Journal of Computational Physics 229 (2010) 4291–4303

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Velocity–vorticity–helicity formulation and a solver for the Navier–Stokes equations

Maxim A. Olshanskii [a,*,1], Leo G. Rebholz [b,2]

[a] Department of Mechanics and Mathematics, Moscow State M.V. Lomonosov University, Moscow 119899, Russia
[b] Department of Mathematical Sciences, Clemson University, Clemson, SC 29634, United States

ARTICLE INFO

Article history:
Received 3 June 2009
Received in revised form 9 November 2009
Accepted 15 February 2010
Available online 20 February 2010

Keywords:
Incompressible Navier–Stokes equations
Finite element method
Vorticity
Helicity

ABSTRACT

For the three-dimensional incompressible Navier–Stokes equations, we present a formulation featuring velocity, vorticity and helical density as independent variables. We find the helical density can be observed as a Lagrange multiplier corresponding to the divergence-free constraint on the vorticity variable, similar to the pressure in the case of the incompressibility condition for velocity. As one possible practical application of this new formulation, we consider a time-splitting numerical scheme based on an alternating procedure between vorticity–helical density and velocity–Bernoulli pressure systems of equations. Results of numerical experiments include a comparison with some well-known schemes based on pressure–velocity formulation and illustrate the competitiveness on the new scheme as well as the soundness of the new formulation.

1. Introduction

Numerical simulation of laminar and turbulent incompressible flows is an important subtask in many industrial applications and remains within the focus of intensive scholarly research. Incompressible viscous flows of a Newtonian fluid are modeled by the system of the Navier–Stokes equations, which read: Given a bounded, connected domain $\Omega \subset \mathbb{R}^3$ with a piecewise smooth boundary $\partial\Omega$, the simulation time T, and a force field $\mathbf{f} : Q \to \mathbb{R}^d$ (here and further $Q := (0,T) \times \Omega$), find a velocity field $\mathbf{u} : Q \to \mathbb{R}^3$ and a pressure field $p : Q \to \mathbb{R}$ such that

$$\frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\nabla)\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in} \quad Q, \tag{1.1}$$

$$\operatorname{div}\mathbf{u} = 0 \quad \text{in} \quad \overline{Q}, \tag{1.2}$$

$$\mathbf{u}|_{t=0} = \mathbf{u}_0 \quad \text{in} \quad \Omega, \tag{1.3}$$

where $\nu > 0$ is the kinematic viscosity coefficient. Some boundary conditions have to be imposed on $\partial\Omega$ to obtain a closed set of equations. We pose the Dirichlet conditions for velocity:

$$\mathbf{u} = \boldsymbol{\phi} \quad \text{on } (0,T) \times \partial\Omega, \tag{1.4}$$

with some $\boldsymbol{\phi}$ satisfying $\int_\Omega \operatorname{div}\boldsymbol{\phi} = 0,\ \forall t \in (0,T)$, but other boundary conditions are also possible.

* Corresponding author. Tel.: +7 495 9393834.
E-mail addresses: Maxim.Olshanskii@mtu-net.ru (M.A. Olshanskii), rebholz@clemson.edu (L.G. Rebholz).

[1] Partially supported by the RAS program "Contemporary problems of theoretical mathematics" through the project No. 01.2.00104588 and RFBR Grant 08-01-00415.
[2] Partially supported by National Science Foundation Grant DMS 0914478.

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.02.012

In this paper we shall consider vorticity and helical density as additional flow variables. Vorticity $\mathbf{w} = \nabla \times \mathbf{u}$ plays a fundamental role in fluid dynamics as well as in mathematical analysis of the Navier–Stokes equations and in many cases it is advantageous to describe dynamics of a flow in terms of the evolution of the vorticity. Taking a rotation of (1.1), one immediately arrives at the following vorticity equation:

$$\frac{\partial \mathbf{w}}{\partial t} - \nu\Delta\mathbf{w} + (\mathbf{u}\nabla)\mathbf{w} - (\mathbf{w}\nabla)\mathbf{u} = \nabla \times \mathbf{f}. \tag{1.5}$$

Using the definition of the material derivative and vector identity (2.2) one can write (1.5) in the equivalent form

$$\frac{D\mathbf{w}}{Dt} - \nu\Delta\mathbf{w} - \mathbb{D}(\mathbf{u})\mathbf{w} = \nabla \times \mathbf{f},$$

where the material derivative $\frac{D\mathbf{w}}{Dt}$ represents convection of $\mathbf{w}$ along the fluid particle trajectories, $-\nu\Delta\mathbf{w}$ is the vorticity diffusion and the third term $\mathbb{D}(\mathbf{u})\mathbf{w}$ is responsible for the intensification (or the decrease) of vorticity depending on the alignment of $\mathbf{w}$ with eigenvectors corresponding to positive (or negative) eigenvalues of the rate of deformation (also known as rate of strain) tensor $\mathbb{D}(\mathbf{u}) := \frac{1}{2}(\nabla\mathbf{u} + [\nabla\mathbf{u}]^T)$. Since $\mathrm{tr}[\mathbb{D}(\mathbf{u})] = \mathrm{div}\,\mathbf{u} = 0$, matrix $\mathbb{D}(\mathbf{u})$, if not identically zero, possesses both positive and negative eigenvalues. This vorticity stretching mechanism is not present in the 2D case and is believed to be of crucial importance in turbulent flow dynamics, e.g. [2,10,23].

In this paper, starting with the vorticity equation we reformulate it in terms of vorticity and helical density $h := \mathbf{u}\mathbf{w}$ as (cf. Section 2)

$$\frac{\partial \mathbf{w}}{\partial t} - \nu\Delta\mathbf{w} + 2\mathbb{D}(\mathbf{w})\mathbf{u} - \nabla h = \nabla \times \mathbf{f}. \tag{1.6}$$

This relation suggests that *the helical density can be treated as an independent variable and in this way considered as a Lagrange multiplier corresponding to the div-free condition for vorticity*, $\mathrm{div}\,\mathbf{w} = 0$, similar to pressure in (1.1) and (1.2). The helical density h relates to the helicity by $H = \int h\, d\mathbf{x}$. The helicity is a fundamental quantity in laminar and turbulent flow: it can be interpreted physically as the degree to which a flow's vortex lines are tangled and intertwined (defined precisely in terms of the total circulation and Gauss linking number of interlocking vortex filaments), is an inviscid invariant, cascades over the inertial range jointly with kinetic energy, manifests the lack of reflectional symmetry of a flow, enjoys a nontrivial topological interpretation, and is believed to be closely related to vortex breakdown [1,7,24]. Recent theoretical studies indicate a possibly important role of helical density for the regularity of the 3D Navier–Stokes solutions, cf. [5], and analytically investigate the asymptotic dichotomy of the helicity and its relation to the energy [13]. Moreover, that helicity is an inviscid invariant and is precisely balanced in the forced viscous case means that computed solutions' helicity can be used as a further diagnostic check for physical accuracy. Thus we shall consider a new formulation for the Navier–Stokes problem based on (1.6) and additional equations involving velocity and vorticity. This formulation might give another insight into the vorticity dynamics, and leads to numerical methods which directly approximate and access such physically important variables as vorticity and helicity.

To obtain a closed set of equations and computationally feasible method based on the vorticity equation, one has to complement (1.5) or (1.6) with relations linking velocity and vorticity. For this purpose, it is common to consider the Poisson equation for velocity

$$-\Delta\mathbf{u} = \nabla \times \mathbf{w} \quad \text{in} \quad Q. \tag{1.7}$$

While the velocity–pressure system (1.1)–(1.3) is usually complemented with some boundary conditions on the velocity or stress tensor, for computations using the vorticity equation it is convenient to have some boundary conditions for $\mathbf{w}$, which are typically not given *a priori*. Setting proper boundary or integral conditions for (1.5) and (1.7) (such that divergence-free velocity and vorticity fields are recovered) is a controversial subject discussed in many publications, see e.g. [9,15,22,29,30,32] and references therein. The situation becomes much easier with (1.6), which allows $\mathrm{div}\,\mathbf{w} = 0$ to be explicitly enforced by treating the helical density as an independent variable, and thus setting appropriate vorticity boundary conditions becomes simple, see Section 2. As an interesting alternative to (1.7) we shall also consider dynamic equations linking vorticity and velocity, which enforce explicitly the $\mathrm{div}\,\mathbf{u} = 0$ condition and lead to a simple stable time-stepping alternating scheme to solve the Navier–Stokes equations numerically.

n general, the advantages of using the vorticity Eq. (1.5) for numerical simulations include the following [15,17,25]: it allows access of the physically relevant variables of vortex dominated flows, simpler elliptic operators arise rather than the saddle point problems because the pressure term is eliminated, and boundary conditions can be easier to implement in external flows where the vorticity at infinity is easier to set than the pressure boundary condition. In particular, in the finite element context, the vorticity–velocity formulation produces a vorticity field that is globally continuous. This is unlike the velocity–pressure formulation. Most of these conclusions still hold if the vorticity–helicity Eq. (1.6) is used instead of (1.5).

