

A High-Order Discontinuous Galerkin Method for 2D Incompressible Flows

Jian-Guo Liu^{*,1} and Chi-Wang Shu^{†,2}

**Institute for Physical Science and Technology and Department of Mathematics, University of Maryland, College Park, Maryland 20742; and †Division of Applied Mathematics, Brown University, Providence, Rhode Island 02912*
E-mail: *jliu@math.umd.edu, †shu@cfm.brown.edu

Received June 4, 1999; revised January 26, 2000

In this paper we introduce a high-order discontinuous Galerkin method for two-dimensional incompressible flow in the vorticity stream-function formulation. The momentum equation is treated explicitly, utilizing the efficiency of the discontinuous Galerkin method. The stream function is obtained by a standard Poisson solver using continuous finite elements. There is a natural matching between these two finite element spaces, since the normal component of the velocity field is *continuous* across element boundaries. This allows for a correct upwinding gluing in the discontinuous Galerkin framework, while still maintaining total energy conservation with no numerical dissipation and total enstrophy stability. The method is efficient for inviscid or high Reynolds number flows. Optimal error estimates are proved and verified by numerical experiments. © 2000 Academic Press

Key Words: incompressible flow; discontinuous Galerkin; high-order accuracy.

1. INTRODUCTION AND THE SETUP OF THE SCHEME

We are interested in solving the following 2D time-dependent incompressible Euler equations in vorticity stream-function formulation;

$$\begin{aligned}\omega_t + \nabla \cdot (\mathbf{u}\omega) &= 0 \\ \Delta\psi &= \omega, \quad \mathbf{u} = \nabla^\perp\psi, \\ \mathbf{u} \cdot \mathbf{n} &= \text{given on } \partial\Omega,\end{aligned}\tag{1.1}$$

¹ Research supported by NSF Grant DMS-9805621.

² Research supported by ARO Grant DAAG55-97-1-0318, NSF Grants DMS-9804985, ECS-9906606, and INT-9601084, NASA Langley Grant NAG-1-2070 and Contract NAS1-97046 while this author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23681-2199, and AFOSR Grant F49620-99-1-0077.

where $\nabla^\perp = (-\partial_y, \partial_x)$. Notice that the boundary condition, plus the fact that $\mathbf{u} \cdot \mathbf{n} = \frac{\partial \psi}{\partial \tau}$, recovers ψ on the boundary (up to a constant) in a simple connected domain

$$\psi|_{\partial\Omega} = \psi_b. \quad (1.2)$$

We are also interested in solving the Navier–Stokes equations with high Reynolds numbers $\text{Re} \gg 1$:

$$\begin{aligned} \omega_t + \nabla \cdot (\mathbf{u}\omega) &= \frac{1}{\text{Re}} \Delta \omega \\ \Delta \psi &= \omega, \quad \mathbf{u} = \nabla^\perp \psi, \\ \mathbf{u} &= \text{given on } \partial\Omega. \end{aligned} \quad (1.3)$$

The boundary condition is now (1.2) plus the non-slip type boundary condition:

$$\left. \frac{\partial \psi}{\partial \mathbf{n}} \right|_{\partial\Omega} = \mathbf{u}_{b,\tau}. \quad (1.4)$$

For simplicity, we only consider the no-flow, no-slip boundary conditions $\psi_b = 0$, $\mathbf{u}_{b,\tau} = 0$ and periodic boundary conditions.

We first emphasize that, for Euler equations (1.1) and high Reynolds number ($\text{Re} \gg 1$) Navier–Stokes equations (1.3), it is advantageous to treat both the convective terms and the viscous terms explicitly. The methods discussed in this paper are stable under standard CFL conditions. Since the momentum equation (the first equation in (1.1) and (1.3)) is treated explicitly in the discontinuous Galerkin framework, there is no global mass matrix to invert, unlike conventional finite element methods. This makes the method highly efficient for parallel implementation, see for example [3]. As any finite element method, our approach has the flexibility for complicated geometry and boundary conditions. The method is adapted from the Runge–Kutta discontinuous Galerkin methods discussed by Cockburn *et al.* in a series of papers [7–13, 20].

