

Natural vorticity boundary conditions on solid walls

Maxim A. Olshanskii* Timo Heister† Leo G. Rebholz‡ Keith J. Galvin§

Abstract

We derive boundary conditions for the vorticity equation with solid wall boundaries. The formulation uses a Dirichlet condition for the normal component of vorticity, and Neumann type conditions for the tangential components. In a Galerkin (integral) formulation the tangential condition is natural, i.e. it is enforced by a right-hand side functional and does not impose a boundary constraint on trial and test spaces. The functional involves the pressure variable, and we discuss several velocity-vorticity formulations where the proposed condition is appropriate. Several numerical experiments are given that illustrate the validity of the approach.

1 Introduction

Fluid flow vorticity is an important dynamic variable and many phenomena can be described in terms of vorticity more readily than in terms of primitive variables. Vorticity plays a fundamental role in understanding the physics of laminar, transitional and turbulent flows [7, 36, 30], in mathematical analysis of fluid equations [25], and in computational fluid dynamics [10].

The vorticity dynamics of incompressible viscous fluid flows is driven by the system of equations

$$\frac{\partial \mathbf{w}}{\partial t} - \nu \Delta \mathbf{w} + (\mathbf{u} \cdot \nabla) \mathbf{w} - (\mathbf{w} \cdot \nabla) \mathbf{u} = \nabla \times \mathbf{f} \quad (1)$$

where \mathbf{u} is the fluid velocity in a non-inertial reference frame, $\mathbf{w} = \nabla \times \mathbf{u}$ is the flow vorticity, ν is the kinematic viscosity coefficient, and \mathbf{f} is a vector function of body forces per unit mass. To obtain a closed system, one should complement (1) with equations for \mathbf{u} and initial conditions, and if a flow problem is posed in domain with boundaries, then boundary conditions should be prescribed. Commonly, boundary conditions are given in terms of primal variables and stress tensor, rather than in terms of the vorticity. However, for analysis and for numerical methods based on vorticity equations, it is important to endow (1) with boundary conditions on \mathbf{w} .

Appropriate vorticity boundary conditions have been a subject of intensive discussion in the literature, especially, in the context of numerical methods for fluid equations. In

*Department of Mathematics, University of Houston, Houston, TX 77204 (molshan@math.uh.edu), partially supported by Army Research Office Grant 65294-MA.

†Department of Mathematical Sciences, Clemson University, Clemson, SC 29634 (heister@clemson.edu), partially supported by the Computational Infrastructure in Geodynamics initiative (CIG), through the National Science Foundation under Award No. EAR-0949446 and The University of California – Davis.

‡Department of Mathematical Sciences, Clemson University, Clemson, SC 29634 (rebholz@clemson.edu), partially supported by Army Research Office Grant 65294-MA.

§Department of Mathematical Sciences, Clemson University, Clemson, SC 29634 (kjalvi@clemson.edu), partially supported by NSF grant DMS1112593

section 2, we include a brief review of several main approaches. One should be especially careful with assigning vorticity boundary conditions on solid walls, i.e. those parts of the boundary where a fluid is assumed to have no-slip velocity, since these regions are responsible for vorticity production and give rise to physical and numerical boundary layers. An obvious choice of using the vorticity definition $\mathbf{w} = \nabla \times \mathbf{u}$ for the boundary condition on \mathbf{w} is not always optimal with respect to numerical accuracy. This motivated our search for an alternative way of prescribing boundary conditions on \mathbf{w} .

The main result of this paper is that on a no-slip boundary, an appropriate vorticity boundary condition is written in terms of a functional corresponding to a certain distribution along the boundary. In PDEs language, such boundary conditions are called *natural*. We derive this condition in section 3. Natural boundary conditions are easy to implement numerically, since they do not impose boundary constraints on trial and test spaces in a Galerkin method, and are less prone to produce numerical boundary layers. Our analysis determines that the functional depends on pressure distribution along the boundary. The critical role of tangential pressure gradients for boundary vorticity generation is known in the literature and discussed, e.g., in [27, 30]. However, this relationship has seemingly not been exploited for devising numerically efficient boundary conditions. In section 3 we also discuss what new insight the boundary conditions may give in a possible role of pressure and surface curvature in the vorticity production along solid boundaries.

The remainder of the paper is organized as follows. In section 2, we give necessary preliminaries and briefly review boundary conditions suggested in the literature to complement the vorticity equations (1). The natural vorticity boundary conditions are derived in section 3. Section 4 discusses several options to close the system of equations by combining (1) with the vector Poisson equations for velocity or the momentum equations with nonlinear terms driven by the Lamb vector. Section 5 presents results of several numerical experiments which demonstrate the utility and efficiency of the new vorticity boundary conditions for computing incompressible viscous flows. Section 6 collects a few closing remarks.

2 Problem setup and boundary conditions review

We consider the flow of an incompressible viscous Newtonian fluid in a bounded domain $\Omega \in \mathbb{R}^3$. In primitive (velocity-pressure) variables, the fluid motion is described by the Navier-Stokes equations

$$\left\{ \begin{array}{l} \frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \mathbf{f}, \\ \operatorname{div} \mathbf{u} = 0, \\ \mathbf{u}|_{t=0} = \mathbf{u}_0. \end{array} \right. \quad (2)$$

We distinguish between the upstream (inflow), downstream (outflow) and no-slip parts of the boundary, Γ_{in} , Γ_{out} , and Γ_w ($\partial\Omega = \Gamma_{in} \cup \Gamma_{out} \cup \Gamma_w$), by the type of boundary conditions imposed on them. On Γ_{in} we assume a prescribed velocity profile \mathbf{u}_{in} and an outflow boundary condition on Γ_{out} , e.g., the vanishing normal component of the stress tensor [17]. On the no-slip boundary Γ_w we have

$$\mathbf{u} = \mathbf{g} \quad \text{on } \Gamma_w, \quad \text{with } \mathbf{g} \cdot \mathbf{n} = 0, \quad (3)$$

where \mathbf{n} is an outward normal vector for Γ_w and $\mathbf{g}(\mathbf{x}, t)$ is a tangential velocity of the solid part of boundary. It is common to have $\mathbf{g} = \mathbf{0}$ for flows past a steady object or channel flow.

For the inflow, one may assume vorticity to be known and set $\mathbf{w} = \nabla \times \mathbf{u}_{in}$ on Γ_{in} , and letting the normal vorticity derivative vanish is a reasonable outflow boundary condition [13, 26]: $(\nabla \mathbf{w})\mathbf{n} = \mathbf{0}$ on Γ_{out} . The situation is more delicate with vorticity boundary conditions on Γ_w and several suggestions can be found in the literature. One common choice, see, e.g., [13, 24, 40, 42, 43, 44], is the kinematic condition

$$\mathbf{w} = \nabla \times \mathbf{u} \quad \text{on } \Gamma_w. \quad (4)$$

Note that opposite to inflow and outflow conditions, the vorticity boundary values on Γ_w depend on a generally unknown velocity field. For a computational treatment, this can be problematic, since numerical differentiation applied to a discrete velocity field on Γ_w may reduce the accuracy of the computed vorticity in the whole domain Ω , see [23]. Indeed, a numerical error introduced to vorticity values on Γ_w propagates into boundary layers and further may be convected in the interior of Ω . To mitigate the accuracy reduction and restrictions on \mathbf{w} caused by (4), some authors [19, 13], consider a weak numerical enforcement of (4), e.g., by using the Nitsche method.

A variant of (4), suggested in [35, 42, 45], uses only tangential part of the kinematic condition and enforces free divergence of \mathbf{w} on Γ_w :

$$\mathbf{w} \times \mathbf{n} = (\nabla \times \mathbf{u}) \times \mathbf{n}, \quad \text{div } \mathbf{w} = 0 \quad \text{on } \Gamma_w. \quad (5)$$

The explicit enforcement of the div-free constraint on \mathbf{w} along the boundary helps to ensure the vorticity to be solenoidal in Ω . However, it does not resolve accuracy issues related to (4).

Another class of vorticity boundary conditions are non-local conditions involving integral or integral-differential constraints, see, e.g. [1, 21, 34]. This approach uses the Biot-Savart formula to express the velocity from \mathbf{w} and further to find the vorticity diffusive flux on Γ_w solving an integral equation. Unlike this approach, we shall consider *local* differential boundary conditions on \mathbf{w} .

If the Navier-Stokes equations are written in terms of vorticity-stream (vector) function, then boundary conditions following from (4) are written in terms of the stream function rather than velocity, cf. [9].

3 Vorticity boundary conditions

In this section, we look for alternative boundary conditions to (4) on solid walls.

One vorticity condition easily follows from (3):

$$\mathbf{w} \cdot \mathbf{n} = (\nabla \times \mathbf{g}) \cdot \mathbf{n} \quad \text{on } \Gamma_w. \quad (6)$$

The right hand-side of (6) is well-defined, since $(\nabla \times \mathbf{u}) \cdot \mathbf{n}$ depends only on boundary values of \mathbf{u} , as is easy to see from the Kelvin-Stokes theorem (see also Proposition 3.1 below). Thus, $\mathbf{w} \cdot \mathbf{n}$ is defined only by given boundary values of \mathbf{u} , rather than by the unknown velocity in the fluid domain.

3.1 Neumann vorticity boundary conditions

We need two more vorticity boundary conditions on Γ_w . To deduce them, we rewrite the momentum equation as

$$\frac{\partial \mathbf{u}}{\partial t} + \nu \nabla \times \mathbf{w} + \mathbf{w} \times \mathbf{u} + \nabla P = \mathbf{f},$$

where $P = \frac{1}{2}|\mathbf{u}|^2 + p$ is the Bernoulli pressure, p is the kinematic pressure. Taking the tangential component on Γ_w and substituting (3) yields:

$$\begin{aligned}\nu(\nabla \times \mathbf{w}) \times \mathbf{n} &= (\mathbf{f} - \nabla P - \mathbf{w} \times \mathbf{g} - \mathbf{g}_t) \times \mathbf{n} \\ &= (\mathbf{f} - \nabla P - \mathbf{g}_t) \times \mathbf{n} - \mathbf{g}((\nabla \times \mathbf{g}) \cdot \mathbf{n}) \quad \text{on } \Gamma_w.\end{aligned}\tag{7}$$

Here we used the identity: $(\mathbf{w} \times \mathbf{g}) \times \mathbf{n} = \mathbf{g}(\mathbf{w} \cdot \mathbf{n}) - \mathbf{w}(\mathbf{g} \cdot \mathbf{n}) = \mathbf{g}(\mathbf{w} \cdot \mathbf{n})$ and (6). We note that the terms on the right hand side of (7) depend only on boundary values of g and p . The relation (7) gives two more boundary conditions on \mathbf{w} . Since $(\nabla|\mathbf{u}|^2) \times \mathbf{n}$ depends only on the boundary values of \mathbf{u} , the right-hand side in (7) can be also rewritten using the kinematic pressure through

$$(\nabla P) \times \mathbf{n} = (\nabla p) \times \mathbf{n} + \frac{1}{2}(\nabla|\mathbf{g}|^2) \times \mathbf{n}.$$

To simplify the notations, we use the identity

$$\mathbf{g}((\nabla \times \mathbf{g}) \cdot \mathbf{n}) = ((\nabla \times \mathbf{g}) \times \mathbf{g}) \times \mathbf{n},$$

for \mathbf{g} satisfying $\mathbf{g} \cdot \mathbf{n} = 0$, and denote $\tilde{\mathbf{g}} = \mathbf{f} - \mathbf{g}_t - (\nabla \times \mathbf{g}) \times \mathbf{g} - \frac{1}{2}(\nabla|\mathbf{g}|^2)$ on Γ_w . The *Neumann vorticity boundary conditions* now take more compact form:

$$\nu(\nabla \times \mathbf{w}) \times \mathbf{n} = (\tilde{\mathbf{g}} - \nabla p) \times \mathbf{n} \quad \text{on } \Gamma_w.\tag{8}$$

We obtained Neumann type vorticity boundary conditions (8). The right hand side in (8) depends on the gradient of the pressure variable. For $\mathbf{f} = 0$ and $\mathbf{g} = 0$, the conditions (8) are discussed in [30] as possessing rich physical and mathematical information relevant to vorticity dynamics. In particular, it is noted that the dynamic boundary conditions for the vorticity and pressure gradient naturally match the acceleration adherence, which makes the boundary conditions and compatibility condition merge into one. This is argued to reflect the correct physics, since the acceleration adherence determines the vorticity creation from the boundary.