The remainder of the paper is organized as follows. In Section 2, we derive the velocity–vorticity–helicity (VVH) formulation and prove it is equivalent to the Navier–Stokes system (1.1)–(1.4). Section 3 presents two time-splitting numerical schemes based on an alternating procedure between vorticity–helical density and velocity–Bernoulli pressure systems of equations, and proves stability of the velocity for the schemes. A finite element algorithm for the two schemes, as well as

the results of few numerical experiments, are given in Section 4. These experiments show the VVH formulation indeed leads to effective and practical numerical simulations.

2. Vorticity–Helicity equation

It is straightforward to check the following vector identities for any sufficiently smooth vector functions $\mathbf{u}, \mathbf{w}$:

$$\nabla(\mathbf{u}\mathbf{w}) = [\nabla \mathbf{w}]^T \mathbf{u} + [\nabla \mathbf{u}]^T \mathbf{w}, \tag{2.1}$$

$$[\nabla \mathbf{u}]^T \mathbf{w} = (\mathbf{w}\nabla)\mathbf{u} + \mathbf{w} \times (\nabla \times \mathbf{u}). \tag{2.2}$$

For $\mathbf{w} = \nabla \times \mathbf{u}$ the last term in (2.2) vanishes and we obtain for nonlinear terms in (1.5):

$$(\mathbf{u}\nabla)\mathbf{w} - (\mathbf{w}\nabla)\mathbf{u} = 2\mathbb{D}(\mathbf{w})\mathbf{u} - \nabla(\mathbf{u}\mathbf{w}),$$

where $2\mathbb{D}(\mathbf{w}) = \nabla \mathbf{w} + [\nabla \mathbf{w}]^T$, and

$$h = \mathbf{u}\mathbf{w}$$

is the helical density. Treating h as a new unknown we can write the vorticity equations in the form:

$$\frac{\partial \mathbf{w}}{\partial t} - \nu \Delta \mathbf{w} + 2\mathbb{D}(\mathbf{w})\mathbf{u} - \nabla h = \nabla \times \mathbf{f} \quad \text{in} \quad Q, \tag{2.3}$$

$$\operatorname{div} \mathbf{w} = 0 \quad \text{in} \quad \overline{Q}, \tag{2.4}$$

$$\mathbf{w}|_{t=0} = \nabla \times \mathbf{u}_0 \quad \text{in} \quad \Omega. \tag{2.5}$$

Some similarities between (1.1)–(2.5) are clear: like pressure in (1.1) the helical density h can be observed as a multiplier corresponding to the div-free constraint for vorticity. The Eq. (2.3) can also be written as

$$\frac{D\mathbf{w}}{Dt} - \nu \Delta \mathbf{w} + [\nabla \mathbf{w}]^T \mathbf{u} - \nabla h = \nabla \times \mathbf{f}.$$

The system (2.3)–(2.5) should be complemented with an additional equation relating $\mathbf{u}$ and $\mathbf{w}$. Following the velocity–vorticity methods based on (1.5), for this purpose one may consider (1.7). An interesting alternative is to consider the dynamic equations:

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \mathbf{w} \times \mathbf{u} + \nabla P = \mathbf{f} \quad \text{in} \quad Q, \tag{2.6}$$

$$\operatorname{div} \mathbf{u} = 0 \quad \text{in} \quad \overline{Q}, \tag{2.7}$$

$$\mathbf{u}|_{t=0} = \mathbf{u}_0 \quad \text{in} \quad \Omega, \tag{2.8}$$

where $P = \frac{1}{2}\mathbf{u}\mathbf{u} + p$ is the Bernoulli pressure variable. As we will see in the next section, coupling (2.3)–(2.5) with (2.6)–(2.8) leads to natural alternating time-splitting schemes, which can be proven stable in a certain sense. For the system (2.6)–(2.8), boundary conditions come from the original problem setting, i.e. conditions (1.4).

The vorticity system (2.3)–(2.5) can be endowed with the consistent boundary conditions:

$$\mathbf{w} = \nabla \times \mathbf{u} \quad \text{on } (0,T) \times \partial\Omega. \tag{2.9}$$

In this case h is defined up to an additive constant for any $t \in (0,T)$. This situation is similar to the definition of pressure for enclosed flows. The non-uniqueness can be easily avoided by prescribing $h = \mathbf{u}\mathbf{w}$ at arbitrary point of domain, although a more computationally sound way [6] would be to enforce a mean condition, e.g. $\int_\Omega h - \mathbf{u}\mathbf{w} = 0$. Another consistent way of setting boundary conditions is to prescribe helicity and the tangential vorticity on the boundary:

$$h = \boldsymbol{\phi}(\nabla \times \mathbf{u}) \quad \text{and} \quad \mathbf{n} \times \mathbf{w} = \mathbf{n} \times (\nabla \times \mathbf{u}) \quad \text{on } (0,T) \times \partial\Omega. \tag{2.10}$$

The equivalence of the new velocity–vorticity–helicity formulation to the original one is verified in the next theorem.

Theorem 2.1. *The system* (1.1)–(1.4) *is equivalent to the system of* Eqs. (2.3)–(2.8), *and boundary conditions* (1.4) *and* (2.9) *or* (2.10), *as well as to the system of* Eqs. (2.3)–(2.5), (1.7), *and boundary conditions* (1.4) and (2.10).

Proof. If $\mathbf{u}, p$ solves (1.1)–(1.4) then it follows immediately that $\mathbf{u}, P = p + \frac{1}{2}\mathbf{u}\mathbf{u}, \mathbf{w} = \nabla \times \mathbf{u}$, and $h = \mathbf{w}\mathbf{u}$ solves (2.3)–(2.8), (1.4) and (2.9), as well as (1.7) and (2.10).

Now suppose that $\mathbf{u}, P, \mathbf{w}, h$ is a solution to (2.3)–(2.8), (1.4) and (2.9). Applying $\nabla\times$ to (2.6) we find

$$\frac{\partial(\nabla \times \mathbf{u})}{\partial t} - \nu \Delta(\nabla \times \mathbf{u}) + 2\mathbb{D}(\mathbf{w})\mathbf{u} - \nabla(\mathbf{w}\mathbf{u}) = \nabla \times \mathbf{f}.$$

Thus for the difference $\psi = \nabla \times \mathbf{u} - \mathbf{w}$ and $r = \mathbf{w}\mathbf{u} - h$ we get thanks to (2.3)

$$\frac{\partial \psi}{\partial t} - \nu \Delta \psi + \nabla r = 0.$$

Using (2.4), (2.9), (2.5), and (2.8) we get the system of equations:

$$\frac{\partial \psi}{\partial t} - \nu \Delta \psi + \nabla r = 0 \quad \text{in} \quad Q, \tag{2.11}$$

$$\operatorname{div} \psi = 0 \quad \text{in} \quad \overline{Q}, \tag{2.12}$$

$$\psi|_{t=0} = 0 \quad \text{in} \quad \Omega, \tag{2.13}$$

$$\psi = 0 \quad \text{on} \quad (0,T) \times \partial\Omega. \tag{2.14}$$

We multiply (2.11) by ψ and integrate over Ω. Due to (2.12) and (2.14) we obtain

$$\frac{d}{dt}\|\psi\|^2 + \nu \|\nabla \psi\|^2 = 0.$$

From this equality and initial condition (2.13) we conclude $\psi = 0$. Therefore it holds

$$\mathbf{w} = \nabla \times \mathbf{u} \quad \text{and} \quad h = \mathbf{w}\mathbf{u} + c(t). \tag{2.15}$$

Substituting $\mathbf{w}$ from (2.15) into (2.6) shows that $\mathbf{u}$ solves (1.1)–(1.3) with $p = P - \frac{1}{2}\mathbf{u}\mathbf{u}$. Thus we proved the equivalence of (1.1)–(1.4) and the system of Eqs. (2.3)–(2.8), and boundary conditions (1.4) and (2.9). The case of boundary conditions (2.10) is treated similarly.