The main difficulties in solving incompressible flows are the incompressibility condition and boundary conditions. The incompressibility condition is global and is thus solved by the standard Poisson solver for the stream function ψ using continuous finite elements. One advantage of our approach is that there is no matching conditions needed for the two finite element spaces for the vorticity ω and for the stream function ψ . The incompressibility condition, represented by the stream function ψ , is exactly satisfied pointwise and is naturally matched with the convective terms in the momentum equation. The normal velocity $\mathbf{u} \cdot \mathbf{n}$ is automatically *continuous* along any element boundary, allowing for correct upwinding for the convective terms and still maintaining a total energy conservation and total enstrophy stability.

There is an easy proof for stability, both in the total enstrophy and in the total energy, which does not depend on the regularity of the exact solutions. For smooth solutions error estimates can be obtained.

We use the vorticity stream-function formulation of the Navier–Stokes equations. This formulation with the local vorticity boundary condition has been revitalized by the recent work of E and Liu [14, 15, 23]. The main idea is to use convectively stable time-stepping procedure to overcome the cell Reynolds number constraint, explicit treatment of the viscous terms and the local vorticity boundary condition. This results in a decoupling of the

computation of stream function and vorticity at every time step. This method is very efficient and accurate for moderate to high Reynolds number flows, as demonstrated in [14, 15, 23].

Our method, as it stands, can only compute 2D flows. In 3D, the normal velocity $\mathbf{u} \cdot \mathbf{n}$ is no longer continuous along an element boundary, hence making the method more complicated to design and to analyze. Similar approaches for the stream-function vorticity formulation, or for the primitive variable formulation, suitable for 3D calculations, are under investigation.

We do not advocate our method for modest or low Reynolds number flows. In such regime viscous terms should be treated implicitly for efficiency. This is a much more challenging task in terms of space matching characterized by the Babuška–Brezzi–Ladyzenskaja condition, projection type methods, and global vorticity boundary conditions; see, for example [4, 17–19, 25, 26, 28] etc. We remark that the only problem of our method for modest or low Reynolds number flows is the small time step dictated by the stability of the explicit time discretization. Of course, if the objective is to *resolve* the full viscous effect, hence a small time step is justified for accuracy, then it is still adequate to use our method.

For convection-dominated flows, as we investigate in this paper, we mention the work of Bell *et al.* [1] for second-order Godunov-type upwinding methods; see also Levy and Tadmor [22] and E and Shu [16]. This is still an active field for research.

We now describe the setup of the scheme. We start with a triangulation \mathcal{T}_h of the domain Ω , consisting of polygons of maximum size (diameter) h , and the following two approximation spaces

$$V_h^k = \{v : v|_K \in P^k(K), \forall K \in \mathcal{T}_h\}, \quad W_{0,h}^k = V_h^k \cap C_0(\Omega), \quad (1.5)$$

where $P^k(K)$ is the set of all polynomials of degree at most k on the cell K .

For the Euler equations (1.1), the numerical method is defined as follows: find $\omega_h \in V_h^k$ and $\psi_h \in W_{0,h}^k$, such that

$$\langle \partial_t \omega_h v \rangle_K - \langle \omega_h \mathbf{u}_h \cdot \nabla v \rangle_K + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} \widehat{\omega}_h v^- \rangle_e = 0, \quad \forall v \in V_h^k, \quad (1.6)$$

$$-\langle \nabla \psi_h \cdot \nabla \varphi \rangle = \langle \omega_h \varphi \rangle, \quad \forall \varphi \in W_{0,h}^k, \quad (1.7)$$

with the velocity field obtained from the stream function by

$$\mathbf{u}_h = \nabla^\perp \psi_h. \quad (1.8)$$

Here $\langle \cdot \rangle$ is the usual integration over either the whole domain Ω or a subdomain denoted by a subscript; similarly for the L^2 norm $\| \cdot \|$.