For the further understanding of conditions (8) assume that the boundary Γ_w is static, there are no external forces and so $\tilde{\mathbf{g}} = 0$. Recall that \mathbf{n} is the normal vector pointing from Γ_w outward into the fluid domain, hence the vector $\frac{\partial \mathbf{w}}{\partial \mathbf{n}}$ is the boundary vorticity flux, i.e. $-\frac{\partial \mathbf{w}}{\partial \mathbf{n}}$ can be considered as the *vorticity production* on the solid boundary. Note the identity

$$(\nabla \times \mathbf{w}) \times \mathbf{n} = [(\nabla \mathbf{w}) - (\nabla \mathbf{w})^T] \mathbf{n} = \frac{\partial \mathbf{w}}{\partial \mathbf{n}} - (\nabla \mathbf{w})^T \mathbf{n}.\tag{9}$$

Denote by \mathbf{H} the Weingarten map or the shape operator for the surface Γ_w . We need the following properties of $\mathbf{H}(\mathbf{x})$, $\mathbf{x} \in \Gamma_w$: \mathbf{H} is a symmetric 3×3 tensor on Γ_w , eigenvalues of \mathbf{H} are $\{0, \kappa_1, \kappa_2\}$, where κ_1, κ_2 are the *principle curvatures* of Γ_w and the eigenvectors are the corresponding principal directions in which the surface bends at each point. It also holds $\mathbf{H} = \nabla_{\Gamma} \mathbf{n}$ for the surface gradient ∇_{Γ} . With the help of \mathbf{H} and recalling $\mathbf{w} \cdot \mathbf{n} = 0$ on Γ , we can rewrite the last term in (9) as

$$(\nabla \mathbf{w})^T \mathbf{n} = \nabla(\mathbf{w} \cdot \mathbf{n}) - (\nabla \mathbf{n}) \mathbf{w} = \frac{\partial(\mathbf{w} \cdot \mathbf{n})}{\partial \mathbf{n}} \mathbf{n} - \mathbf{H} \mathbf{w}.$$

Substituting this into (9) and (8), gives the expression for the production of the streamwise and spanwise vorticity on the solid wall boundary:

$$-\left(\frac{\partial \mathbf{w}}{\partial \mathbf{n}} - \frac{\partial(\mathbf{w} \cdot \mathbf{n})}{\partial \mathbf{n}} \mathbf{n}\right) = -\nu^{-1} \mathbf{n} \times \nabla p + \mathbf{H} \mathbf{w} \quad \text{on } \Gamma_w.\tag{10}$$

On left hand side of (10) one has the total vorticity production on the boundary minus the flux into the normal vorticity component. Hence the left hand side corresponds to the boundary flux for the longitudinal and latitudinal vorticity components. The right hand side indicates that the vorticity production depends on the *variation of pressure* along the boundary and on the *shape* of the boundary. It is well-known in the literature [27] that the tangential pressure gradients plays an important role for the generation vorticity on a boundary. Note that for convex (as viewed from the fluid domain) boundaries, the bilinear form $\langle \mathbf{H}, \cdot \rangle$ is negative definite on planes tangent to Γ_w , while it is positive definite for concave boundaries. Thus, (10) explicitly shows that the vorticity production is suppressed along convex boundaries and intensified along concave boundaries. In particular, this may give a formal mathematical explanation to the well-known effects of convex (stabilizing) and concave (destabilizing) shape on turbulent boundary layers [18, 28], and longitudinal vorticity production along concave walls [41]. We shall pursue investigating implications of (10) on the vorticity generation and boundary layer properties elsewhere. Summarizing, conditions (8) possess rich physical, mathematical and geometrical information relevant to vorticity dynamics, and in this paper we concentrate on employing them for efficient numerical treatment of incompressible viscous flows.

In previous studies, the boundary $\mathbf{w} - p$ coupling was deemed to cause the basic difficulty of vorticity formulations and an effort was made to bypass this coupling numerically at the expense of stability/accuracy restrictions in terms of time step and Reynolds numbers. A different point of view and numerical approach is taken in the present article. Several observations play further a key role and contrast conditions (8) to (4) or (5) and our approach to decoupling strategies:

- First, Neumann boundary conditions are natural in an integral formulation of the Navier-Stokes equations and so in any Galerkin method, i.e. the conditions are accounted for by a right-hand side functional and do not enter the definitions of trial or test spaces.
- Second, $(\nabla p) \times \mathbf{n}$ depends only on boundary values of p and one can apply integration by parts over Γ_w to avoid computing the pressure gradient.
- Finally, the vorticity equations can be complemented with velocity-pressure equations driven by the Lamb vector, which gives the consistent and numerically efficient coupling and provides the pressure for the boundary functional.

We explain these observations below in more detail.

From the implementation viewpoint, one may consider involving the pressure gradient along the boundary as a disadvantage of (8). Below we show that (8) can be efficiently implemented and computing the pressure derivatives is avoided.

3.2 Vorticity boundary conditions as a functional

Assume \mathbf{u} and p are given and consider the integral formulation of the vorticity equation (1) subject to conditions (6), (8) as well as inflow and outflow boundary conditions : Find \mathbf{w} , satisfying $\mathbf{w} \cdot \mathbf{n} = (\nabla \times \mathbf{g}) \cdot \mathbf{n}$ on Γ_w , $\mathbf{w} = \nabla \times \mathbf{u}_{\text{in}}$ on Γ_{in} and

$$\begin{aligned} \int_{\Omega} \left(\frac{\partial \mathbf{w}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{w} - (\mathbf{w} \cdot \nabla) \mathbf{u} \right) \cdot \mathbf{v} + \nu (\nabla \times \mathbf{w}) \cdot (\nabla \times \mathbf{v}) + \nu \operatorname{div} \mathbf{w} \operatorname{div} \mathbf{v} \, d\mathbf{x} \\ = \int_{\Omega} (\nabla \times \mathbf{f}) \cdot \mathbf{v} \, d\mathbf{x} + \int_{\Gamma_w} ((\tilde{\mathbf{g}} - \nabla p) \times \mathbf{n}) \cdot \mathbf{v} \, ds \quad \text{for all } t > 0, \end{aligned} \quad (11)$$

for any smooth \mathbf{v} such that $\mathbf{v} \cdot \mathbf{n} = 0$ on Γ_w and $\mathbf{v} = 0$ on Γ_{in} . The outflow conditions we used in (11) read:

$$(\nabla \times \mathbf{w}) \times \mathbf{n} = 0 \quad \text{and} \quad \text{div } \mathbf{w} = 0 \quad \text{on } \Gamma_{\text{out}}.$$

Other outflow boundary conditions are possible, but would lead to extra boundary integral terms in (11).

To avoid computing pressure gradient over Γ_w , we shall rewrite the last term in (11) using integration by parts on Γ_w . For a smooth surface Γ , recall the definition of the surface gradient and divergence:

$$\nabla_{\Gamma} p = \nabla p - (\mathbf{n} \cdot \nabla p) \mathbf{n} \quad \text{and} \quad \text{div}_{\Gamma} \mathbf{v} = \text{tr}(\nabla_{\Gamma} \mathbf{v})$$

which are the intrinsic surface quantities and do not depend on extensions of a scalar function p and a vector function \mathbf{v} off a surface, see, e.g., [12]. Assume that Γ is sufficiently smooth and has a boundary $\partial\Gamma$ whose intrinsic unit outer normal (conormal) is denoted by μ . We will need the following identity for integration by parts over Γ (see [8, 11] for more details):

$$\int_{\Gamma} (\nabla_{\Gamma})_i p \, ds = \int_{\Gamma} \kappa p n_i \, ds + \int_{\partial\Gamma} p \mu_i \, dl$$

where κ is the surface mean curvature ($\kappa = \text{div}_{\Gamma} \mathbf{n}$). This leads to the following identity:

$$\int_{\Gamma} p \, \text{div}_{\Gamma} \mathbf{v} + \mathbf{v} \cdot \nabla_{\Gamma} p \, ds = \int_{\Gamma} \kappa (\mathbf{v} \cdot \mathbf{n}) p \, ds + \int_{\partial\Gamma} p \mathbf{v} \cdot (d\mathbf{l} \times \mathbf{n}). \quad (12)$$

The definition of the surface gradient immediately yields the identity:

$$(\nabla p) \times \mathbf{n} = (\nabla_{\Gamma} p) \times \mathbf{n}.$$

Hence assuming Γ_w is smooth and thanks to (12) we manipulate with the pressure term from (11):

$$\begin{aligned} \int_{\Gamma_w} ((\nabla p) \times \mathbf{n}) \cdot \mathbf{v} \, ds &= \int_{\Gamma_w} ((\nabla_{\Gamma} p) \times \mathbf{n}) \cdot \mathbf{v} \, ds = - \int_{\Gamma_w} (\mathbf{v} \times \mathbf{n}) \cdot \nabla_{\Gamma} p \, ds \\ &= \int_{\Gamma_w} \text{div}_{\Gamma} (\mathbf{v} \times \mathbf{n}) p \, ds - \int_{\partial\Gamma_w} p (\mathbf{v} \times \mathbf{n}) \cdot (d\mathbf{l} \times \mathbf{n}). \end{aligned} \quad (13)$$

For the last term in (13), one gets using $\mathbf{v} \cdot \mathbf{n} = 0$, $|\mathbf{n}|^2 = 1$ and vector identities

$$(\mathbf{v} \times \mathbf{n}) \cdot (d\mathbf{l} \times \mathbf{n}) = \mathbf{v} \cdot d\mathbf{l}.$$

This identity shows that (13) is valid also for piecewise smooth surfaces Γ_w . Indeed, the formula (12) can be applied on each smooth part of Γ_w and the contour integrals over shared boundaries of these parts cancel out.