Further suppose that $\mathbf{u}$, $\mathbf{w}$, h is a solution to (2.3)–(2.5), (1.7), (2.10). The Eq. (1.7) and the first condition from (2.10) implies (cf. [29] p. 30) the equalities $\mathbf{w} = \nabla \times \mathbf{u}$ and $\operatorname{div} \mathbf{u} = 0$. Now we substitute $\mathbf{w} = \nabla \times \mathbf{u}$ into (2.3) and through simple manipulations obtain

$$\nabla \times \left(\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u}\nabla)\mathbf{u} - \mathbf{f} \right) = \nabla(h - \mathbf{w}\mathbf{u}). \tag{2.16}$$

Taking divergence of the both sides of (2.16) we find that $r = h - \mathbf{w}\mathbf{u}$ is harmonic on Ω for any $t \in (0,T)$. The second boundary condition in (2.10) implies $r = 0$ on $\partial\Omega$. Thus $r = 0$ and $h = \mathbf{w}\mathbf{u}$. Since the right-hand side in (2.16) vanishes, there exists a scalar function p such that

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u}\nabla)\mathbf{u} + \nabla p = \mathbf{f}. \qquad \square$$

Remark 2.1. Consider the following boundary conditions for the original velocity–pressure formulation:

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad \text{and} \quad \mathbf{n} \times (\nabla \times \mathbf{u}) = 0 \quad \text{on } \partial\Omega. \tag{2.17}$$

One easily verifies that such conditions lead to the following *homogeneous* boundary conditions for vorticity and helical density:

$$\mathbf{w} \times \mathbf{n} = 0 \quad \text{and} \quad h = 0 \quad \text{on } \partial\Omega. \tag{2.18}$$

Conditions (2.17) and (2.18) may be of interest for a theoretical analysis since they ensure boundary integrals vanish in typical integration by parts in the vorticity–helicity Eq. (2.3) as well as in velocity–pressure equations. It is interesting that boundary conditions (2.17) are quite often used for the analysis of the (Navier)–Stokes equations in a bounded domain, see e.g. [3,26] and references therein, and are physical motivated [8].

Remark 2.2. If we complement the vorticity–helicity equation with (2.6)–(2.8), then an obvious complication of the new setting is that compared to the original velocity–pressure formulation the number of variables is doubled. This is the price that must be paid for the direct approximation of all four physically fundamental quantities: Bernoulli pressure and helicity (both are important inviscid invariants) as well as velocity and vorticity.

3. Numerical scheme

For a moment we omit any spatial discretization and consider only time-stepping techniques. We use the notation $\mathbf{u}^n$ for an approximation to $\mathbf{u}(n\tau)$, where τ is a time step; the same notation is applied for other variables and the right-hand side. We will also use the notation $u^{n+1/2} = \frac{u^n + u^{n+1}}{2}$, and define $u^{-1} := u^0$ and $w^{-1} := w^0$. Let us consider the following two second order in time splitting algorithms based on vorticity–helicity and velocity–Bernoulli pressure equations. Note that algorithm A1 is more implicit in the vorticity step and A2 is more implicit in the velocity step. Both algorithms are second order accurate in time.

Algorithm A1

Step 1. Given $\mathbf{u}^n, \mathbf{w}^n, \mathbf{w}^{n-1}$ and $\mathbf{w}^* = \frac{3}{2}\mathbf{w}^n - \frac{1}{2}\mathbf{w}^{n-1}$, find $\mathbf{u}^{n+1}$ and $P^{n+\frac{1}{2}}$ from

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\tau} - \nu\Delta\mathbf{u}^{n+\frac{1}{2}} + \mathbf{w}^* \times \mathbf{u}^{n+\frac{1}{2}} + \nabla P^{n+\frac{1}{2}} = \mathbf{f}^{n+\frac{1}{2}} \tag{3.1}$$

$$\operatorname{div}\mathbf{u}^{n+1} = 0 \tag{3.2}$$

$$\mathbf{u}^{n+1}|_{\partial\Omega} = \boldsymbol{\phi}^{n+1} \tag{3.3}$$

Step 2. Given $\mathbf{u}^{n+1}, \mathbf{u}^n$, and $\mathbf{w}^n$ find $\mathbf{w}^{n+1}$ and $h^{n+\frac{1}{2}}$ from

$$\frac{\mathbf{w}^{n+1}-\mathbf{w}^n}{\tau} - \nu\Delta\mathbf{w}^{n+\frac{1}{2}} + 2\mathbb{D}(\mathbf{w}^{n+\frac{1}{2}})\mathbf{u}^{n+\frac{1}{2}} - \nabla h^{n+\frac{1}{2}} = \nabla \times \mathbf{f}^{n+\frac{1}{2}} \tag{3.4}$$

$$\operatorname{div}\mathbf{w}^{n+1} = 0 \tag{3.5}$$

$$\mathbf{w}^{n+1}|_{\partial\Omega} = \nabla \times \mathbf{u}^{n+1} \tag{3.6}$$

The second algorithm changes the order of the velocity–pressure and the vorticity–helicity update.

Algorithm A2

Step 1. Given $\mathbf{u}^n, \mathbf{u}^{n-1}, \mathbf{w}^n$ and $\mathbf{u}^* = \frac{3}{2}\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}$, find $\mathbf{w}^{n+1}$ and $h^{n+\frac{1}{2}}$ from

$$\frac{\mathbf{w}^{n+1}-\mathbf{w}^n}{\tau} - \nu\Delta\mathbf{w}^{n+\frac{1}{2}} + 2\mathbb{D}\left(\mathbf{w}^{n+\frac{1}{2}}\right)\mathbf{u}^* - \nabla h^{n+\frac{1}{2}} = \nabla \times \mathbf{f}^{n+\frac{1}{2}} \tag{3.7}$$

$$\operatorname{div}\mathbf{w}^{n+1} = 0 \tag{3.8}$$

$$\mathbf{w}^{n+1}|_{\partial\Omega} = \nabla \times (2\mathbf{u}^n - \mathbf{u}^{n-1}) \tag{3.9}$$

Step 2. Given $\mathbf{u}^n, \mathbf{w}^n$ and $\mathbf{w}^{n+1}$, find $\mathbf{u}^{n+1}$ and $P^{n+\frac{1}{2}}$ from

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\tau} - \nu\Delta\mathbf{u}^{n+\frac{1}{2}} + \mathbf{w}^{n+\frac{1}{2}} \times \mathbf{u}^{n+\frac{1}{2}} + \nabla P^{n+\frac{1}{2}} = \mathbf{f}^{n+\frac{1}{2}} \tag{3.10}$$

$$\operatorname{div}\mathbf{u}^{n+1} = 0 \tag{3.11}$$

$$\mathbf{u}^{n+1}|_{\partial\Omega} = \boldsymbol{\phi}^{n+1} \tag{3.12}$$

Both algorithms involve only *linear problems to solve on each time step*. At the same time they are unconditionally stable in the following sense. The velocity approximations satisfy the energy balance and an *a priori* estimate given in the following lemma.

Lemma 3.1. *Assume* $\boldsymbol{\phi} = 0$*, then*

$$\|\mathbf{u}^n\|^2 + 2\nu\sum_{k=1}^{n}\tau\|\nabla\mathbf{u}^{k-\frac{1}{2}}\|^2 = \|\mathbf{u}^0\|^2 + 2\sum_{k=1}^{n}\tau\left(\mathbf{f}^{k-\frac{1}{2}}, \mathbf{u}^{k-\frac{1}{2}}\right) \tag{3.13}$$

and

$$\|\mathbf{u}^n\|^2 + \nu\sum_{k=1}^{n}\tau\|\nabla\mathbf{u}^{k-\frac{1}{2}}\|^2 \leqslant \|\mathbf{u}^0\|^2 + \nu^{-1}\sum_{k=1}^{n}\tau\|\mathbf{f}^{k-\frac{1}{2}}\|^2 \tag{3.14}$$

Proof. These results follow immediately from the velocity step of either algorithm by multiplying by $u^{n+\frac{1}{2}}$ and integrating over the domain, summing over timesteps, and applying Cauchy-Schwarz and Young's inequalities to the forcing term. □

Remark 3.1. The statement of the lemma is easily generalized for finite element approximations and for finite difference schemes, which preserve the skew-symmetry of the vector product and keep the discrete divergence and gradient operators adjoint.

Remark 3.2. We are not aware of similar stability result in the literature on velocity–vorticity numerical schemes. Indeed, it is not clear how to approach stability estimates if instead of (2.6)–(2.8) one adopts (1.7), see also next remark.

Remark 3.3. Stability estimate (3.14) involves only the velocity variable. Getting control of $\|\mathbf{w}^n\|$ is a delicate issue. Indeed, proving a global in time L^2 estimate for vorticity would imply the regularity of weak solutions to the 3D Navier–Stokes equations, which is an outstanding open problem. Thus we may hope to obtain only local (for sufficiently small T, $\|\mathbf{f}\|$, $\|\mathbf{u}_0\|$ and ν^{-1}) stability estimates in terms of vorticity. This is, however, a rather technical issue and we will not treat it in the current paper.

Remark 3.4. Both algorithms ensure both velocity and vorticity are divergence-free without engaging any integral or non-local boundary conditions for vorticity.

Remark 3.5. Using (2.3) and (2.4) instead of (1.5) for a numerical method leads back to a saddle point problem to solve instead of elliptic one. However, due to the significant progress in solution methods for saddle point problems, see e.g. [4,12], this may be considered as a reasonable price for getting the direct access to helicity and the explicit enforcement of the div-free condition for vorticity.