Notice that the normal velocity $\mathbf{u}_h \cdot \mathbf{n}$ is continuous across any element boundary e , but both the solution ω_h and the test function v are discontinuous there. We take the values of the test function from within the element K , denoted by v^- . The solution at the edge is taken as a single-valued flux $\widehat{\omega}_h$, which can be either a central or an upwind-biased average. For example, the central flux is defined by

$$\widehat{\omega}_h = \frac{1}{2}(\omega_h^+ + \omega_h^-), \quad (1.9)$$

where ω_h^+ is the value of ω_h on the edge e from outside K , the complete upwind flux is

defined by

$$\widehat{\omega}_h = \begin{cases} \omega_h^- & \text{if } \mathbf{u}_h \cdot \mathbf{n} \geq 0, \\ \omega_h^+ & \text{if } \mathbf{u}_h \cdot \mathbf{n} < 0, \end{cases} \tag{1.10}$$

and the Lax–Friedrichs upwind biased flux is defined by

$$\mathbf{u}_h \cdot \mathbf{n} \widehat{\omega}_h = \frac{1}{2}[\mathbf{u}_h \cdot \mathbf{n}(\omega_h^+ + \omega_h^-) - \alpha(\omega_h^+ - \omega_h^-)], \tag{1.11}$$

where α is the maximum of $|\mathbf{u}_h \cdot \mathbf{n}|$ either locally (local Lax–Friedrichs) or globally (global Lax–Friedrichs).

We remark that, for general boundary conditions (1.2), the space $W_{0,h}^k$ in (1.5) should be modified to take the boundary value into consideration. Moreover, additional physical vorticity boundary condition for any inlet should be given.

Navier–Stokes equations (1.3) can be handled in a similar way, with the additional viscous terms treated by the local discontinuous Galerkin technique in [13], and with a local vorticity boundary condition in [14]. The details are left to Section 3. Section 2 is devoted to the discussion of stability and error estimates for the Euler equations. Accuracy check and numerical examples are given in Section 4. Concluding remarks are given in Section 5.

2. STABILITY AND ERROR ESTIMATES FOR THE EULER EQUATIONS

For stability analysis, we take the test function $v = \omega_h$ in (1.6), obtaining

$$\frac{d}{dt} \frac{1}{2} \|\omega_h\|_K^2 - \frac{1}{2} \langle \nabla \cdot (\omega_h^2 \mathbf{u}_h) \rangle_K + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} \widehat{\omega}_h \omega_h^- \rangle_e = 0,$$

where we have used the exact incompressibility condition satisfied by \mathbf{u}_h for the second term. Performing an integration by parts for the second term, we obtain

$$\frac{d}{dt} \frac{1}{2} \|\omega_h\|_K^2 + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} (\widehat{\omega}_h \omega_h^- - \frac{1}{2}(\omega_h^-)^2) \rangle_e = 0.$$

Now, using the fact that

$$\omega^- = \bar{\omega} - \frac{1}{2}[\omega], \quad (\omega^-)^2 = \bar{\omega}^2 - \bar{\omega}[\omega],$$

where

$$\bar{\omega} = \frac{1}{2}(\omega^+ + \omega^-), \quad [\omega] = \omega^+ - \omega^-,$$

we obtain

$$\frac{d}{dt} \frac{1}{2} \|\omega_h\|_K^2 + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} (\widehat{\omega}_h \bar{\omega}_h - \frac{1}{2} \bar{\omega}_h^2) \rangle_e + \frac{1}{2} \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} [\omega_h] (\bar{\omega}_h - \widehat{\omega}_h) \rangle_e = 0.$$

Notice that the second term is of opposite sign for adjacent elements sharing a common edge e , hence it becomes zero after summing over all the elements K (using the no-flow

boundary condition on the physical boundary or periodic boundary conditions). The third term is the numerical dissipation: when $\widehat{\omega}_h$ is taken as the central flux (1.9), the third term is exactly zero; for the upwind flux (1.10), the third term becomes a positive quantity

$$\frac{1}{4} \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} | [\omega_h]^2 \rangle_e, \quad (2.1)$$

which is the total enstrophy dissipation. The effect of this is to control the size of the jump across the element interface and essentially “gluing” the solution there. Other upwind-biased fluxes such as the Lax–Friedrichs flux (1.11) would produce a positive term similar to that of the total enstrophy dissipation. For smooth flows these jumps are of the order $O(h^k)$ within the truncation error of the scheme. We thus obtain the enstrophy inequality

$$\frac{d}{dt} \|\omega_h\|^2 \leq 0, \quad (2.2)$$

which becomes an equality if the central flux (1.9) is used.