Finally, one can also rewrite the first term on the right-hand side of (13) using the following simple result.

Proposition 3.1. *Assume \mathbf{v} is a vector field defined in a \mathbb{R}^3 neighborhood of a smooth surface Γ and $\mathbf{v} \cdot \mathbf{n} = 0$, then it holds*

$$\text{div}_{\Gamma} (\mathbf{v} \times \mathbf{n}) = (\nabla \times \mathbf{v}) \cdot \mathbf{n} \quad \text{on } \Gamma. \quad (14)$$

In particular, (14) implies that $(\nabla \times \mathbf{v}) \cdot \mathbf{n}$ depends only on boundary values of \mathbf{v} .

Proof. Fix any $\mathbf{p} \in \Gamma$ and consider a sufficiently small neighborhood $\mathcal{O}(\mathbf{p}) \subset \mathbb{R}^3$ of \mathbf{p} . Let ϕ be a signed distance function of Γ in $\mathcal{O}(\mathbf{p})$. Then $\mathbf{n} = \nabla\phi$ is defined in $\mathcal{O}(\mathbf{p})$ and coincides with the normal vector on Γ . Denote by $\mathbf{H} = \nabla^2\phi$ the Hessian of ϕ and $\mathbf{P} = I - \mathbf{nn}^T$ the normal projector on level sets of ϕ . Then it holds (cf, e.g., [12]):

$$\nabla_\Gamma = \mathbf{P}\nabla \quad \text{on } \Gamma \quad \text{and} \quad \mathbf{P}\mathbf{H} = \mathbf{H}\mathbf{P} = \mathbf{H} \quad \text{in } \mathcal{O}(\mathbf{p}). \quad (15)$$

For a vector \mathbf{n} , denote by $[\mathbf{n}]_\times$ a 3×3 skew-symmetric matrix, such that $[\mathbf{n}]_\times \mathbf{a} = \mathbf{n} \times \mathbf{a}$ for any $\mathbf{a} \in \mathbb{R}^3$. Using this formalism, we compute

$$\operatorname{div}_\Gamma(\mathbf{v} \times \mathbf{n}) = \operatorname{tr}(\mathbf{P}\nabla(\mathbf{v} \times \mathbf{n})) = -\operatorname{tr}(\mathbf{P}[\mathbf{n}]_\times(\nabla\mathbf{v})) + \operatorname{tr}(\mathbf{P}[\mathbf{v}]_\times\mathbf{H}). \quad (16)$$

Using the elementary properties of the trace operation and (15) we get:

$$\operatorname{tr}(\mathbf{P}[\mathbf{v}]_\times\mathbf{H}) = \operatorname{tr}([\mathbf{v}]_\times\mathbf{H}\mathbf{P}) = \operatorname{tr}([\mathbf{v}]_\times\mathbf{H}) = 0.$$

The last equality holds since $\mathbf{H} = \mathbf{H}^T$ and $[\mathbf{v}]_\times = -[\mathbf{v}]_\times^T$. Further, note that $\mathbf{P}[\mathbf{n}]_\times = [\mathbf{n}]_\times$. For the first term on the right-hand side of (16), we have

$$-\operatorname{tr}(\mathbf{P}[\mathbf{n}]_\times(\nabla\mathbf{v})) = -\operatorname{tr}([\mathbf{n}]_\times(\nabla\mathbf{v})) = \sum_{i,j=1}^3 ([\mathbf{n}]_\times)_{i,j} \frac{\partial v_i}{\partial x_j} = (\nabla \times \mathbf{v}) \cdot \mathbf{n}.$$

■

Summarizing, *the integral formulation of the vorticity equations* reads: Find \mathbf{w} , satisfying $\mathbf{w} \cdot \mathbf{n} = (\nabla \times \mathbf{g}) \cdot \mathbf{n}$ on Γ_w , $\mathbf{w} = \nabla \times \mathbf{u}_{\text{in}}$ on Γ_{in} and

$$\begin{aligned} & \int_{\Omega} \left(\frac{\partial \mathbf{w}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{w} - (\mathbf{w} \cdot \nabla) \mathbf{u} \right) \cdot \mathbf{v} + \nu (\nabla \times \mathbf{w}) \cdot (\nabla \times \mathbf{v}) + \nu \operatorname{div} \mathbf{w} \operatorname{div} \mathbf{v} \, d\mathbf{x} \\ & = \int_{\Omega} (\nabla \times \mathbf{f}) \cdot \mathbf{v} \, d\mathbf{x} + \int_{\Gamma_w} (\tilde{\mathbf{g}} \times \mathbf{n}) \cdot \mathbf{v} \, ds - \int_{\Gamma_w} p (\nabla \times \mathbf{v}) \cdot \mathbf{n} \, ds + \int_{\partial\Gamma_w} p \mathbf{v} \cdot d\mathbf{l} \quad \forall t > 0, \end{aligned} \quad (17)$$

for any smooth \mathbf{v} such that $\mathbf{v} \cdot \mathbf{n} = 0$ on Γ_w and $\mathbf{v} = 0$ on Γ_{in} .

The Neumann vorticity boundary conditions (8) are accounted by the surface and contour integrals on the right-hand side of (17) and impose no restrictions on a functional space, where vorticity is sought. If the pressure distribution is known along Γ_w , then these integrals are functionals defined for test functions \mathbf{v} and this constitutes the mathematically sound problem formulation. Although we avoid computing pressure derivatives now, one might consider bringing back pressure variable into the velocity-vorticity system of equations as a potential downside of the natural vorticity boundary conditions. Thus, the next section discusses several ways of closing the system such that the pressure variable is naturally recovered.

4 Velocity-vorticity coupling

There are several ways to couple equations for velocity and pressure to the vorticity equation. In this section we discuss three such couplings. We start with recalling the most commonly found in the literature, which is based on the velocity and pressure Poisson equations.

4.1 Velocity and pressure Poisson equations

Typical coupling of velocity to the vorticity equation is done via the relation

$$-\Delta \mathbf{u} = \nabla \times \mathbf{w}. \quad (18)$$

These relations do not involve pressure, and so they are not applicable directly for the proposed boundary condition. However, taking divergence of the Navier-Stokes momentum equation provides

$$-\Delta p = \operatorname{div}(\mathbf{u} \cdot \nabla \mathbf{u} - \mathbf{f}). \quad (19)$$

for kinematic pressure or

$$-\Delta P = \operatorname{div}(\mathbf{w} \times \mathbf{u} - \mathbf{f}), \quad (20)$$

for the Bernoulli pressure.

If (20) or (19) is equipped with appropriate boundary conditions, then pressure could easily be incorporated into a time stepping scheme for a velocity-vorticity system of the form (1), (18).

4.2 Lamb vector and Bernoulli pressure

Another way is to couple vorticity equations to the momentum equations written in the rotations form. This gives the coupled system

$$\left\{ \begin{array}{l} \frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \mathbf{w} \times \mathbf{u} + \nabla P = \mathbf{f}, \\ \operatorname{div} \mathbf{u} = 0, \\ \frac{\partial \mathbf{w}}{\partial t} - \nu \Delta \mathbf{w} + (\mathbf{u} \cdot \nabla) \mathbf{w} - (\mathbf{w} \cdot \nabla) \mathbf{u} = \nabla \times \mathbf{f}. \end{array} \right. \quad (21)$$

In this system, the Bernoulli pressure is unknown variable and the velocity dynamics is mainly driven by the Lamb vector $\mathbf{w} \times \mathbf{u}$, which gives the strong coupling of the vorticity and the velocity.

It is discussed in [31] how this system can be decoupled in an energy stable way in a time-stepping scheme, by extrapolating vorticity in time in the momentum equation (which linearizes it), then using the velocity solution in the (now linear) vorticity equation. Furthermore, [15] proves that for 2D flows the numerical method's discrete vorticity and velocity are both long-time stable in the L^2 and H^1 norms, without any timestep restriction.

4.3 Lamb vector, rate of deformation tensor and kinematic pressure

For certain problems, the use of Bernoulli pressure may lead to increased numerical error in finite element methods where mass conservation is not strongly enforced [22]. If one is interested in computing lift and drag around an object, solving for Bernoulli pressure can lead to worse accuracy, since a recovered kinematic pressure may be less accurate than directly computed.

In such cases, it may be advantageous to use a variant of (21) that utilizes the vector identity

$$D(\mathbf{u})\mathbf{v} = \mathbf{v} \cdot \nabla \mathbf{u} - \frac{1}{2}(\nabla \times \mathbf{u}) \times \mathbf{v}, \quad (22)$$

where $D(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$ represents the rate of deformation tensor. Relation (22) enables the velocity – vorticity coupling by including the Lamb vector in the momentum

equations without altering the kinematic pressure. To our knowledge, the identity (22) was first pointed out in [4] for use in the Navier-Stokes equations. This leads to the coupled system

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \frac{1}{2} \mathbf{w} \times \mathbf{u} + D(\mathbf{u})\mathbf{u} + \nabla p = \mathbf{f}, \\ \operatorname{div} \mathbf{u} = 0, \\ \frac{\partial \mathbf{w}}{\partial t} - \nu \Delta \mathbf{w} + (\mathbf{u} \cdot \nabla) \mathbf{w} - (\mathbf{w} \cdot \nabla) \mathbf{u} = \nabla \times \mathbf{f}. \end{cases} \quad (23)$$

Similar to (21) the system (23) can be also integrated numerically using time splitting techniques as shown in the next section.

Given these three coupling strategies, the authors of this paper give some preference to those in (21) and (23), since these formulations do not require pressure boundary conditions, directly lead to energy stable finite element discretizations, cf. [23], and admit natural and simple time-stepping strategies. In the numerical examples section, we will test finite element algorithms based on (21) and (23).

5 Numerical examples

We provide four numerical experiments in this section, with the goals of 1) testing the feasibility of the proposed natural boundary condition for vorticity, and 2) testing the accuracy of the velocity-vorticity schemes that use the natural vorticity boundary conditions. Our results all indicate that both the proposed boundary condition and the schemes perform very well, and at least as good as often better than related schemes that use only velocity-pressure variables.

The first test is a steady flow with known analytical solution, and we calculate convergence rates of a method that uses the proposed boundary conditions; optimal convergence rates are found. The second test is for channel flow past a normal flat plate at $\operatorname{Re}=100$ and $\operatorname{Re}=150$, and we find the proposed method to work significantly better than an analogous method that uses velocity-pressure variables only. For the third test, we consider is time dependent flow around a cylinder, and we compare the solutions of schemes for (21) and (23). Here, we find for more accurate lift and drag predictions, it is better to use (23) since it uses the usual pressure instead of the Bernoulli pressure. Our final test is for 3D steady flow around a square cylinder at $\operatorname{Re}=20$. Because of the singular solution behavior near the sharp edges of the cylinder, getting accurate lift and drag is very challenging, especially if a numerical approach involves higher order flow dynamics variables, such as vorticity. Hence, this test requires locally adapted fine meshes. We compare solutions of the proposed (steady) velocity-vorticity method (23) to the standard scheme in velocity-pressure variables on meshes up to 5 million dof, and find the proposed method works well, giving a significantly better lift prediction and slightly worse drag prediction.