4. Numerical experiments

In this section, we test the viability of the proposed formulation with three numerical experiments. We begin by defining finite element discretizations based on Algorithms A1 and A2 that use inf–sup stable element pairs, although we note that many other discretization choices could be made, including variational multiscale methods that would allow for equal order element choices, see Remark 4.3. We first test on the 3D driven cavity benchmark problem, and find the methods are able to find the correct steady state solution. The final two experiments are done using the Ethier–Steinman exact Navier–Stokes solution. The first test verifies expected asymptotic convergence rates, and the final experiment compares the new methods against commonly used methods.

4.1. Finite element method

We test the new algorithms using a finite element (FE) spatial discretization. To formalize the numerical schemes, let (X_h, Q_h) be an inf–sup stable FE pair; in our experiments we use the Taylor–Hood element pair to approximate velocity–pressure as well as vorticity–helical density. Define the space of discretely divergence free functions in X_h to be

$$V_h = \{\mathbf{v}_h \in X_h : (\operatorname{div}\mathbf{v}_h, q_h) = 0 \ \forall q_h \in Q_h\}$$

Then given $\mathbf{u}_h^0 \in V_h$, calculate $\mathbf{w}_h^0$ as the L^2 projection of $\nabla\times\mathbf{u}_h^0$ into V_h, and define $\mathbf{u}_h^{-1} = \mathbf{u}_h^0$ and $\mathbf{w}_h^{-1} = \mathbf{w}_h^0$. The fully discrete numerical algorithms for the velocity–vorticity–helicity (VVH) formulations are as follows. We formally define one algorithm, the finite element discretization for A1; the counterpart for A2 is defined in the same way. Further in numerics these schemes are denoted by (VVH1) and (VVH2), respectively.

Algorithm 4.1. (VVH1)

Step 1. Given $\mathbf{u}_h^n, \mathbf{w}_h^n$ and $\mathbf{w}_h^{n-1}$, find $\left(\mathbf{u}_h^{n+1}, P_h^{n+\frac12}\right) \in (X_h, Q_h)$ satisfying: $\mathbf{u}_h^{n+1}|_{\partial\Omega}$ interpolates $\boldsymbol{\phi}^{n+1}$ on $\partial\Omega$, and $\forall(\mathbf{v}_h, q_h) \in (X_h, Q_h)$,

$$\frac{1}{\tau}\left(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n, \mathbf{v}_h\right) + \nu\left(\nabla\mathbf{u}_h^{n+\frac12}, \nabla\mathbf{v}_h\right) + \left(\left(\frac32\mathbf{w}_h^n - \frac12\mathbf{w}_h^{n-1}\right)\times\mathbf{u}_h^{n+\frac12}, \mathbf{v}_h\right) - \left(P_h^{n+\frac12}, \operatorname{div}\mathbf{v}_h\right) + \gamma_1\left(\operatorname{div}\mathbf{u}_h^{n+\frac12}, \operatorname{div}\mathbf{v}_h\right) = \left(\mathbf{f}^{n+\frac12}, \mathbf{v}_h\right) \qquad (\operatorname{div}\mathbf{u}_h^{n+1}, q_h) = 0 \tag{4.1}$$

Step 2. Given $\mathbf{u}_h^{n+1}, \mathbf{u}_h^n$, and $\mathbf{w}_h^n$, find $\left(\mathbf{w}_h^{n+1}, h_h^{n+\frac12}\right) \in (X_h, Q_h)$ satisfying: $\mathbf{w}_h^{n+1}|_{\partial\Omega}$ interpolates the L^2 projection of $\nabla\times\mathbf{u}_h^{n+1}$ in V_h and $\forall(\mathbf{v}_h, q_h) \in (X_h, Q_h)$,

$$\frac{1}{\tau}\left(\mathbf{w}_h^{n+1} - \mathbf{w}_h^n, \mathbf{v}_h\right) + \nu\left(\nabla\mathbf{w}_h^{n+\frac12}, \nabla\mathbf{v}_h\right) + 2\left(\mathbb{D}\left(\mathbf{w}_h^{n+\frac12}\right)\mathbf{u}_h^{n+\frac12}, \mathbf{v}_h\right) + \left(h_h^{n+\frac12}, \operatorname{div}\mathbf{v}_h\right) + \gamma_2\left(\operatorname{div}\mathbf{w}_h^{n+\frac12}, \operatorname{div}\mathbf{v}_h\right) = \left(\nabla\times\mathbf{f}^{n+\frac12}, \mathbf{v}_h\right) \qquad (\operatorname{div}\mathbf{w}_h^{n+1}, q_h) = 0 \tag{4.2}$$

Remark 4.1. From the formulation of the numerical schemes, the helical density is only unique up to a time dependent additive constant. It can be recovered after each time step by solving for the constant via $\int_\Omega (h^n + k^n - \mathbf{u}^n\mathbf{w}^n)\,d\mathbf{x} = 0$. When solid walls are on the boundary, another way would be to enforce $h = 0$ directly in the scheme at these locations.

Remark 4.2. It was already mentioned in the introduction that correctly determining and implementing vorticity boundary conditions, especially at a no-slip – no-penetration wall, is important. In our implementation we use the projection of the kinematic definition of vorticity, see step 2 of Algorithm 4.1, as was first suggested in [17,18]. We note that the accuracy of the vorticity boundary condition at a no-slip wall can be increased to the second order by using more sophisticated approach based nodal velocity expansions in the near-boundary nodes as suggested in [32]. Below we present numerical tests for two enclosed flows. We remark that in the case of inflow–outflow boundaries one may prescribe velocity and vorticity profiles at the inflow boundary and Neumann boundary condition for vorticity at the outflow, see e.g. [31,32]. For the outflow velocity condition the common choice is the Neumann or convection condition for $\mathbf{u}$, see [16]. Note that

setting "do-nothing" outflow condition [19], the popular choice in FE community, together with (2.6) would lead to a nonlinear boundary condition since it originally involves kinematic rather than Bernoulli pressure variable: It would be implemented as $\int_{\Gamma_{out}} \mathbf{v}\big((\nu\nabla\mathbf{u} - (P - \frac{1}{2}\mathbf{u}^2)I)\mathbf{n}\big) = 0$.

Remark 4.3. Besides the usual Galerkin terms, Eqs. (4.1) and (4.2) involve two terms, which additionally penalize the violation of the div-free constraint by the FE velocity and vorticity. Including such terms is often a part of Petrov–Galerkin FE formulations for the Navier–Stokes equations [14] and is well-known as the grad-div stabilization [28]; $\gamma_1 \geqslant 0$ and $\gamma_2 \geqslant 0$ are user-defined stabilization parameters. The motivation for including the grad-div stabilization with $O(1)$ stabilization parameter in the velocity equation comes from recent work in [21,27], where its use in similar (rotational form) schemes was found effective at relieving the velocity error of an undesired scaling with the large error associated with the Bernoulli pressure. The vorticity equation was stabilized in a similar manner since helical density could adversely affect the vorticity error analogously to how Bernoulli or kinematic pressure acts in the velocity equation. It is not our intention to find optimal stabilization parameters, although that in itself could be an interesting topic of future study. However, the general strategy is for higher Reynolds numbers the parameters should be $O(1)$ and for lower Reynolds number they should be much smaller or zero.

Remark 4.4. In this paper we do not use any stabilization or multiscale model other than the grad-div stabilization. We note, however, that the vorticity–helicity system (2.3) and (2.4) might be more convenient for the well-developed variational multiscale modeling or Petrov–Galerkin type streamline–upwinding [20] than the vorticity Eq. (1.5), since the structure of (2.3) largely mimics the velocity equations in the convection form (1.1). In particular, in the Petrov–Galerkin method the nonlinear terms from (2.3) would give rise to numerical streamline-dissipation of the form $\delta_T \int_T ([\mathbf{u}_h \otimes \mathbf{u}_h]\,\mathbb{D}(\mathbf{w}_h)) : \mathbb{D}(\mathbf{v}_h)\mathrm{d}\mathbf{x}$ on every element T. Moreover, the fact that the helical density and Bernoulli pressure variables may share boundary and internal layers of velocity and vorticity naturally calls for using equal order elements for approximating all variables, which is a common choice for variational multiscale approach. We will address this discretization approach in more detail elsewhere.