The stability for the velocity field is now straightforward: we take $\varphi = \psi_h$ in (1.7) to obtain

$$\langle \nabla \psi_h \cdot \nabla \psi_h \rangle = -\langle \omega_h \psi_h \rangle \leq \|\psi_h\| \|\omega_h\| \leq C \|\nabla \psi_h\| \|\omega_h\|$$

by the Poincaré inequality, which implies

$$\|\mathbf{u}_h\| = \|\nabla \psi_h\| \leq C \|\omega_h\|. \quad (2.3)$$

Indeed, we can obtain a total energy conservation through the following arguments. Taking $v = \psi_h$ in (1.6), we obtain

$$\langle \partial_t \omega_h \psi_h \rangle_K - \langle \omega_h \mathbf{u}_h \cdot \nabla \psi_h \rangle_K + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} \widehat{\omega}_h \psi_h \rangle_e = 0.$$

Now the second term is zero since $\mathbf{u}_h \cdot \nabla \psi_h = 0$. The third term vanishes after summing over all elements since ψ_h is continuous. Finally, noticing that

$$-\langle \partial_t \omega_h \psi_h \rangle = \frac{d}{dt} \frac{1}{2} \|\nabla \psi_h\|^2 = \frac{d}{dt} \frac{1}{2} \|\mathbf{u}_h\|^2,$$

we obtain the conservation of energy

$$\frac{d}{dt} \|\mathbf{u}_h\| = 0 \quad (2.4)$$

even for a upwind flux. Thus there is *no numerical dissipation* for the energy.

We remark that the stability proof above for the total energy and total enstrophy does not need any hypotheses on the regularity of the solution or the mesh.

We now turn to the error estimates. For these we would need to assume that the solution is regular ($\omega \in H^{k+1}$ for $k > 1$) and the mesh is quasi-uniform. Conceptionally, since this is a finite element method, the exact solution of the PDE satisfies the scheme exactly. As

usual, we define the two projection operators: P is the standard L^2 projection into the space V_h^k ; and Π is the standard projection into $W_{0,h}^k$:

$$\langle \nabla(\psi - \Pi\psi) \cdot \nabla\varphi \rangle = 0, \quad \forall \varphi \in W_{0,h}^k.$$

Denote the error functions by

$$\varepsilon = \omega - \omega_h, \quad \delta = \psi - \psi_h$$

and their projections by

$$\varepsilon_h = P\varepsilon = P\omega - \omega_h, \quad \delta_h = \Pi\delta = \Pi\psi - \psi_h.$$

We first obtain a control of δ_h in terms of ε ,

$$\langle \nabla\delta \cdot \nabla\varphi \rangle = -\langle \varepsilon\varphi \rangle, \quad \forall \varphi \in W_{0,h}^k$$

from the scheme (1.7) and the fact that the exact solution also satisfies (1.7). Now, taking $\varphi = \delta_h$, we obtain

$$\langle \nabla\delta_h \cdot \nabla\delta_h \rangle = \langle \nabla\delta \cdot \nabla\delta_h \rangle = -\langle \varepsilon\delta_h \rangle,$$

which gives

$$\|\nabla\delta_h\| \leq C\|\varepsilon\|.$$

This leads to a bound for the velocity field

$$\begin{aligned} \|\mathbf{u} - \mathbf{u}_h\| &= \|\nabla(\psi - \psi_h)\| \leq \|\nabla(\psi - \Pi\psi)\| + \|\nabla(\Pi\psi - \psi_h)\| \\ &\leq \|\nabla(\psi - \Pi\psi)\| + C\|\varepsilon\|. \end{aligned} \quad (2.5)$$

Since both the numerical solution and the exact solution satisfy (1.6),

$$\langle \partial_t \varepsilon v \rangle_K - \langle (\omega \mathbf{u} - \omega_h \mathbf{u}_h) \cdot \nabla v \rangle_K + \sum_{e \in \partial K} \langle (\mathbf{u} \cdot \mathbf{n}\omega - \mathbf{u}_h \cdot \mathbf{n}\widehat{\omega}_h) v^- \rangle_e = 0, \quad \forall v \in V_h^k. \quad (2.6)$$