5.1 Finite element algorithms

For the two tests of steady flows, the velocity boundary condition is given as Dirichlet on the entire boundary, so we take $\Gamma = \Gamma_w$ and apply the proposed vorticity boundary condition on the entire boundary. Denoting by τ_h a mesh of Ω , we define the space

$$\mathbf{W}_h^g := \{\mathbf{w}_h \in \mathbf{H}^1(\Omega) \cap \mathbf{P}_k(\tau_h), \mathbf{w}_h \cdot \mathbf{n}|_\Gamma = (\nabla \times \mathbf{g}) \cdot \mathbf{n}\}.$$

In particular, the test space is $\mathbf{W}_h^0 := \mathbf{W}_h^g$ for $g = 0$. Next, we compute using the following finite element method: For a given $\mathbf{u} \in \mathbf{H}^1(\Omega)$, $p \in L^2(\Gamma)$, and $\mathbf{f} \in \mathbf{L}^2(\Omega)$, set $\tilde{\mathbf{g}} = \mathbf{f} - (\nabla \times \mathbf{g}) \times \mathbf{g}$, and find $\mathbf{w}_h \in \mathbf{W}_h^g$, satisfying

$$\begin{aligned} & ((\mathbf{u} \cdot \nabla) \mathbf{w}_h, \mathbf{v}_h) - ((\mathbf{w}_h \cdot \nabla) \mathbf{u}, \mathbf{v}_h) + \nu(\nabla \times \mathbf{w}_h, \nabla \times \mathbf{v}_h) + \nu(\operatorname{div} \mathbf{w}_h, \operatorname{div} \mathbf{v}_h) \\ &= (\nabla \times \mathbf{f}, \mathbf{v}_h) + \int_{\Gamma} (\tilde{\mathbf{g}} \times \mathbf{n}) \cdot \mathbf{v}_h \, ds - \int_{\Gamma} p(\nabla \times \mathbf{v}_h) \cdot \mathbf{n} \, ds + \int_{\Gamma} p \mathbf{v}_h \cdot d\mathbf{l} \quad \forall \mathbf{v}_h \in \mathbf{W}_h^0. \end{aligned} \quad (24)$$

For the time dependent problem, there will be an inflow and outflow which will be given as analytic functions for velocity ($\mathbf{u} = \mathbf{g}$ on Γ_{in} and Γ_{out}). Hence we will take the vorticity as fully Dirichlet on these boundaries, and enforce $\mathbf{w} = \nabla \times \mathbf{g}$ on Γ_{in} and Γ_{out} .

We test finite element implementation of both (21) and (23), which use a natural splitting in the time stepping and thus decouple the velocity-pressure system from the vorticity system by extrapolating the vorticity through previous timesteps. Hence for this problem we define the spaces as

$$\begin{aligned} \mathbf{V}_h^g &:= \{ \mathbf{v}_h \in \mathbf{H}^1(\Omega) \cap \mathbf{P}_k(\tau_h), \mathbf{v}_h|_{\Gamma} = \mathbf{g} \}, \quad \mathbf{V}_h^0 = \mathbf{V}_h^g \\ \mathbf{W}_h^g &:= \{ \mathbf{w}_h \in \mathbf{H}^1(\Omega) \cap \mathbf{P}_k(\tau_h), \mathbf{w}_h \cdot \mathbf{n}|_{\Gamma_w} = (\nabla \times \mathbf{g}) \cdot \mathbf{n}, \mathbf{w}_h|_{\Gamma_{in,out}} = \nabla \times \mathbf{g} \}. \end{aligned}$$

Denoting

$$\phi^{n+\frac{1}{2}} := \frac{\phi^n + \phi^{n+1}}{2},$$

the linearized Crank-Nicolson, finite element discretization of (21) is then (at each timestep):

Step 1: Find $(\mathbf{u}_h^{n+1}, P_h^{n+\frac{1}{2}}) \in (\mathbf{V}_h^g, Q_h)$ satisfying

$$\begin{aligned} & \frac{1}{\Delta t} (\mathbf{u}_h^{n+1} - \mathbf{u}_h^n, \mathbf{v}_h) + \left(\left(\frac{3}{2} \mathbf{w}_h^n - \frac{1}{2} \mathbf{w}_h^{n-1} \right) \times \mathbf{u}_h^{n+\frac{1}{2}}, \mathbf{v}_h \right) \\ & \quad - (P_h^{n+\frac{1}{2}}, \nabla \cdot \mathbf{v}_h) + \nu(\nabla \mathbf{u}_h^{n+1/2}, \nabla \mathbf{v}_h) = 0 \quad \forall \mathbf{v}_h \in \mathbf{V}_h^0 \\ & \quad (\nabla \cdot \mathbf{u}_h^{n+1}, q_h) = 0 \quad \forall q_h \in Q_h. \end{aligned}$$

Step 2: Find $\mathbf{w}_h^{n+1} \in \mathbf{W}_h^g$ satisfying

$$\begin{aligned} & \frac{1}{\Delta t} (\mathbf{w}_h^{n+1} - \mathbf{w}_h^n, \boldsymbol{\chi}_h) + (\mathbf{u}_h^{n+\frac{1}{2}} \cdot \nabla \mathbf{w}_h^{n+\frac{1}{2}}, \boldsymbol{\chi}_h) - (\mathbf{w}_h^{n+\frac{1}{2}} \cdot \nabla \mathbf{u}_h^{n+\frac{1}{2}}, \boldsymbol{\chi}_h) \\ & \quad + \nu(\nabla \cdot \mathbf{w}_h^{n+\frac{1}{2}}, \nabla \cdot \boldsymbol{\chi}_h) + \nu(\nabla \times \mathbf{w}_h^{n+\frac{1}{2}}, \nabla \times \boldsymbol{\chi}_h) = ((\nabla \times \mathbf{f}(t^{n+\frac{1}{2}})), \boldsymbol{\chi}_h) \\ & \quad + \int_{\Gamma_w} (\tilde{\mathbf{g}}(t^{n+\frac{1}{2}}) \times \mathbf{n}) \cdot \boldsymbol{\chi}_h \, ds - \int_{\Gamma_w} P_h^{n+\frac{1}{2}} (\nabla \times \boldsymbol{\chi}_h) \cdot \mathbf{n} \, ds + \int_{\partial \Gamma_w} P_h^{n+\frac{1}{2}} \boldsymbol{\chi}_h \cdot d\mathbf{l} \quad \forall \boldsymbol{\chi}_h \in \mathbf{W}_h^0. \end{aligned} \quad (25)$$

Since the right-hand side of (25) contains the Bernoulli pressure, the definition of $\tilde{\mathbf{g}}$ is adjusted here to $\tilde{\mathbf{g}} = \mathbf{f} - \mathbf{g}_t - (\nabla \times \mathbf{g}) \times \mathbf{g}$ on Γ_w .

We will also test a finite element algorithm for the coupled system (23). This scheme splits the system in a similar way as for (21). The Step 1 is replaced by

Alternate Step 1: Find $(\mathbf{u}_h, p_h) \in (\mathbf{V}_h, Q_h)$ satisfying

$$\begin{aligned} & \frac{1}{\Delta t} (\mathbf{u}_h^{n+1} - \mathbf{u}_h^n, \mathbf{v}_h) + \frac{1}{2} (\mathbf{w}_h^{n+\frac{1}{2}} \times \mathbf{u}_h^{n+\frac{1}{2}}, \mathbf{v}_h) + (D(\mathbf{u}_h^{n+\frac{1}{2}}) \mathbf{u}_h^{n+\frac{1}{2}}, \mathbf{v}_h) \\ & \quad - (p_h^{n+\frac{1}{2}}, \nabla \cdot \mathbf{v}_h) + \nu(\nabla \mathbf{u}_h^{n+1/2}, \nabla \mathbf{v}_h) = (\mathbf{f}^{n+\frac{1}{2}}, \mathbf{v}_h) \quad \forall \mathbf{v}_h \in \mathbf{V}_h^0 \\ & \quad (\nabla \cdot \mathbf{u}_h^{n+1}, q_h) = 0 \quad \forall q_h \in Q_h. \end{aligned}$$

Alternate Step 2 is the same as (21) with $P_h^{n+\frac{1}{2}}$ replaced by the kinematic pressure $p_h^{n+\frac{1}{2}}$ and $\tilde{\mathbf{g}} = \mathbf{f} - \mathbf{g}_t - (\nabla \times \mathbf{g}) \times \mathbf{g} - \frac{1}{2}(\nabla|\mathbf{g}|^2)$ on Γ_w .

5.2 Numerical Experiment 1: convergence rates

Our first numerical experiment is designed to test the accuracy of the proposed boundary condition, by calculating convergence rates of the following finite element method solution, for a steady problem with known analytical solution given on $\Omega = (0, 1)^3$ by

$$\begin{aligned}\mathbf{u}(x, y, z) &= \{\sin(2\pi y), \cos(2\pi z), e^x\}^T \\ p(x, y, z) &= \sin(2\pi x) + \cos(2\pi y) + \sin(2\pi z).\end{aligned}$$

We choose $\nu = 1$, $\mathbf{g} = \mathbf{u}$ on $\partial\Omega$, and the forcing \mathbf{f} is calculated from the Navier-Stokes momentum equation and the analytical solution. Vorticity approximations are computed using Step 2, with $\Delta t = \infty$ and taking \mathbf{u}_h^{n+1} as the nodal interpolant of the analytical velocity solution, using \mathbf{Q}_1 , \mathbf{Q}_2 and \mathbf{Q}_3 elements on uniform quadrilateral meshes, and the software deal.II [2, 3]. The calculated vorticity errors and rates are given in Tables 1-3, and we observe the rates are optimal in both the L^2 and H^1 norms, for both choices of elements. For \mathbf{Q}_3 elements, we observe a slightly higher than optimal rate of convergence in the H^1 norm, but believe that with finer meshes this will reduce to third order.

h	$\ \mathbf{w} - \mathbf{w}_h\ _0$	rate	$\ \mathbf{w} - \mathbf{w}_h\ _1$	rate
1/4	1.440E-0		1.776E+1	
1/8	3.736E-1	1.95	8.942E-0	0.99
1/16	9.427E-2	1.99	4.475E-0	1.00
1/32	2.362E-2	2.00	2.238E-0	1.00

Table 1: Vorticity L^2 and H^1 errors and rates from \mathbf{Q}_1 element computations.

h	$\ \mathbf{w} - \mathbf{w}_h\ _0$	rate	$\ \mathbf{w} - \mathbf{w}_h\ _1$	rate
1/4	1.158E-1		3.587E-0	
1/8	1.466E-2	2.98	9.106E-1	1.98
1/16	1.835E-3	3.00	2.278E-1	2.00
1/24	5.436E-4	3.00	1.012E-1	2.00

Table 2: Vorticity L^2 and H^1 errors and rates from \mathbf{Q}_2 element computations.

h	$\ \mathbf{w} - \mathbf{w}_h\ _0$	rate	$\ \mathbf{w} - \mathbf{w}_h\ _1$	rate
1/2	2.511E-1		1.019E-0	
1/4	1.741E-2	3.85	7.995E-2	3.67
1/8	1.123E-3	3.95	6.082E-3	3.72
1/12	2.234E-4	3.98	1.369E-3	3.68

Table 3: Vorticity L^2 and H^1 errors and rates from \mathbf{Q}_3 element computations.