4.2. Numerical experiment 1: The 3D driven cavity benchmark problem

We first test the ability of the methods to find the steady state solution of the well-known 3D lid driven cavity problem. The "cavity" domain is the unit cube, equipped with homogeneous Dirichlet boundary conditions for the velocity except on the $z = 1$ face, the lid, where $\mathbf{u}_{\text{lid}} = (1, 0, 0)^T$. The kinematic viscosity is set to $\nu = 0.01$, which gives Reynolds number $Re = 100$. A uniform mesh of Taylor Hood tetrahedral elements provided 33^3 velocity/vorticity nodes and 17^3 pressure/helical density nodes for a total of 207,010 degrees of freedom. These tests were run without stabilization, that is, with $\gamma_1 = \gamma_2 = 0$. The initial condition for velocity is zero in the interior and satisfies the boundary conditions. The initial condition for vorticity was defined as in Algorithms A1 and A2, that is, to be the projection of the curl of the initial velocity into the discretely divergence-free subspace.

A steady solution was computed by setting $\Delta t = \infty$, and iterating between the velocity and vorticity equations until convergence. This is identical to VVH1 and VVH2 except instead of using extrapolated terms for linearization, we use the previous iterate of the nonlinear iteration. Using a stopping criteria for the nonlinear iteration $tol = 10^{-5}$ for the relative change in velocity field, both algorithms converged to the same solution, although VVH2 converged in 18 iterations while VVH1 needed 27; this is not surprising from the physical nature of this particular problem since the first step of VVH1 would be using a zero vorticity field and thus it takes several iterations to recover. The plots of the steady state midplane velocity fields are shown in Fig. 1, and agree well with those of [32,33]. We also compute the centerline $((0.5, 0.5, z),\ 0 \leqslant z \leqslant 1)u$-velocities of the steady state solution, see Fig. 2, and find good agreement with the results of Wong and Baker [32], who used many more degrees of freedom.

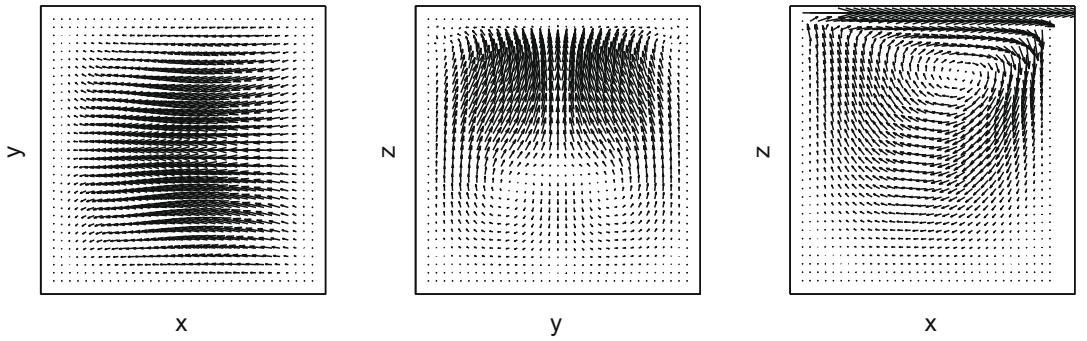

Fig. 1. Shown above is the steady state velocity fields at the midplanes, computed with VVH, of the cavity for the 3D driven cavity benchmark problem with $Re = 100$.

4.3. Numerical experiment 2: convergence rates

The goal of the next numerical experiment is to confirm expected convergence rates for a well-known 3D benchmark test problem. This will provide some measure of fidelity of both the formulation and the schemes.

We use as a test problem the well-known Ethier–Steinman exact NSE solution from [11]. This problem was developed as a 3D analogue to the Taylor vortex problem, for the purpose of benchmarking. Although unlikely to be physically realized, it is a good test problem because it is an exact NSE solution and has nontrivial helicity which implies the existence of turbulent structure [24] in its velocity field. For chosen parameters a, d and viscosity ν, the exact NSE solution is given on $[-1,1]^3$ by

$$u_1 = -a(e^{ax}\sin(ay+dz) + e^{az}\cos(ax+dy))e^{-\nu d^2 t} \tag{4.3}$$

$$u_2 = -a(e^{ay}\sin(az+dx) + e^{ax}\cos(ay+dz))e^{-\nu d^2 t} \tag{4.4}$$

$$u_3 = -a(e^{az}\sin(ax+dy) + e^{ay}\cos(az+dx))e^{-\nu d^2 t} \tag{4.5}$$

$$p = -\frac{a^2}{2}(e^{2ax} + e^{2ay} + e^{2az} + 2\sin(ax+dy)\cos(az+dx)e^{a(y+z)} + 2\sin(ay+dz)\cos(ax+dy)e^{a(z+x)} + 2\sin(az+dx)\cos(ay+dz)e^{a(x+y)})e^{-2\nu d^2 t} \tag{4.6}$$

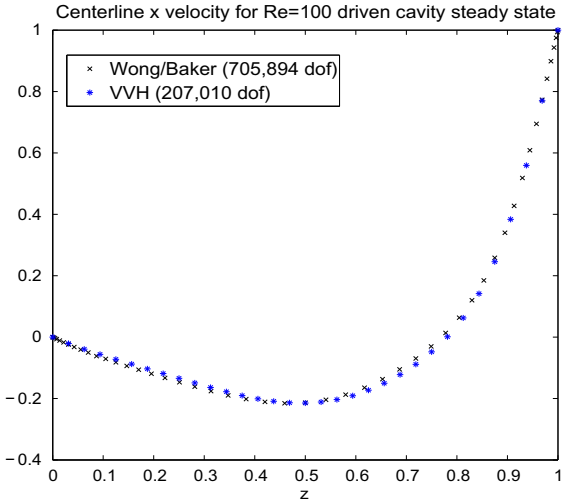

Fig. 2. u-Velocity on the centerline of the cavity compared to those from [32].

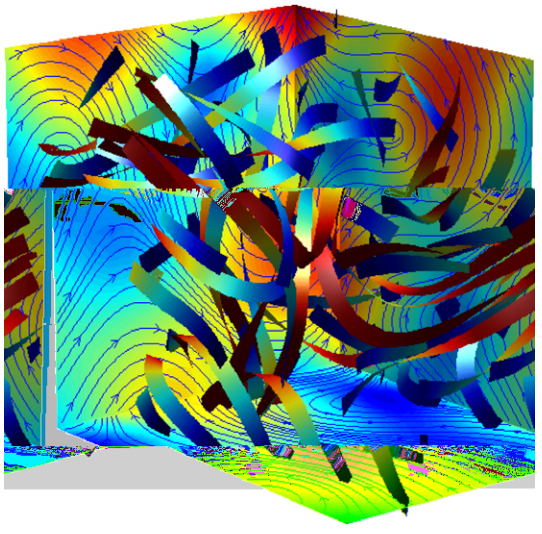

Fig. 3. The visualization of the Eithier–Steinman solution at $t = 0$.

Table 1
The velocity and Bernoulli pressure errors and convergence rates for successive mesh and timestep refinements.

h	Δt	$\|u-u_h\|_{\infty,0}$	Rate	$\|u-u_h\|_{2,1}$	Rate	$\|P-P_h\|_{2,0}$	Rate
1	0.02	0.1463	–	0.2011	–	0.3284	–
0.5	0.00667	0.0188	2.96	0.05107	1.98	0.03547	3.21
0.25	0.00222	0.002339	3.01	0.01282	2.00	0.005338	2.73
0.125	0.000741	0.000295	2.99	0.00321	2.00	0.001558	1.78

Table 2
The vorticity and helicity density errors and convergence rates for successive mesh and timestep refinements.

h	Δt	$\|\nabla\times u - w_h\|_{\infty,0}$	Rate	$\|h-h_h\|_{2,0}$	Rate
1	0.02	0.6940	–	1.133	–
0.5	0.00667	0.1395	2.31	0.3233	1.81
0.25	0.00222	0.03106	2.17	0.08834	1.87
0.125	0.000741	0.007635	2.02	0.02431	1.86

and from here the exact vorticity $\mathbf{w}$, helical density h and Bernoulli pressure P can all be easily calculated. For $t=0$ the solution is shown in Fig. 3 for parameters $a=1.25,\ d=1$: The complex flow structure is seen in the streamribbons in the box and the velocity streamlines and speed contours on the sides. The analytical solution (4.3)–(4.5) was used to prescribe velocity Dirichlet boundary conditions in numerical method. For a fair comparison we do not use (4.3)–(4.5) to define vorticity boundary conditions; they has been computed as described on step 2 of Algorithm 4.1.

Taylor–Hood (P_2,P_1) tetrahedral elements and scheme VVH1 is used to compute approximations to the test problem on 4 successive mesh refinements and corresponding timestep reductions, using parameters $a=d=\nu=1$, with end time $T=0.02$, and without stabilization $\gamma_1=\gamma_2=0$. Tables 1 and 2 show the relative errors and corresponding convergence rates. We use standard notation for the norms $\|\phi\|_{p,k}:=\|\phi\|_{L^p(0,T;H^k(\Omega)}$.

Although we do not provide a convergence analysis for the finite element schemes, Tables 1 and 2 show the scheme's velocity, vorticity and pressure converge in the given norms as one would expect under ideal conditions for (P_2,P_1) elements and a trapezoidal time discretization. Results are similar for VVH2, and so we omit them.

