*, **43**, pp. 1043–1068, 2005.
[31] J.-H. PYO AND J. SHEN, Gauge-Uzawa methods for incompressible flows with variable density, *J. Comput. Phys.*, **221**, pp. 181–197, 2007.
[32] R. TEMAM, Sur l'approximation de la solution des équations de navier-stokes par la méthode des pas fractionnaires (ii), *Archive for Rational Mechanics and Analysis*, **33**, pp. 377–385, 1969.
[33] R. TEMAM, *Navier-Stokes Equations: Theory and Numerical Analysis*, North-Holland, 1977.
[34] M. TOKMAN, *Efficient integration of large stiff systems of ODEs with exponential propagation iterative (EPI) methods*, *J. Comput. Phys.*, **213**, pp. 748–776, 2006.
[35] K. L. WONG AND A. J. BAKER, *A 3D incompressible Navier–Stokes velocity–vorticity weak form finite element algorithm*, *Inter. J. Num. Meth. Fluids*, **38**(2), pp 99–123, 2002.
[36] C. WANG AND J.-G. LIU, Convergence of gauge method for incompressible flow, *Math. Comp.*, **69**(232), pp. 1385–1407, 2000.
[37] L. ZHU, L. JU AND W. ZHAO, Fast high-order compact exponential time differencing Runge–Kutta methods for second-order semilinear parabolic equations, *J. Sci. Comput.*, **67**(3), pp. 1043–1065, 2016.

https://doi.org/10.4208/cicp.OA-2016-0234 Published online by Cambridge University Press