5.3 Numerical Experiment 2: Flow past a normal flat plate

We consider next a numerical experiment for flow past normal flat plate, following [37, 29, 38]. We take as the domain the $[-7, 20] \times [-10, 10]$ rectangle channel with a 0.125×1 flat plate placed 10 units into the channel from the left, and centered top to bottom. The inflow velocity is taken to be $u_{in} = \langle 1, 0 \rangle$, and $\mathbf{f} = \mathbf{0}$. We run tests with $\nu = \frac{1}{100}$ and $\nu = \frac{1}{150}$, giving Reynolds numbers of $Re = 100$ and $Re = 150$, respectively, based on the height of the plate. On the walls and plate, no-slip conditions are enforced for velocity, and no-penetration for vorticity along with the additional natural boundary conditions derived herein. For the outflow, the zero-traction boundary condition is enforced for velocity, and the homogeneous Neumann condition for vorticity. We note that, due to the outflow condition, we use only Alternate Step 1, since it uses usual pressure and thus more easily enforces zero traction (with the ‘do-nothing’ condition).

We compute both with the proposed scheme, and for comparison, we also compute using typical schemes for velocity-pressure in primitive variables. In particular, we use the standard Crank-Nicolson linear extrapolation algorithm, which is given by

CNLE-UP: Find $(\mathbf{u}_h, p_h) \in (\mathbf{V}_h, Q_h)$ satisfying

$$\begin{aligned} \frac{1}{\Delta t}(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n, \mathbf{v}_h) + \left(\left(\frac{3}{2}\mathbf{u}_h^n - \frac{1}{2}\mathbf{u}_h^{n-1} \right) \cdot \nabla \mathbf{u}_h^{n+\frac{1}{2}}, \mathbf{v}_h \right) \\ - (p_h^{n+\frac{1}{2}}, \nabla \cdot \mathbf{v}_h) + \nu(\nabla \mathbf{u}_h^{n+1/2}, \nabla \mathbf{v}_h) &= (\mathbf{f}^{n+\frac{1}{2}}, \mathbf{v}_h) \quad \forall \mathbf{v}_h \in \mathbf{V}_h^0, \\ (\nabla \cdot \mathbf{u}_h^{n+1}, q_h) &= 0 \quad \forall q_h \in Q_h. \end{aligned}$$

For the $Re = 150$ simulation, we use an analogous decoupled linearized scheme with BDF3 timestepping.

The quantities of interest in the simulations are the time averaged drag coefficient and the Strouhal number. The drag coefficients are calculated at each timestep by the formula,

$$C_d(t^m) = \frac{2}{LU^2} \int_S \left(\nu \frac{\partial \mathbf{u}_{t_S}(t^m)}{\partial n} n_y - p_h^m n_x \right) dS,$$

where S is the plate boundary, $\mathbf{n} = \langle n_x, n_y \rangle$ is the outward normal to S , $\mathbf{u}_{t_S}(t^m)$ is the tangential velocity, the maximum inlet velocity $U = 1$, and $L = 1$ is the plate length. The drag coefficients are then averaged over the last 10 periods in the simulation. In all cases, volume integral formulas are used, as they are believed more accurate [20]. The Strouhal number was calculated as in [37, 38], using the fast Fourier transform of the transverse velocity at (4.0,0.0) from $T=130$ to $T=200$ to calculate the frequency f , and then $St = fL/U = f$

5.3.1 $Re = 100$

The $Re = 100$ simulation with the Alternate Step 1 velocity-vorticity (CNLE-VV) scheme was run using (P_2, P_1, P_2) elements for velocity, pressure and vorticity, and used grad-div stabilization [32]. This provided 24,264 velocity dof, 3,073 pressure dof, and 12,132 vorticity dof, using a Delaunay generated mesh with aspect ratio of approximately 80. A timestep was chosen to be $\Delta t = 0.02$, and the simulation was run to the end time of $T = 200$; the flow reached a statistically steady, periodic-in-time state by around $T = 120$. For a fair comparison, we also ran CNLE-UP using the same mesh, timestep, and with the same (P_2, P_1) velocity-pressure element choice.

Table 4 shows the time averaged drag coefficients and Strouhal numbers from the solutions, along with reference values from [38]. Comparing to the reference values, we observe that both CNLE-VV and CNLE-UP accurately predict the Strouhal number, however the velocity-vorticity method with the proposed boundary conditions gives a much better approximation of the time averaged drag coefficient.

Figures 1 shows both the time averaged vorticities and the $T=200$ instantaneous vorticities for CNLE-VV and CNLE-UP. For CNLE-UP, the curl of the velocity was used as the vorticity and for VV, the vorticity variable w_h was used. We observe that CNLE-UP vorticity and averaged vorticity appears significantly less resolved than for CNLE-VV on this same discretization.

Re	scheme	C_d	Strouhal number
100	CNLE-VV	2.61	0.183
100	CNLE-UP	2.74	0.183
100	Reference [38]	2.60	0.183

Table 4: Long-time average drag coefficients and Strouhal numbers for the $Re=100$ simulations.

5.3.2 $Re=150$

For $Re = 150$ simulations, we found second order in time schemes to be too inaccurate, and so we used BDF3 timestepping, and changed velocity-vorticity scheme and primitive variable scheme accordingly. We will refer to the velocity-vorticity scheme as BDF3-VV, and the primitive variable scheme by BDF3-UP. We again use (P_2, P_1) elements for velocity and pressure, and for vorticity we use P_2 in the BDF3-VV scheme. Here we used a finer Delaunay-generated mesh, which provided 71,370 velocity dof, 8990 pressure dof, and 35,635 vorticity dof, and had an aspect ratio of around 90. A timestep of $\Delta t = 0.01$ was used to the end time of $T=200$; a statistically steady, periodic-in-time behavior was reached by around $T=120$. Strouhal numbers were calculated from solutions from $T=130$ to 200, and the average drag coefficient was taken by averaging the drag coefficients from the final 10 periods.

Table 5 shows the time averaged drag coefficients and Strouhal numbers from the solutions, along with reference values from [38]. Comparing to the reference values, we observe that both BDF3-VV and BDF3-UP predict the Strouhal number with good accuracy; the calculated Str number 0.171 is the closest discrete frequency to 0.167 (recall an FFT is used from $T=130$ to $T=200$). We also observe that BDF3-VV with the proposed boundary conditions is significantly closer than BDF3-UP to the reference solution’s time averaged drag coefficient. Figures 2 show both the time averaged vorticities and the $T=200$ instantaneous vorticities for BDF3-VV and BDF3-UP. We observe, as in the $Re=100$ case, that the primitive variable formulation’s vorticity and averaged vorticity appears significantly less resolved than for the velocity-vorticity method.

5.4 Numerical Experiment 3: Time dependent flow around a cylinder

Our next experiment tests the algorithms above based on the coupling (21) (Step 1) and (23) (Alternate Step 1), both using the proposed boundary condition, on the benchmark problem of time dependent flow around a cylinder. This test problem is taken from [20, 39],

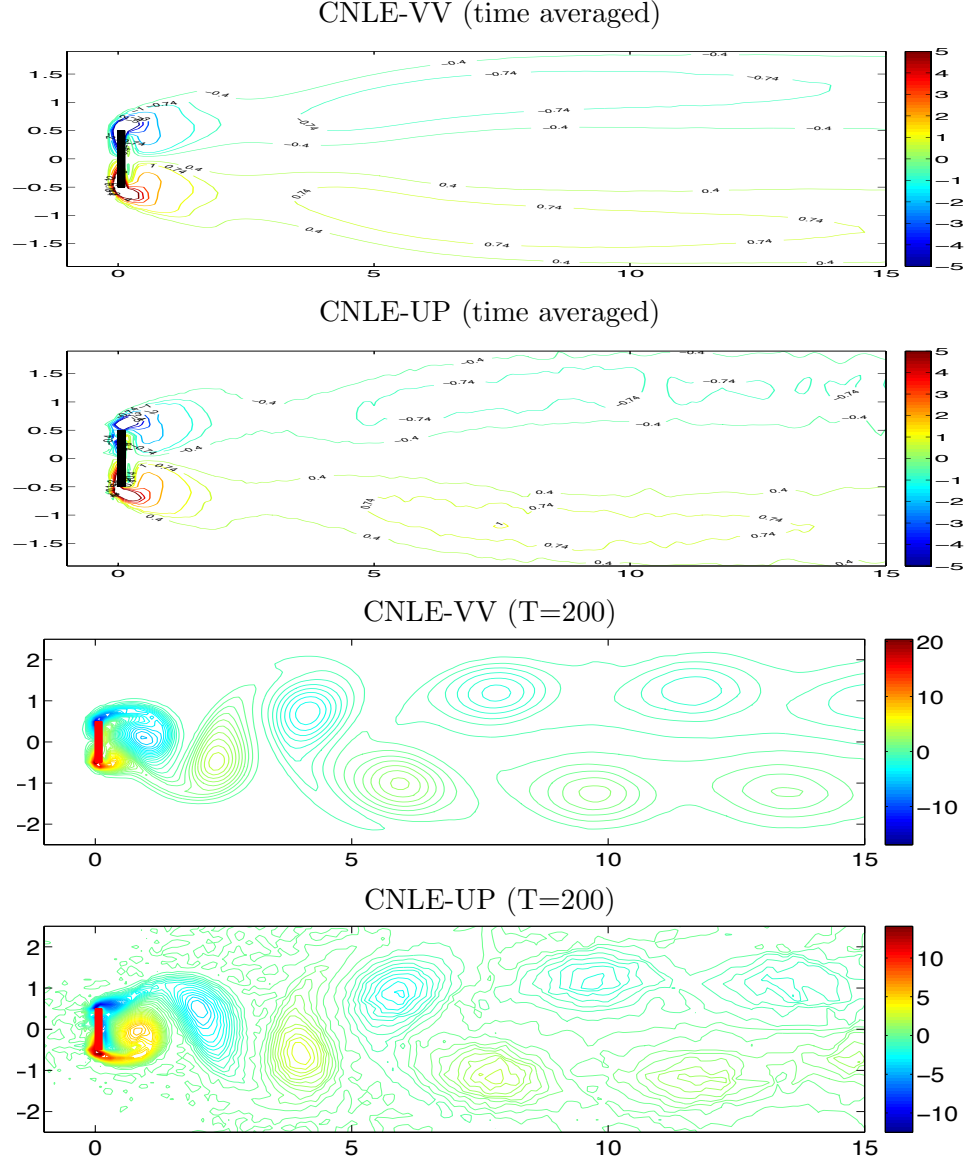


Figure 1: Shown above are plots of the vorticity contours for $Re=100$ for CNLE-VV and CNLE-UP, both time averaged and instantaneous at $T=200$.