4.4. Numerical experiment 3: comparison to related, well-known schemes

In order to gauge a reasonable assessment of the new schemes, we additionally will test three related schemes. One is the (skew-symmetric) convective form nonlinear Galerkin finite element scheme for the Navier–Stokes equations, i.e. the finite element method applied directly to (1.1)–(1.3). This is certainly one of the most popular approaches to treat the Navier–Stokes equation numerically. Similar to algorithms (VVH1) and (VVH2) for treating nonlinear terms we use linear extrapolation, which preserves second order accuracy in time and the energy balance from Lemma 3.1. We will denote this algorithm by (Conv). Note there are two trilinear terms due to the use of skew-symmetry.

Algorithm 4.2 (*Conv*). Given $\mathbf{u}_h^n$, find $\left(\mathbf{u}_h^{n+1},p_h^{n+\frac12}\right)\in(X_h,Q_h)$ satisfying $\forall(\mathbf{v}_h,q_h)\in(X_h,Q_h)$:

$$\frac{1}{\tau}(\mathbf{u}_h^{n+1}-\mathbf{u}_h^n,\mathbf{v}_h)+\frac12\left((\mathbf{u}_h^*\nabla)\mathbf{u}_h^{n+\frac12},\mathbf{v}_h\right)-\frac12\left((\mathbf{u}_h^*\nabla)\mathbf{v}_h,\mathbf{u}_h^{n+\frac12}\right)$$
$$-\left(p_h^{n+\frac12},\operatorname{div}\mathbf{v}_h\right)+\nu\left(\nabla\mathbf{u}_h^{n+\frac12},\nabla\mathbf{v}_h\right)=\left(\mathbf{f}^{n+\frac12},\mathbf{v}_h\right)\qquad(\operatorname{div}\mathbf{u}_h^{n+1},q_h)=0$$

with $\mathbf{u}_h^*=\frac32\mathbf{u}^n-\frac12\mathbf{u}^{n-1}$.

Another closely related approach is to use the rotation form (2.6)–(2.8) of the Navier–Stokes equations, with the vorticity computed directly from the velocity approximation. Hence no vorticity equation is involved in the computations. Thus the first step of the scheme (Rot) below coincides with Step 1 of (VVH1) and the second step consists of vorticity recovery through the L^2 orthogonal projection of $\nabla\times\mathbf{u}_h$ onto velocity FE space.

Algorithm 4.3 (*Rot*).

Step 1. Given $\mathbf{u}_h^n,\mathbf{w}_h^n$ and $\mathbf{w}_h^{n-1}$, find $\left(\mathbf{u}_h^{n+1},P_h^{n+\frac12}\right)\in(X_h,Q_h)$ satisfying: $\mathbf{u}_h^{n+1}|_{\partial\Omega}$ interpolates $\boldsymbol{\phi}^{n+1}$ on $\partial\Omega$, and $\forall(\mathbf{v}_h,q_h)\in(X_h,Q_h)$,

$$\frac{1}{\tau}(\mathbf{u}_h^{n+1}-\mathbf{u}_h^n,\mathbf{v}_h)+\nu\left(\nabla\mathbf{u}_h^{n+\frac12},\nabla\mathbf{v}_h\right)+\left(\left(\frac32\mathbf{w}_h^n-\frac12\mathbf{w}_h^{n-1}\right)\times\mathbf{u}_h^{n+\frac12},\mathbf{v}_h\right)$$
$$-\left(P_h^{n+\frac12},\operatorname{div}\mathbf{v}_h\right)+\gamma_1(\operatorname{div}\mathbf{u}_h^{n+\frac12},\operatorname{div}\mathbf{v}_h)=\left(\mathbf{f}^{n+\frac12},\mathbf{v}_h\right)\qquad(\operatorname{div}\mathbf{u}_h^{n+1},q_h)=0$$

Step 2: Given $\mathbf{u}_h^{n+1}$, find $\mathbf{w}_h^{n+1} \in X_h$ satisfying $\forall \mathbf{v}_h \in X_h$:

$$(\mathbf{w}_h^{n+1}, \mathbf{v}_h) = (\nabla \times \mathbf{u}_h^{n+1}, \mathbf{v}_h)$$

We also tried the variation of the algorithm (Rot) with $\mathbf{w}_h^{n+1}$ computed via the projection of $\nabla \times \mathbf{u}_h^{n+1}$ on the space V_h of discretely div-free functions. This modification was found to have a minor influence on the algorithm performance, so we will not include it in our comparisons.

Both schemes (Conv) and (Rot) require a linear saddle point type problem to be solved on every time step. Thus the computational complexity of these schemes is comparable to the one of VVH schemes, which need two linear saddle point type problem to be solved on every time step. However, from experiments we will see that (Conv) and (Rot) appear to be far less stable than VVH schemes, especially for higher Reynolds number. Therefore, for an 'ultimately stable' scheme in primitive variables we also consider the fully implicit second order in time scheme (ConvIm) below.

Algorithm 4.4 (*ConvIm*). Given $\mathbf{u}_h^n$, find $\left(\mathbf{u}_h^{n+1}, p_h^{n+\frac{1}{2}}\right) \in (X_h, Q_h)$ satisfying $\forall (\mathbf{v}_h, q_h) \in (X_h, Q_h)$:

$$\frac{1}{\tau}(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n, \mathbf{v}_h) + \frac{1}{2}\left(\left(\mathbf{u}_h^{n+\frac{1}{2}}\nabla\right)\mathbf{u}_h^{n+\frac{1}{2}}, \mathbf{v}_h\right) - \frac{1}{2}\left(\left(\mathbf{u}_h^{n+\frac{1}{2}}\nabla\right)\mathbf{v}_h, \mathbf{u}_h^{n+\frac{1}{2}}\right)$$
$$- \left(p_h^{n+\frac{1}{2}}, \operatorname{div}\mathbf{v}_h\right) + \nu\left(\nabla\mathbf{u}_h^{n+\frac{1}{2}}, \nabla\mathbf{v}_h\right) = \left(\mathbf{f}^{n+\frac{1}{2}}, \mathbf{v}_h\right) \qquad (\operatorname{div}\mathbf{u}_h^{n+1}, q_h) = 0$$

Note that the algorithm (ConvIm) requires a nonlinear problem to be solved on every time step. A fixed point iteration was used to solve the nonlinearity at each time step. In general, this makes (ConvIm) computationally more expensive than Algorithms 4.1, 4.2, 4.3.

We compute approximations to (4.3)–(4.6) with $a = 0.75, d = 0.5$, timestep $\tau = 0.01$, endtime $T = 1$, and initial velocity $\mathbf{u}^0 = (u_1(0), u_2(0), u_3(0))^T$, using 10,368 (P_2, P_1) tetrahedral elements, enforcing Dirichlet boundary conditions from (4.3)–(4.5) on the sides of the box by setting $\boldsymbol{\phi}^n = (u_1(t^n), u_2(t^n), u_3(t^n))^T$ on $\partial\Omega$, and using $\gamma_1 = \gamma_2 = 1$.

For all five schemes, we compute for two different viscosities: $\nu = 0.01$ and $\nu = 0.001$. The results will be given in this order. Note that when $\nu = 0.01$, (Conv) and (ConvIm) are indistinguishable, and so for more readable plots they are plotted together. For $\nu = 0.001$, the results of these two scheme are different, and thus are plotted separately.

From the theoretical point view all five schemes are of the second order in time and formally obey the energy stability estimate in (3.14). Indeed, Fig. 4 (left) shows that the kinetic energy stays well-bounded in time with all five schemes for $\nu = 0.01$, although for the (Rot) scheme the balance of energy appears somewhat perturbed for larger times.

For $\nu = 0.01$, for all five schemes we plot the velocity and vorticity errors versus time in Figs. 4(right) and 5, respectively. Since vorticity is not directly solved for in the velocity–pressure scheme (Conv) or (ConvIm), $\nabla \times \mathbf{u}_h$ is used to compute the estimate of vorticity in this algorithm. It is clear in both figures that schemes (VVH1) and (VVH2) provide more accurate predictions than (Conv), (ConvIm) and (Rot), both for the velocity and vorticity. The scheme (Rot) based on the rotation form of the Navier–Stokes equations and without vorticity equation shows the worst results. This suggests that if one wishes to use the rotation form, then it is better to obtain vorticity dynamically from the vorticity equation rather than as a projection of the rotation of velocity. Alternatively, one may use the fully implicit in time schemes with the rotation form [21]. This however leads to solving a nonlinear problem on every time step, similar to (ConvIm).