Take $v = \varepsilon_h$. The second term becomes

$$\langle (\omega \mathbf{u} - \omega_h \mathbf{u}_h) \cdot \nabla \varepsilon_h \rangle_K = \langle \omega (\mathbf{u} - \mathbf{u}_h) \cdot \nabla \varepsilon_h \rangle_K + \langle \varepsilon \mathbf{u}_h \cdot \nabla \varepsilon \rangle_K - \langle \varepsilon \mathbf{u}_h \cdot \nabla (\omega - P\omega) \rangle_K. \quad (2.7)$$

Noticing that $\mathbf{u} - \mathbf{u}_h$ is exactly divergence free, we may perform integration by parts to the first term on the right side of (2.7) to obtain

$$\langle \omega (\mathbf{u} - \mathbf{u}_h) \cdot \nabla \varepsilon_h \rangle_K = -\langle \varepsilon_h (\mathbf{u} - \mathbf{u}_h) \cdot \nabla \omega \rangle_K + \sum_{e \in \partial K} \langle (\mathbf{u} - \mathbf{u}_h) \cdot \mathbf{n}\omega \varepsilon_h^- \rangle_e.$$

The second term on the right side of (2.7) is a complete derivative and hence can be integrated to give a pure boundary term

$$\langle \varepsilon \mathbf{u}_h \cdot \nabla \varepsilon \rangle_K = \frac{1}{2} \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} (\varepsilon^-)^2 \rangle_e$$

Plugging all these into (2.6) with $v = \varepsilon_h$, and collecting boundary terms, we obtain

$$(\partial_t \varepsilon_h \varepsilon_h)_K + \langle \varepsilon_h (\mathbf{u} - \mathbf{u}_h) \cdot \nabla \omega \rangle_K + \langle \varepsilon \mathbf{u}_h \cdot \nabla (\omega - P\omega) \rangle_K + \sum_{e \in \partial K} I_e = 0$$

where the boundary terms

$$\begin{aligned} I_e &= -\langle (\mathbf{u} - \mathbf{u}_h) \cdot \mathbf{n} \omega \varepsilon_h^- \rangle_e - \frac{1}{2} \langle (\mathbf{u}_h \cdot \mathbf{n} (\varepsilon^-)^2) \rangle_e + \langle (\mathbf{u} \cdot \mathbf{n} \omega - \mathbf{u}_h \cdot \widehat{\mathbf{n}} \omega_h) \varepsilon_h^- \rangle_e \\ &= \langle \mathbf{u}_h \cdot \mathbf{n} (\hat{\varepsilon} \varepsilon_h^- - \frac{1}{2} (\varepsilon^-)^2) \rangle_e \\ &= \langle \mathbf{u}_h \cdot \mathbf{n} (\hat{\varepsilon} \varepsilon^- - \frac{1}{2} (\varepsilon^-)^2) \rangle_e - \langle \mathbf{u}_h \cdot \mathbf{n} \hat{\varepsilon} (\omega - (P\omega)^-) \rangle_e. \end{aligned}$$

Using the stability analysis in (2.2), we are left with

$$\begin{aligned} \frac{d}{dt} \frac{1}{2} \|\varepsilon_h\|^2 &\leq \sum_K \left\{ -\langle \varepsilon_h (\mathbf{u} - \mathbf{u}_h) \cdot \nabla \omega \rangle_K - \langle \varepsilon \mathbf{u}_h \cdot \nabla (\omega - P\omega) \rangle_K \right. \\ &\quad \left. + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} \hat{\varepsilon} (\omega - (P\omega)^-) \rangle_e \right\}. \end{aligned} \quad (2.8)$$

Assuming for the moment

$$\|\mathbf{u}_h\|_\infty \leq C, \quad (2.9)$$

we can first estimate the boundary term, using the fact that the mesh is quasi-uniform:

$$\begin{aligned} \sum_K \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} \hat{\varepsilon} (\omega - (P\omega)^-) \rangle_e &\leq \sum_K \sum_{e \in \partial K} C \|[P\omega]\|_e \|\hat{\varepsilon}\|_e \\ &\leq \|\varepsilon\|^2 + \frac{C}{h} \sum_K \sum_{e \in \partial K} \|[P\omega]\|_e^2. \end{aligned}$$