Re	scheme	C_d	Strouhal number
150	BDF3-VV	2.65	0.171
150	BDF3-UP	2.78	0.171
150	Reference [38]	2.54	0.167

Table 5: Shown above are long-time average drag coefficients and Strouhal numbers for the $Re=150$ simulations.

and the domain for the problem is a 2.2×0.41 rectangular channel with a cylinder of radius 0.05 centered at $(0.2, 0.2)$ (taking the bottom left corner of the rectangle as the origin), see

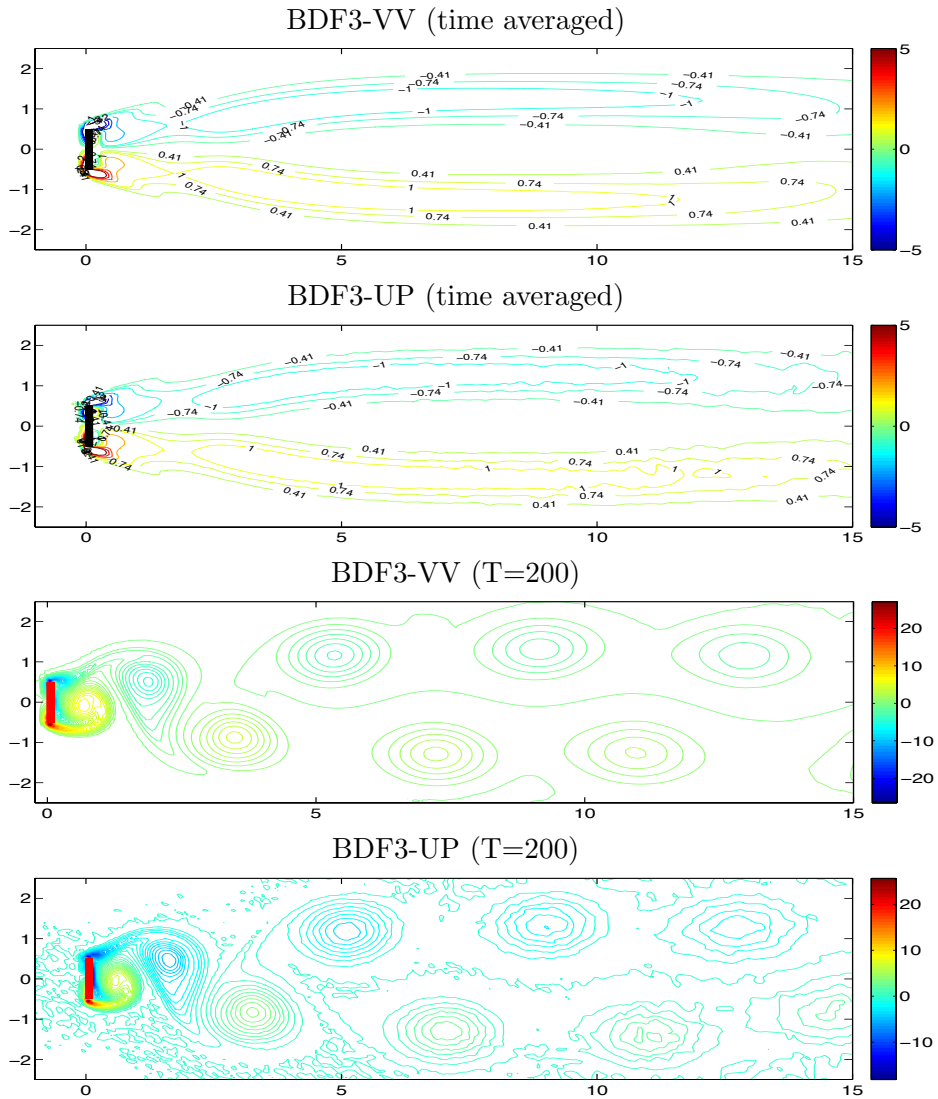


Figure 2: Shown above are plots of the vorticity contours for $Re=150$ for VV and CNLE, both time averaged and instantaneous at $T=200$.

Figure 3.

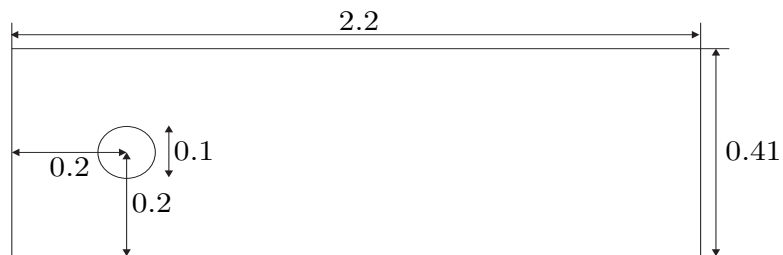


Figure 3: The domain for the channel flow around a cylinder numerical experiment.

The cylinder, top and bottom of the channel are prescribed no slip boundary conditions, and the time dependent inflow and outflow profile are

$$\begin{aligned} u_1(0, y, t) = u_1(2.2, y, t) &= \frac{6}{0.41^2} \sin(\pi t/8)y(0.41 - y), \\ u_2(0, y, t) = u_2(2.2, y, t) &= 0. \end{aligned}$$

The viscosity is set as $\nu = 10^{-3}$ and the external force $\mathbf{f} = \mathbf{0}$. The Reynolds number of the flow, based on the diameter of the cylinder and on the mean velocity inflow is $0 \leq Re(t) \leq 100$.

It is known that as the flow rate increases from time $t = 2$ to $t = 4$, two vortices start to develop behind the cylinder. They then separate into the flow, and soon after a vortex street forms which can be visible through the final time $t = 8$. Lift and drag coefficients for fully resolved flows will lie in the reference intervals ([39])

$$c_{d,max}^{ref} \in [2.93, 2.97], \quad c_{l,max}^{ref} \in [0.47, 0.49]$$

For the lift and drag to be accurate, the correct prediction of the boundary layer is critical, and thus we believe this is another good test for the proposed vorticity boundary condition.

We compute with the finite element algorithms discussed above, for the systems (21) (Step 1- Step 2) and (23) (Alternate Step 1 - Step 2). A Delaunay triangulation is used as the mesh, and it provided 65,080 velocity dof (and 48,366 pressure dof) using $((P_2)^2, P_1^{disc}, P_2)$ velocity-pressure-vorticity elements. A time step of $\Delta t = 0.01$ is used for the timestepping to an endtime of $T = 8$. Plots of solutions at $T=4$ and $T=6$ are shown in Figure 4 for the solution of the scheme for (21), and agree well with the literature [20, 14]. The solution plots of the scheme for (23) are visually indistinguishable from that of (21).

The lift and drag coefficients were calculated using volume integral formulas (see e.g. [20]) to be:

$$\begin{aligned} (21) \text{ (Step 1-Step 2): } & \quad c_{d,max}^{VV} = 2.877 \quad c_{l,max}^{VV} = 0.508 \\ (23) \text{ (Alternate Step 1-Step 2): } & \quad c_{d,max}^{VV} = 2.955 \quad c_{l,max}^{VV} = 0.470 \end{aligned}$$

Hence using the scheme for (23) gave better lift and drag coefficient predictions that lie in the reference intervals, while the scheme for (21) did not. We believe this difference in accuracies is due to the use of the Bernoulli pressure in (21), since Bernoulli pressure is much more complex (and thus more error prone) than usual pressure in flows around objects [22]. Significant error in the Bernoulli pressure would cause error in the vorticity through the natural boundary condition, particularly near the boundaries, which in turn would cause velocity error through the nonlinearity in the momentum equation.

5.5 Numerical Experiment 4: 3D flow around a square cylinder

As a final test we compute the 3d flow around a square cylinder benchmark with Reynolds number 20. The geometry setup is given in Figure 5, see [33, 5] for more details and reference values. An important feature of the flow past square cylinder problem is the singularity of the geometry, which destroys the regularity of the Navier-Stokes solution. The regularity theory from [6] predicts $p \notin H^1(\Omega)$ and $\mathbf{u} \notin H^2(\Omega)^3$. This, in particular, implies that the pressure and vorticity gradients are both unbounded in the vicinity of the edges. This lack of solution smoothness makes the correct prediction of drag and lift coefficients

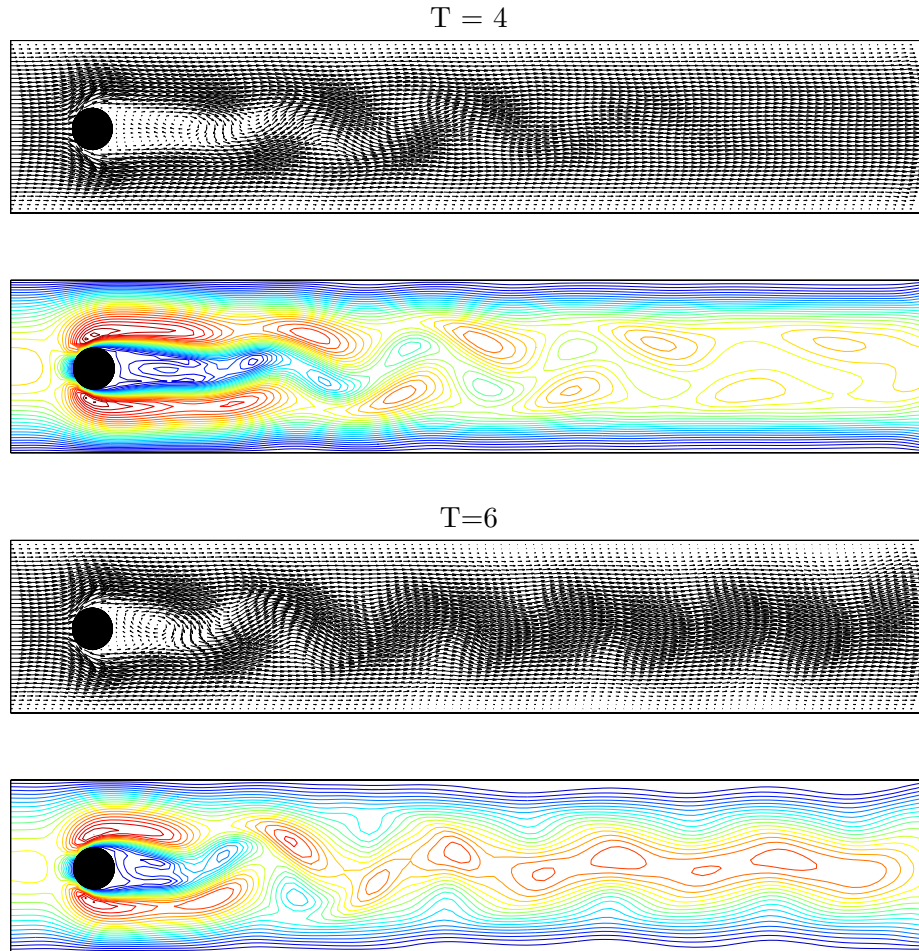


Figure 4: Velocity vector plot and speed contours at $T=4$ and $T=6$ for the 2D flow around a cylinder using the scheme for (23); the solution plots of (21) are visually indistinguishable.