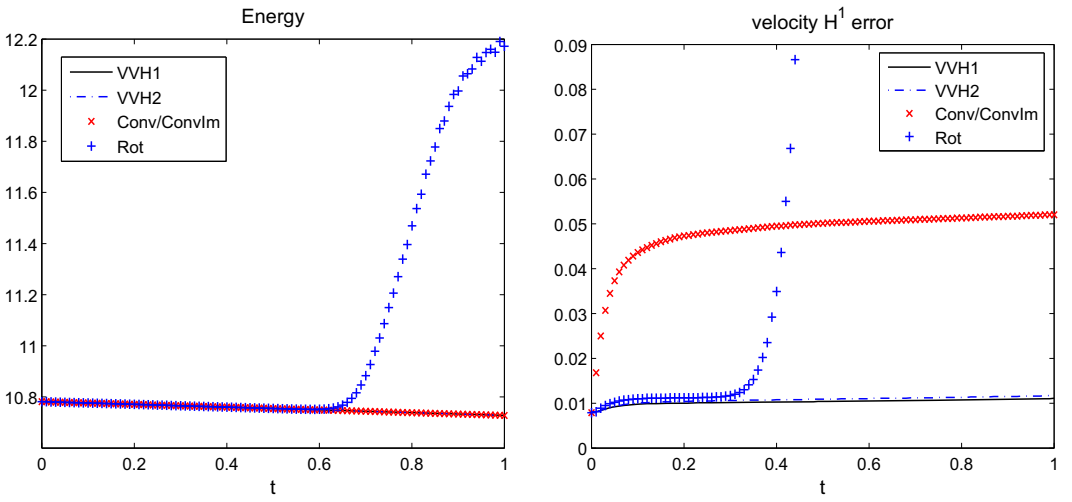

Fig. 4. LEFT: The kinematic energy balance for all five schemes for $\nu = 0.01$. RIGHT: The velocity H^1 errors versus time for all five schemes for $\nu = 0.01$.

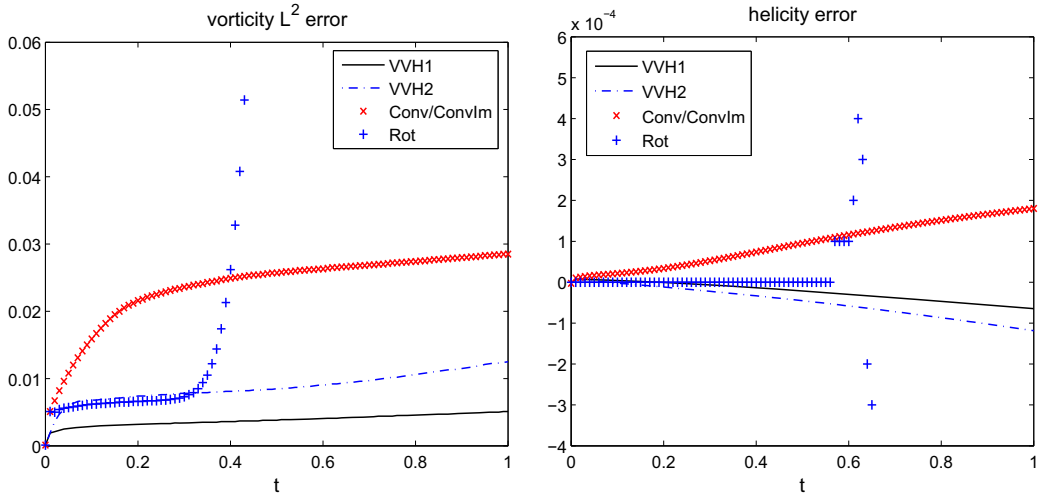

Fig. 5. LEFT: The vorticity L^2 errors versus time for all five schemes for $\nu = 0.01$. RIGHT: The helicity balance for all five schemes for $\nu = 0.01$.

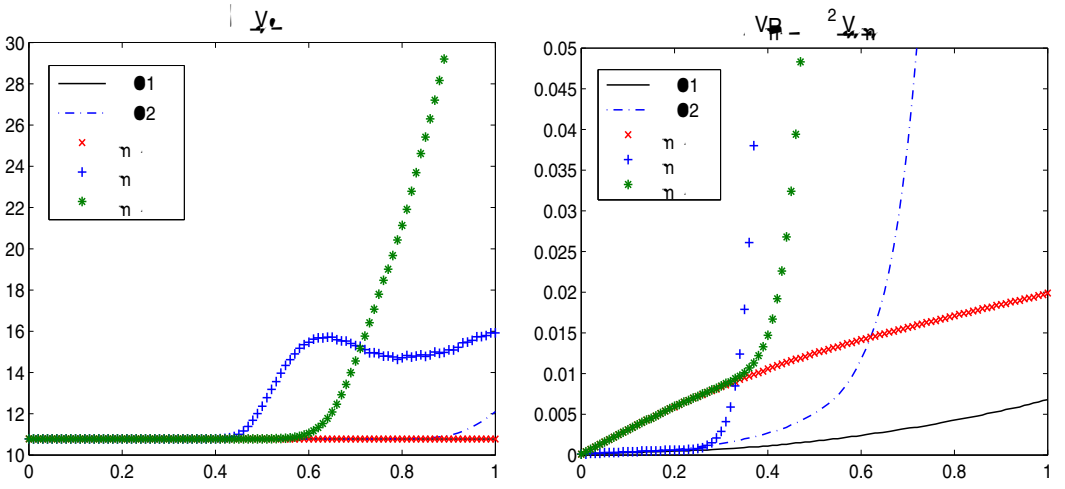

Fig. 6. LEFT: The kinematic energy balance for all five schemes for $\nu = 0.001$. RIGHT: The velocity L^2 errors versus time for all five schemes for $\nu = 0.001$.

Finally, Fig. 5 (right) shows the balance of helicity for all five schemes. The results shows that the (VVH1) scheme ensures the best balance of helicity.

Further, we repeat the experiment with higher Reynolds number by setting $\nu = 0.001$. The energy balance as well as velocity and vorticity errors versus time for $\nu = 0.001$ are plotted in Figs. 6 and 7. These results show (VVH1) is the most accurate in all plots. Interestingly, fully implicit (ConvIm) is more accurate in all plots than the linear (semi-implicit) schemes (Conv) and (VVH2), which gave equivalent and better results, respectively, than (ConvIm) in the $\nu = 0.01$ experiment. This is likely due to the greater inaccuracy of linearization at lower viscosity. Thus it is interesting that (VVH1) remains accurate here, even though it is also linear at every time step. In these tests (Rot) and (Conv) both give the largest errors in all plots. Note that (VVH2) is not becoming unstable; its error is bounded but these plots are "zoomed in" so that the difference between (VVH1) and (ConvIm) can be more easily seen. Fig. 8 compares helicity balance for all five schemes. From the left figure we see that schemes (VVH1) and (ConvIm) balance helicity much better than other three schemes. On the right figure we removed the helicity error plots for (Rot), (Conv) and (VVH2) allowing the closer comparison of (VVH1) and (ConvIm). The (VVH1) scheme shows the best result.

Comparing two VVH schemes, it follows from results that the more implicit in vorticity step scheme (VVH1) is more accurate especially for higher Reynolds numbers than the scheme (VVH2), which is more implicit in velocity step.

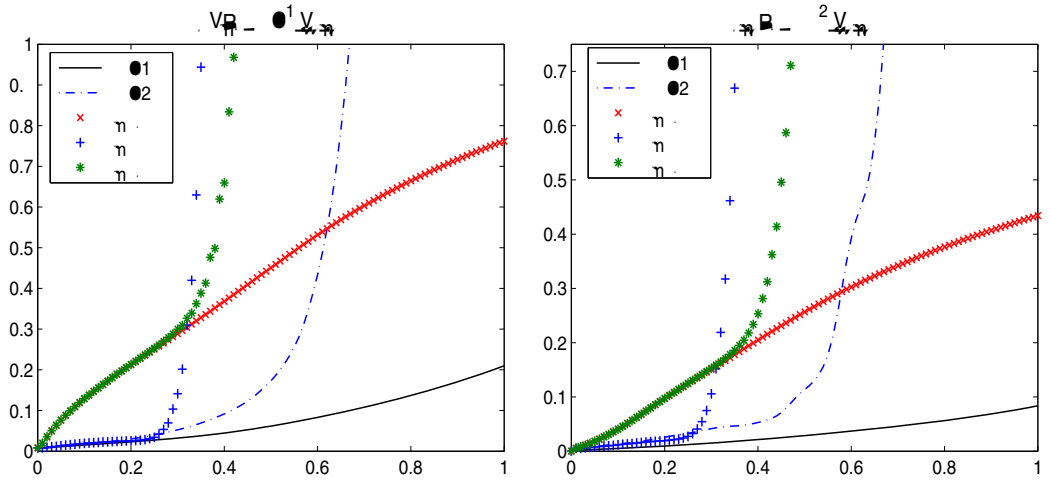

Fig. 7. LEFT: The velocity H^1 errors versus time for all five schemes for $\nu = 0.001$. RIGHT: The vorticity L^2 errors versus time for all five schemes for $\nu = 0.001$.