Using the above inequality together with (2.5) and (2.8), we now obtain (with the regularity assumption $\omega \in H^{k+1}$)

$$\frac{d}{dt} \|\varepsilon_h\|^2 \leq C \left(\|\varepsilon_h\|^2 + \|\nabla(\psi - \Pi\psi)\|^2 + \|\omega - P\omega\|_{H^1}^2 + \frac{1}{h} \sum_K \sum_{e \in \partial K} \|[P\omega]\|_e^2 \right).$$

Here we understand the norms as a summation of the same norm on each K . Using the standard interpolation theory [6], we obtain

$$\frac{d}{dt} \|\varepsilon_h\|^2 \leq C \|\varepsilon_h\|^2 + Ch^{2k},$$

which yields

$$\|\varepsilon_h\| \leq Ch^k.$$

Together with (2.5), we have

$$\|\mathbf{u} - \mathbf{u}_h\| + \|\omega - \omega_h\| \leq Ch^k. \quad (2.10)$$

Using an inverse inequality, we have

$$\|\mathbf{u} - \mathbf{u}_h\|_\infty \leq Ch^{k-1};$$

with the assumption $k > 1$, this justifies the *a priori* assumption (2.9).

The estimate (2.10) is optimal in terms of the space $W_{0,h}^k$, which is important since the main cost for the scheme is in the Poisson solver in $W_{0,h}^k$. The vorticity estimate in (2.10) is, however, suboptimal with respect to the space V_h^k . If we use $W_{0,h}^{k+1}$ instead for the stream function and use the upwind flux (1.10), then a more detailed analysis will produce an order $O(h^{k+1/2})$ for the error in ω ; see [13, 21] for details. However, we do not recommend this choice in practice, as the increase of half-order accuracy is obtained with the price of one degree higher polynomials in the most expensive part of the algorithm, namely the Poisson solver. In our numerical experiments in Section 5, we observe that close to $(k + 1)$ th order of accuracy is generally achieved when k th degree polynomials are used in both the discontinuous space for ω and the continuous space for ψ , both for uniform and for non-uniform meshes.

3. THE SCHEME FOR THE NAVIER–STOKES EQUATIONS

For the Navier–Stokes equations (1.3), there are two additional ingredients that require our attention:

1. The viscous terms cannot be directly implemented in the discontinuous space V_h^k . Instead, the stress tensor is first obtained locally using the same discontinuous Galerkin framework.

2. Vorticity boundary values are not known physically. We obtain vorticity boundary conditions locally from the stream function using the kinematic relation in (1.3).

We use the same finite element spaces V_h^k and $W_{0,h}^k$ defined in (1.5) for the vorticity and stream function, respectively. Denote by $V_{0,h}^k$ the subspace of V_h^k with zero value at the boundary. Let W_h^k be the finite element space extended from $W_{0,h}^k$ with general non-zero values at the boundary. The numerical method now becomes

$$\begin{aligned} & \langle \partial_t \omega_h v \rangle_K - \langle \omega_h \mathbf{u}_h \cdot \nabla v \rangle_K + \sum_{e \in \partial K} \langle \mathbf{u}_h \cdot \mathbf{n} \widehat{\omega}_h v^- \rangle_e \\ & = -\langle \sigma_h \cdot \nabla v \rangle_K + \sum_{e \in \partial K} \langle \widetilde{\sigma}_h \cdot \mathbf{n} v^- \rangle_e, \quad \forall v \in V_{0,h}^k. \end{aligned} \quad (3.1)$$

Notice that the test function is now in $V_{0,h}^k$, (see [23]), and the stress tensor $\sigma_h \in (V_h^k)^2$ is obtained from the vorticity ω_h by the same discontinuous Galerkin framework:

$$\operatorname{Re} \langle \sigma_h \mathbf{v} \rangle_K = -\langle \omega_h \nabla \cdot \mathbf{v} \rangle_K + \sum_{e \in \partial K} \langle \widetilde{\omega}_h \mathbf{v}^- \cdot \mathbf{n} \rangle_e, \quad \forall \mathbf{v} \in (V_h^k)^2. \quad (3.2)$$