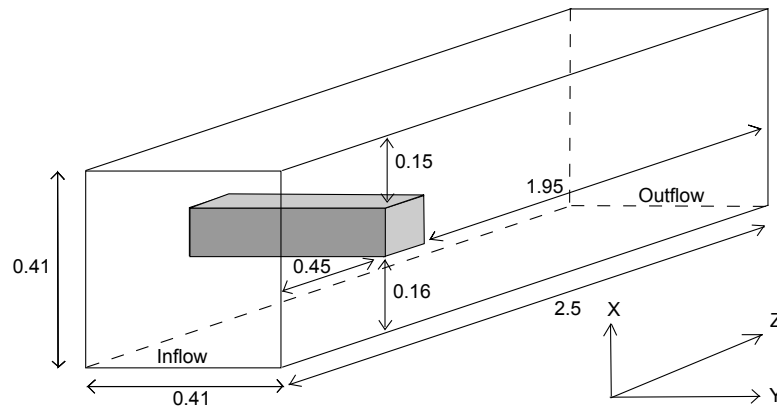


Figure 5: Setup for the 3d flow around a square cylinder

a challenging test for a numerical method based on vorticity as well as for boundary conditions involving pressure.

For the computations we use a code written in `deal.II` ([2, 3]) using Q2-Q1 Taylor-Hood elements on quadrilaterals. The starting mesh is adaptively refined using a gradient jump error estimator based on the previous velocity. A cut of the meshes used in the computations can be seen in Figure 6.

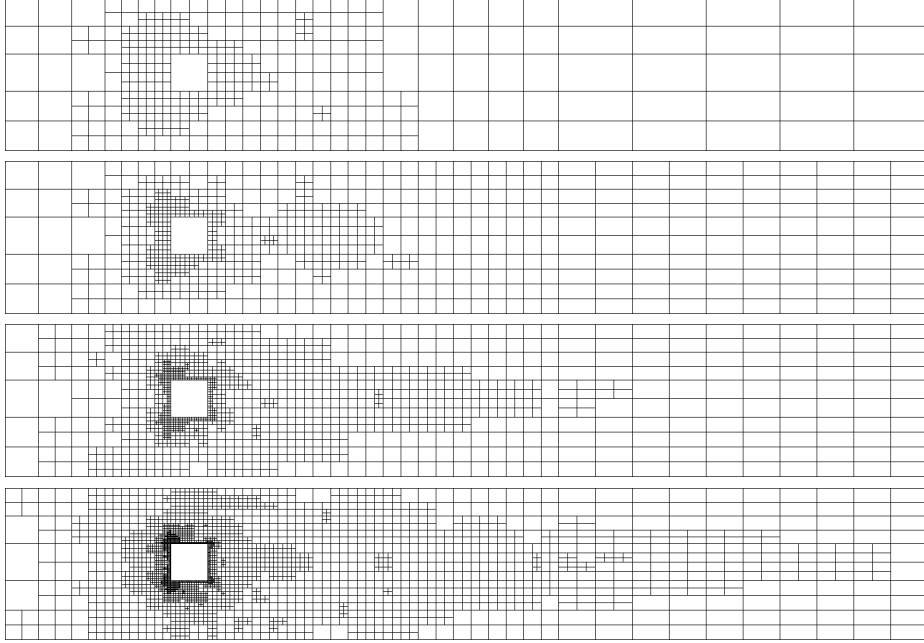


Figure 6: Cuts of the adaptively refined meshes in the middle of the domain. The total number of unknowns for velocity and pressure are 153,728, 498,807, 1,694,279, and 5,851,731, respectively.

As the solution for this Reynolds number is stationary, we solve the coupled vorticity-Navier-Stokes system as a stationary problem (in contrast to the examples before). We first solve the nonlinear velocity-pressure formulation

$$\begin{aligned} \nu(\nabla \mathbf{u}_h, \nabla \mathbf{v}_h) + (\mathbf{u}_h \cdot \nabla \mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h) + \gamma(\nabla \cdot \mathbf{u}_h, \nabla \cdot \mathbf{v}_h) &= (\mathbf{f}, \mathbf{v}_h) \quad \forall \mathbf{v}_h \in \mathbf{V}_h^0 \\ (\nabla \cdot \mathbf{u}_h, q_h) &= 0 \quad \forall q_h \in Q_h. \end{aligned} \tag{26}$$

using a damped Newton iteration until convergence. Note that we are adding grad-div stabilization with $\gamma = 0.1$ to the system to help with accuracy and the linear solvers for the saddle point system (see [16]). We then proceed with a fixed point iteration for the

vorticity formulation

$$\begin{aligned}
& \nu(\nabla \mathbf{u}_h, \nabla \mathbf{v}_h) + \frac{1}{2}(\mathbf{w}_h \times \mathbf{u}_h, \mathbf{v}_h) + (D(\mathbf{u}_h)\mathbf{u}_h, \mathbf{v}_h) \\
& \quad - (p_h, \nabla \cdot \mathbf{v}_h) + \gamma(\nabla \cdot \mathbf{u}_h, \nabla \cdot \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h) \quad \forall \mathbf{v}_h \in \mathbf{V}_h^0 \\
& \quad (\nabla \cdot \mathbf{u}_h, q_h) = 0 \quad \forall q_h \in Q_h \\
& \nu(\nabla \times \mathbf{w}_h, \nabla \times \boldsymbol{\chi}_h) + \nu(\nabla \cdot \mathbf{w}_h, \nabla \cdot \boldsymbol{\chi}_h) - (\mathbf{w}_h \cdot \nabla \mathbf{u}_h, \boldsymbol{\chi}_h) + (\mathbf{u}_h \cdot \nabla \mathbf{w}_h, \boldsymbol{\chi}_h) \\
& \quad - \int_{\Gamma_w} p_h (\nabla \times \boldsymbol{\chi}_h) \cdot \mathbf{n} \, ds + \int_{\partial \Gamma_w} p_h \boldsymbol{\chi}_h \cdot d\mathbf{l} = (\nabla \times \mathbf{f}, \boldsymbol{\chi}_h) \quad \forall \boldsymbol{\chi}_h \in \mathbf{W}_h^0
\end{aligned} \tag{27}$$

with the previously computed Navier-Stokes solution as a starting guess. In each fixed point iteration, we alternate between solving the stationary vorticity system and the stationary and linearized Navier-Stokes formulation. The iteration is stopped when the nonlinear residuals reach 10^{-6} .

The results for lift and drag are given in Table 6. For the vorticity-velocity method with the proposed boundary conditions, we observe the convergence of the statistics to the reference values despite a non-smooth and singular behavior of the pressure and vorticity in the vicinity of the upstream cylinder edges, see Figure 7, where the computed solution is visualized. We run experiments with both the standard velocity-pressure formulation (26) and the vorticity-velocity formulation (27) with the proposed vorticity boundary conditions. While for both formulations the statistics of interest converge to reference values, it is interesting that the vorticity formulation gives much better results for the drag, while the lift is better without using the vorticity form. Note that the meshes are coarse compared to the reference values in [5], which were computed with 20 to 30 million unknowns.

dofs	lift	err%	drag	err%	lift	err%	drag	err%
reference [5]	0.06893		7.767		0.06893		7.767	
	{ $\mathbf{w}, \mathbf{u}, p$ }-formulation				{ \mathbf{u}, p }-formulation			
153,728	0.0968	40.4%	<u>7.7843</u>	0.22%	<u>0.0772</u>	12.0%	8.0476	3.61%
497,886	0.0789	14.5%	<u>7.6619</u>	1.35%	<u>0.0758</u>	9.97%	7.9049	1.78%
1,691,920	<u>0.0720</u>	4.45%	7.6794	1.13%	0.0725	5.18%	<u>7.8306</u>	0.83%
5,848,298	<u>0.0684</u>	0.77%	7.7021	0.84%	0.0706	2.42%	<u>7.7993</u>	0.42%

Table 6: Values for lift and drag for the stationary flow around a 3d square cylinder on a sequence of adaptively refined meshes. Underlined numbers are more accurate. The number of unknowns in column one corresponds to the sum of velocity and pressure unknowns.

6 Conclusions

We have derived and tested new natural vorticity boundary conditions. The conditions are local and have been derived directly from the momentum balance for an incompressible fluid without invoking any further empirical or ad hoc assumptions. We argued that the devised condition possesses rich physical and geometrical information relevant to vortex dynamics, and we concentrated on employing them for the numerical simulation of viscous incompressible flows. Since methods that solve directly for vorticity are believed to be more accurate near the boundary for vortex dominated flows, using physically-derived boundary conditions should help to improve their accuracy. Despite the vorticity-pressure coupling, it appears that the conditions are easy to implement in finite element or other Galerkin