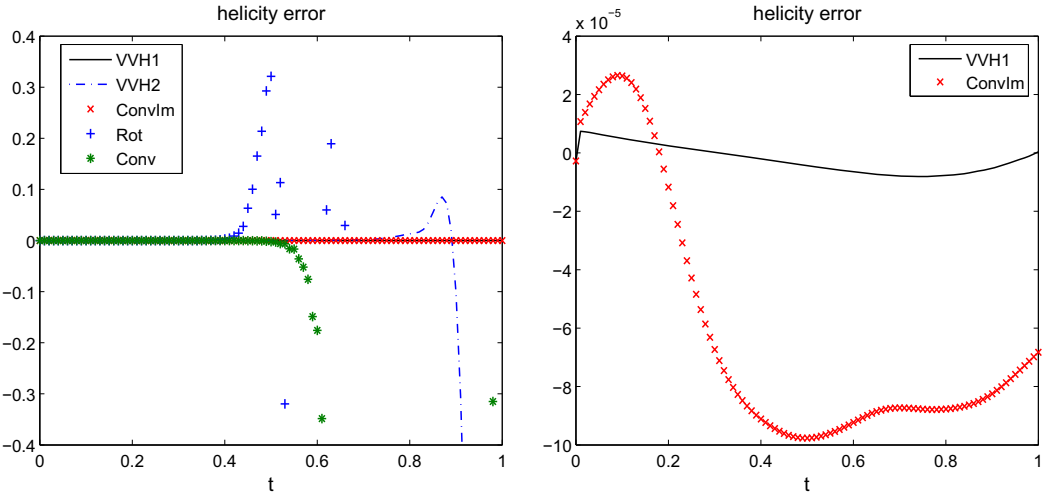

Fig. 8. LEFT: The helicity balance for all five schemes for $\nu = 0.001$. RIGHT: The helicity balance for (VVH1) and fully implicit primitive variable scheme (ConvIm) for $\nu = 0.001$.

5. Conclusions

We introduced a variant of the vorticity equation for the Navier–Stokes problem, which leads to the treatment of the helical density as an independent variable and relates it to the div-free constraint on the vorticity vector field. A natural choice of local boundary conditions for vorticity or vorticity and helical density makes the resulting formulation equivalent to the original problem. One possible application of the new formulation is the simple time-splitting numerical scheme based on an alternating procedure between vorticity–helical density and velocity–Bernoulli pressure systems of equations. Numerical results illustrate the viability of the new scheme and even its superiority over some other common approaches including the fully implicit second order scheme in primitive variables. These results allow us to suggest that the new formulation is theoretically and numerically sound and can be a useful tool for numerical or theoretical treatment of vortex dominated three-dimensional flows.

References

[1] J.C. Andre, M. Lesieur, Influence of helicity on the evolution of isotropic turbulence at high Reynolds number, J. Fluid Mech. 81 (1977) 187–207.

[2] Wm. T. Ashurst, A.R. Kerstein, R.M. Kerr, G.H. Gibson, Alignment of vorticity and scalar gradient with strain rate in simulated Navier–Stokes turbulence, Phys. Fluids 30 (1987) 2343–2353.
[3] H. Beirao da Veiga, L.C. Berselli, Navier–Stokes equations: Greens matrices, vorticity direction, and regularity up to the boundary, J. Differ. Equat. 246 (2009) 597–628.
[4] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numer. 14 (2005) 1–137.
[5] L.C. Berselli, D. Cordoba, On the regularity of the solutions to the 3D Navier–Stokes equations: a remark on the role of the helicity, Comptes Rendus Math. 347 (11–12) (2009) 613–618.
[6] P. Bochev, R. Lehoucq, On the finite element solution of the pure Neumann problem, SIAM Rev. 47 (1) (2005) 50–66.
[7] Q. Chen, S. Chen, G.L. Eyink, The joint cascade of energy and helicity in three-dimensional turbulence, Phys. Fluids 15 (2003) 361–374.
[8] C. Conca, C. Pares, O. Pironneau, M. Thiriet, Navier–Stokes equations with imposed pressure and velocity fluxes, Int. J. Numer. Methods Fluids 20 (1995) 267–287.
[9] C. Davies, P.W. Carpenter, A novel velocity–vorticity formulation of the Navier–Stokes equations with applications to boundary layer disturbance evolution, J. Comput. Phys. 172 (2001) 119–165.
[10] C. Doering, J. Gibbon, Applied Analysis of the Navier–Stokes Equations, Cambridge University Press, 1995.
[11] C. Ethier, D. Steinman, Exact fully 3D Navier–Stokes solutions for benchmarking, Int. J. Numer. Methods Fluids 19 (5) (1994) 369–375.
[12] H.C. Elman, D.J. Silvester, A.J. Wathen, Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics, in: Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, UK, 2005.
[13] C. Foias, L. Hoang, B. Nicolaenko, On the helicity in 3D-periodic Navier–Stokes equations I: the non-statistical case, Proc. London Math. Soc. 94 (2007) 53–90.
[14] L.P. Franca, S.L. Frey, Stabilized finite element methods. II. The incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 99 (1992) 209–233.
[15] T.B. Gatski, Review of incompressible fluid flow computations using the vorticity–velocity formulation, Appl. Numer. Math. 7 (1991) 227–239.
[16] R.L. Sani, P.M. Gresho, Resume and remarks on the open boundary condition minisymposium, Int. J. Numer. Methods Fluids 18 (1994) 983–1008.
[17] G. Guevremont, W.G. Habashi, M.M. Hafez, Finite element solution of the Navier–Stokes equations by a velocity–vorticity method, Int. J. Numer. Methods Fluids 10 (1990) 461–475.
[18] G. Guevremont, W.G. Habashi, Kotiuga, M.M. Hafez, Finite element solution of the 3D compressible Navier–Stokes equations by a velocity–vorticity method, J. Comput. Phys. 107 (1993) 176–187.
[19] J.G. Heywood, R. Rannacher, S. Turek, Artificial boundaries and flux and pressure conditions for the incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 22 (1996) 325–352.
[20] T.J.R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Comput. Methods Appl. Mech. Eng. 127 (1995) 387–401.
[21] W. Layton, C. Manica, M. Neda, M. Olshanskii, L. Rebholz, On the accuracy of the rotation form in simulations of the Navier–Stokes equations, J. Comput. Phys. 228 (9) (2009) 3433–3447.
[22] D.C. Lo, D.L. Young, K. Murugesan, An accurate numerical solution algorithm for 3D velocity–vorticity Navier–Stokes equations by the DQ method, Commun. Numer. Methods Eng. 22 (2006) 235–250.
[23] A. Majda, Vorticity and the mathematical theory of incompressible fluid flow, Commun. Pure Appl. Math. 39 (1986) 187–220.
[24] H.K. Moffatt, A. Tsinober, Helicity in laminar and turbulent flow, Ann. Rev. Fluid Mech. 24 (1992) 281–312.
[25] H.L. Meitz, H.F. Fasel, A compact-difference scheme for the Navier–Stokes equations in vorticity–velocity formulation, J. Comput. Phys. 157 (2000) 371–403.
[26] M.A. Olshanskii, On the Stokes problem with model boundary conditions, Sbornik: Math. 188 (1997) 603–620.
[27] M.A. Olshanskii, A low order Galerkin finite element method for the Navier–Stokes equations of steady incompressible flow: a stabilization issue and iterative methods, Comput. Methods Appl. Mech. Eng. 191 (2002) 5515–5536.
[28] M.A. Olshanskii, A. Reusken, Grad-Div stabilization for the Stokes equations, Math. Comput. 73 (2004) 1699–1718.
[29] V. Ruas, A new formulation of the three-dimensional velocity–vorticity system in viscous incompressible flow, ZAMM Z. Angew. Math. Mech. 79 (1999) 29–36.
[30] J. Trujillo, G.E. Karniadakis, A penalty method for the vorticity–velocity formulation, J. Comput. Phys. 149 (1999) 32–58.
[31] X.H. Wu, J.Z. Wu, J.M. Wu, Effective vorticity–velocity formulations for the three-dimensional incompressible viscous flows, J. Comput. Phys. 122 (1995) 68–82.
[32] K.L. Wong, A.J. Baker, A 3D incompressible Navier–Stokes velocity–vorticity weak form finite element algorithm, Int. J. Numer. Methods Fluids 38 (2002) 99–123.
[33] Z. Zunic, M. Hribersek, L. Skerget, J. Ravnik, 3D driven cavity flow by mixed boundary and finite element method, in: P. Wesseling, E. Onate, J. Periaux (Eds.), European Conference on Computational Fluid Dynamics, ECCOMAS CFD, 2006.