We remark that (3.2) gives a local solution for the stress tensor σ_h , given the vorticity ω_h ; neither a global inversion nor a global storage is needed. The fluxes $\tilde{\sigma}_h$ and $\tilde{\omega}_h$ can be chosen as central averages

$$\tilde{\sigma}_h = \frac{1}{2}(\sigma_h^- + \sigma_h^+), \quad \tilde{\omega}_h = \frac{1}{2}(\omega_h^- + \omega_h^+) \quad (3.3)$$

or better still, as alternate one-sided fluxes, namely, at each edge e with an arbitrarily fixed orientation, one of $\tilde{\sigma}_h$ and $\tilde{\omega}_h$ is taken as the left value and the other taken as the right value. It can be verified that, for $k=0$ and a rectangular triangulation, the central fluxes (3.3) produce a wide stencil central approximation to the second derivatives (ω_{i-2} , ω_i and ω_{i+2} are used for approximating ω_{xx}), while the alternate one-sided fluxes produce a compact stencil central approximation (ω_{i-1} , ω_i and ω_{i+1} are used for approximating ω_{xx}). Also, numerical and theoretical evidence show that the alternate one-sided fluxes produce more accurate results [13]. In this paper we use only the alternate one-sided fluxes for the viscous terms.

For periodic boundary conditions, the scheme is now well defined. For the non-periodic case, we advocate using the approach in [23]. Although the basic idea in computing the vorticity boundary condition is similar to that of the standard finite element method in [23], there is some difference due to the fact that the approximation space for the vorticity in the discontinuous Galerkin method is large than that in a continuous finite element method. We outline the detailed steps here for completeness.

Since (3.1) is treated explicitly and the local discontinuous Galerkin method is used, ω_h^{n+1} in the interior elements can be directly computed. However, for the boundary elements, we need to compute ω_h^{n+1} in three steps.

First, for all test functions $v \in V_{0,h}^k$, we can compute the inner product $\langle \omega_h^{n+1} v \rangle_K$ directly from (3.1) thanks to the explicit time stepping.

Second, since $W_{0,h}^k$ is a subspace of $V_{0,h}^k$, from (3.1), the inner products, $\langle \omega_h^{n+1} v \rangle_K$ for all $v \in W_{0,h}^k$ has already been computed. This is sufficient for obtaining the stream function ψ_h^{n+1} from

$$-\langle \nabla \psi_h^{n+1} \cdot \nabla \varphi \rangle = \langle \omega_h^{n+1} \varphi \rangle, \quad \forall \varphi \in W_{0,h}^k, \quad (3.4)$$

with the velocity field obtained from the stream function by

$$\mathbf{u}_h^{n+1} = \nabla^\perp \psi_h^{n+1}. \quad (3.5)$$

Finally, we are able to compute the vorticity ω_h^{n+1} at boundary elements directly from

$$\langle \omega_h^{n+1} \varphi \rangle = -\langle \nabla \psi_h^{n+1} \cdot \nabla \varphi \rangle$$

for all the test functions $\varphi \in V_h^k$ thanks to the fact that ψ_h^{n+1} has already been computed.

For problems with periodic boundary conditions, the formulation above admits the following stability results,

$$\frac{d}{dt} \|\omega_h\|^2 + 2\|\sigma_h\| \leq 0, \quad (3.6)$$

which in turn implies stability for the velocity field (2.3). The proof is similar to the Euler case; see [13] for the details. With the vorticity boundary condition mentioned above, we are unable to obtain a stability estimate. However, this type of vorticity boundary treatment for conventional finite difference and finite elements is stable; see [15, 23]. Without such a stability estimate, there is an issue about the uniqueness of the solution to the method of lines discretization with the boundary condition treatment mentioned above. However, the full discretized version works well in the numerical experiments; see Example 1 of the next section.