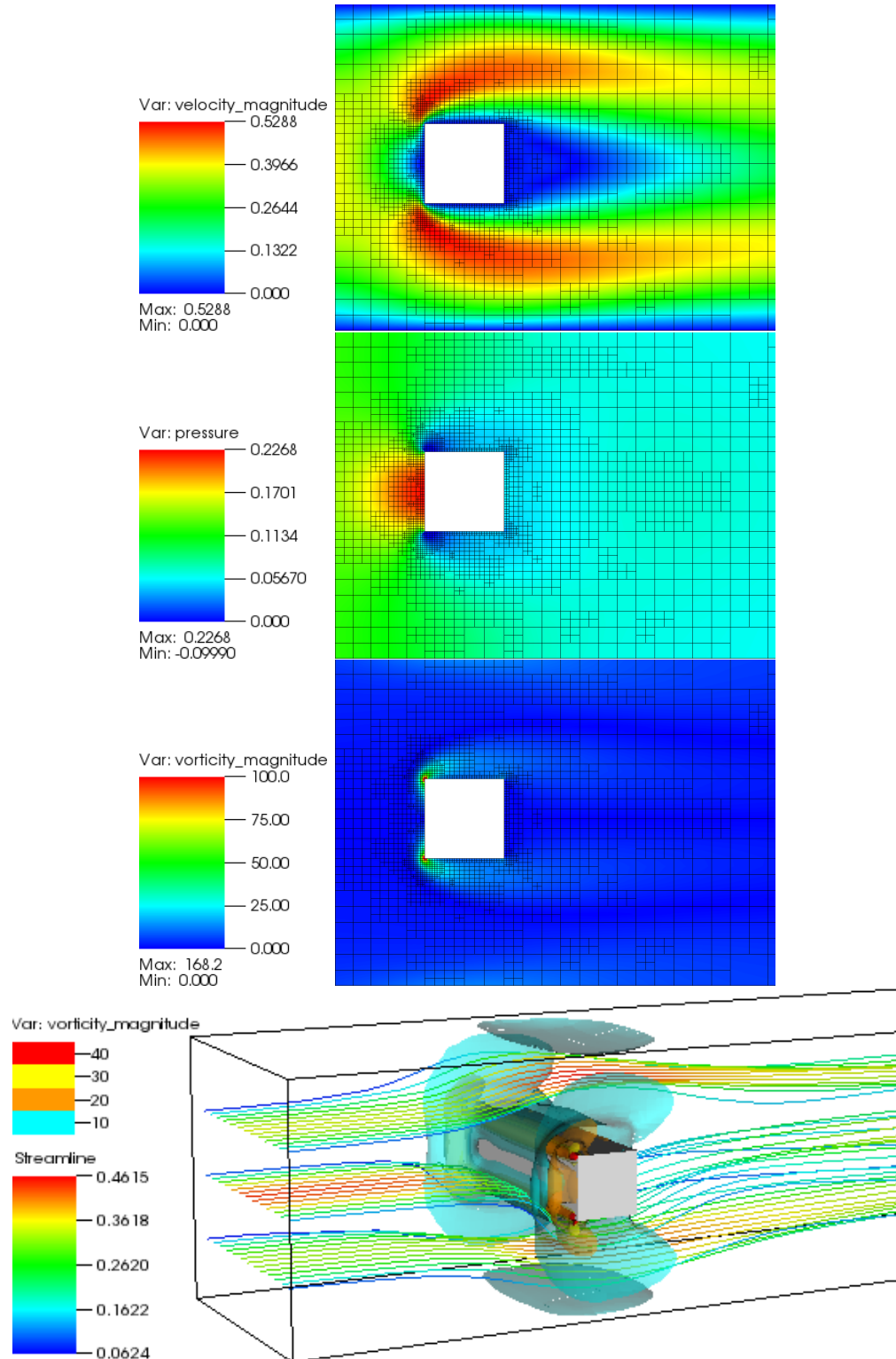


Figure 7: Velocity magnitude, pressure, and vorticity magnitude on the midplane around the cylinder (zoomed) and 3d view of streamlines and vorticity magnitude contours around the cylinder. The singularities in the pressure and vorticity in the corners of the cylinder making this a very challenging problem.

methods for velocity-vorticity formulations that solve for pressure. Two numerical formulations were suggested that benefit from the new vorticity boundary conditions and solve for velocity, vorticity and pressure in a decoupled time-stepping fashion. Several numerical experiments with laminar flows were provided to demonstrate the consistency and accuracy of the approach. In all experiments, computed solutions converge to reference data, and in those problems where near or far wake flow dynamics behind an object are of interest, the approach based on the vorticity equation and new boundary conditions demonstrated a superior performance.

References

- [1] C. R. Anderson. Vorticity boundary conditions and boundary vorticity generation for two-dimensional viscous incompressible flows. *Journal of Computational Physics*, 80:72–97, 1989.
- [2] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.
- [3] W. Bangerth, T. Heister, L. Heltai, G. Kanschat, M. Kronbichler, M. Maier, B. Turcksin, and T. D. Young. The deal.II library, version 8.2. *Archive of Numerical Software*, 3, 2015.
- [4] R. Bensow and M. Larson. Residual based VMS subgrid modeling for vortex flows. *Computer Methods in Applied Mechanics and Engineering*, 199:802–809, 2010.
- [5] M. Braack and T. Richter. Solutions of 3D Navier–Stokes benchmark problems with adaptive finite elements. *Computers & fluids*, 35(4):372–392, 2006.
- [6] M. Dauge. Stationary Stokes and Navier–Stokes systems on two-or three-dimensional domains with corners. Part I. linearized equations. *SIAM Journal on Mathematical Analysis*, 20(1):74–97, 1989.
- [7] P.A. Davidson. *Turbulence: an introduction for scientists and engineers*. Oxford University Press: New-York, 2004.
- [8] G. Dziuk and C. M. Elliott. L^2 -estimates for the evolving surface finite element method. *Mathematics of Computations*, 82(281):1–24, 2013.
- [9] W. E and J. G. Liu. Vorticity boundary condition and related issues for finite difference schemes. *Journal of Computational Physics*, 124:368–382, 1996.
- [10] T. B. Gatski. Review of incompressible fluid flow computations using the vorticity-velocity formulation. *Applied Numerical Mathematics*, 7:227–239, 1991.
- [11] D. Gilbarg and N. S. Trudinger. *Elliptic partial differential equations of second order*. Springer-Verlag, 1998.
- [12] S. Großand A. Reusken. *Numerical Methods for Two-phase Incompressible Flows*. Springer, Berlin, 2011.
- [13] G. Guevremont, W.G. Habashi, P.L. Kotiuga, and M.M. Hafez. Finite element solution of the 3D compressible Navier-Stokes equations by a velocity-vorticity method. *Journal of Computational Physics*, 107:176–187, 1993.

- [14] D. Hannasch and M. Neda. On the accuracy of the viscous form in simulations of incompressible flow problems. *Numerical Methods for Partial Differential Equations*, 28:523–541, 2012.
- [15] T. Heister, M.A. Olshanskii, and L. Rebholz. Unconditional long-time stability of a velocity-vorticity method for the 2D Navier–Stokes equations. Numerical Analysis & Scientific Computing Preprint Series 33, University of Houston, March 2015.
- [16] T. Heister and G. Rapin. Efficient augmented Lagrangian-type preconditioning for the Oseen problem using Grad-Div stabilization. *International Journal for Numerical Methods in Fluids*, 71(1):118–134, 2013.
- [17] J.G. Heywood, R. Rannacher, and S. Turek. Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 22:325–352, 1996.
- [18] P.H. Hoffmann, K.C. Muck, and P. Bradshaw. The effect of concave surface curvature on turbulent boundary layers. *Journal of Fluid mechanics*, 161:371–403, 1985.
- [19] T.J.R. Hughes J. A. Evans. Isogeometric divergence-conforming B-splines for the unsteady Navier-Stokes equations. *Journal of Computational Physics*, 241:141–167, 2013.
- [20] V. John. Reference values for drag and lift of a two dimensional time-dependent flow around a cylinder. *International Journal for Numerical Methods in Fluids*, 44:777–788, 2004.
- [21] P. Koumoutsakos, A. Leonard, and F. Pepin. Boundary conditions for viscous vortex methods. *Journal of Computational Physics*, 113:52–61, 1994.
- [22] W. Layton, C. Manica, M. Neda, M.A. Olshanskii, and L. Rebholz. On the accuracy of the rotation form in simulations of the Navier-Stokes equations. *J. Comput. Phys.*, 228(5):3433–3447, 2009.
- [23] H.K. Lee, M.A. Olshanskii, and L.G. Rebholz. On error analysis for the 3D Navier-Stokes equations in Velocity-Vorticity-Helicity form. *SIAM Journal on Numerical Analysis*, 49(2):711–732, 2011.
- [24] D. C. Lo, D. L. Young, and K. Murugesan. An accurate numerical solution algorithm for 3d velocity-vorticity Navier-Stokes equations by the DQ method. *Communications in Numerical Methods in Engineering*, 22:235–250, 2006.
- [25] A. J. Majda and A. L. Bertozzi. *Vorticity and incompressible flow*. Cambridge Univ. Press, Cambridge, 2001.
- [26] H. L. Meitz and H. F. Fasel. A compact-difference scheme for the Navier-Stokes equations in vorticity-velocity formulation. *Journal of Computational Physics*, 157:371–403, 2000.
- [27] B.R. Morton. The generation and decay of vorticity. *Geophysical & Astrophysical Fluid Dynamics*, 28(3-4):277–308, 1984.
- [28] K.C. Muck, P.H. Hoffmann, and P. Bradshaw. The effect of convex surface curvature on turbulent boundary layers. *Journal of Fluid Mechanics*, 161:347–369, 1985.

- [29] F. Najjar and S. Vanka. Simulations of the unsteady separated flow past a normal flat plate. *International Journal of Numerical Methods in Fluids*, 21:525–547, 1995.
- [30] J.Z. Wu and H.-Y. Ma and M-D. Zhou. *Vorticity and vortex dynamics*. Springer, Berlin, 2006.
- [31] M.A. Olshanskii and L. Rebholz. Velocity-Vorticity-Helicity formulation and a solver for the Navier-Stokes equations. *Journal of Computational Physics*, 229:4291–4303, 2010.
- [32] M.A. Olshanskii and A. Reusken. Grad-div stabilization for Stokes equations. *Mathematics of Computation*, 73(248):1699–1718, 2004.
- [33] M.A. Olshanskii, K.M. Terekhov, and Y.V. Vassilevski. An octree-based solver for the incompressible Navier–Stokes equations with enhanced stability and low dissipation. *Computers & Fluids*, 84:231–246, 2013.
- [34] L. Quartapelle. *Numerical solution of the incompressible Navier-Stokes equations*. ISNM 113. Birkhäuser, Basel, 1993.
- [35] V. Ruas. A new formulation of the three-dimensional velocity-vorticity system in viscous incompressible flow. *Math. Mech.*, 79:29–36, 1999.
- [36] P.G. Saffman. *Vortex dynamics*. Cambridge University Press, Cambridge, 1992.
- [37] A. Saha. Far-wake characteristics of two-dimensional flow past a normal flat plate. *Physics of Fluids*, 19:128110:1–4, 2007.
- [38] S. Saha. Direct numerical simulation of two-dimensional flow past a normal flat plate. *Journal of Engineering Mechanics*, 139(12):1894–1901, 2013.
- [39] M. Schäfer and S. Turek. The benchmark problem ‘flow around a cylinder’ flow simulation with high performance computers ii. in *E.H. Hirschel (Ed.), Notes on Numerical Fluid Mechanics*, 52, Braunschweig, Vieweg:547–566, 1996.
- [40] C.G. Speziale. On the advantages of the vorticity-velocity formulation of the equations of fluid dynamics. *Journal of Computational Physics*, 73:476–480, 1987.
- [41] Itiro Tani. Production of longitudinal vortices in the boundary layer along a concave wall. *Journal of Geophysical Research*, 67(8):3075–3080, 1962.
- [42] J. Trujillo and G. E. Karniadakis. A penalty method for the vorticity-velocity formulation. *Journal of Computational Physics*, 149:32–58, 1999.
- [43] W.-Z. Shen and T.-P. Loc. Numerical method for unsteady 3D Navier-Stokes equations in vorticity-velocity form. *Computers & Fluids*, 26:193–216, 1997.
- [44] K.L. Wong and A.J. Baker. A 3d incompressible Navier-Stokes velocity-vorticity weak form finite element algorithm. *International Journal for Numerical Methods in Fluids*, 38:99–123, 2002.
- [45] X.H. Wu, J.Z. Wu, and J.M. Wu. Effective vorticity-velocity formulations for the three dimensional incompressible viscous flows. *Journal of Computational Physics*, 122:68–82, 1995.