4. ACCURACY CHECK AND NUMERICAL EXAMPLES

We implement our method on triangulations based on rectangles. When a P^k/Q^k result is referred to it is obtained with P^k elements for the vorticity ω and Q^k elements for the stream function ψ , where Q^k refers to the space of tensor products of 1D polynomials of degree up to k . We remark that some of the theoretical results in previous sections do not apply for these choices of mesh and spaces. For example, Q^k elements should also be used for the vorticity ω for the exact energy conservation (2.4) to hold; however, to save cost we use P^k elements for the vorticity ω instead. Energy stability (2.3) and enstrophy stability (2.2) still hold in this case. We have used both the upwind flux (1.10) and the (global) Lax–Friedrichs flux (1.11) for the calculations; however, we will only show the results obtained with the Lax–Friedrichs flux to save space. The time discretization is by the third-order positive Runge–Kutta methods in [27].

EXAMPLE 1. This example is used to check the accuracy of our schemes, both for the Euler equations (1.1) and for the Navier–Stokes equations (1.3) with $\text{Re} = 100$, for both the periodic and the Dirichlet boundary conditions, and with both a uniform mesh and a non-uniform mesh. The Dirichlet boundary conditions use the data taken from the exact solution. The non-uniform mesh is obtained by alternating between $0.9\Delta x$ and $1.1\Delta x$ for the mesh sizes in the x direction; similarly for the mesh sizes in the y direction. The initial condition is taken as

$$\omega(x, y, 0) = -2 \sin(x) \sin(y), \quad (4.1)$$

which was used in [5]. The exact solution for this case is known:

$$\omega(x, y, t) = -2 \sin(x) \sin(y) e^{-2t/\text{Re}}. \quad (4.2)$$

We use the domain $[0, 2\pi] \times [0, 2\pi]$ for the periodic case and $[0, \pi] \times [0, \pi]$ for the Dirichlet case and compute the errors at $t = 2$ for the periodic case and at $t = 1$ for the Dirichlet case. We list in Tables 5.1 (uniform mesh) and 5.2 (non-uniform mesh) the L_1 and L_∞ errors, at $t = 2$, measured at the center of the cells, for the periodic boundary conditions. Tables 5.3 (uniform mesh) and 5.4 (non-uniform mesh) contain the results with the Dirichlet boundary conditions at $t = 1$. We remark that, because of the difference in the sizes of the domains of the periodic and Dirichlet cases, the errors with the same number of cells are of different values, but the orders of accuracy are similar. We have also computed the errors of the relevant derivatives at the centers of the cells, which help in giving us truly L_∞ errors throughout the domain. We will not show them to save space. To get an idea about the effect of the viscous terms on the time step restriction and CPU time, we point out that for P^3/Q^3

TABLE 5.1
Accuracy Test, Uniform Meshes, Periodic Boundary Conditions

Mesh	Euler				Navier–Stokes with Re = 100			
	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
	P^1/Q^1							
16^2	7.77E-03	—	1.80E-02	—	7.65E-03	—	1.82E-02	—
32^2	1.01E-03	2.94	2.46E-03	2.87	1.03E-03	2.89	2.55E-03	2.83
64^2	1.28E-04	2.99	3.14E-04	2.97	1.36E-04	2.92	3.44E-04	2.89
128^2	1.60E-05	3.00	3.94E-05	2.99	1.80E-05	2.92	4.63E-05	2.89
	P^2/Q^2							
16^2	6.26E-04	—	1.58E-03	—	2.06E-04	—	5.85E-04	—
32^2	5.52E-05	3.50	2.75E-04	2.52	1.37E-05	3.90	3.24E-05	4.17
64^2	4.82E-06	3.52	3.81E-05	2.85	2.40E-06	2.51	4.10E-06	2.98
128^2	4.04E-07	3.58	4.96E-06	2.94	4.05E-07	2.57	6.44E-07	2.67
	P^3/Q^3							
16^2	9.74E-05	—	2.31E-04	—	9.68E-05	—	2.33E-04	—
32^2	6.81E-06	3.84	1.67E-05	3.79	6.22E-06	3.96	1.50E-05	3.96
64^2	4.36E-07	3.96	1.05E-06	3.99	3.82E-07	4.02	9.25E-07	4.02
128^2	2.71E-08	4.01	6.59E-08	3.99	2.33E-08	4.04	5.70E-08	4.02

with a 16^2 mesh, the Navier–Stokes code takes about twice as many time steps and about three times as much CPU time as the Euler code to reach the same physical time. We have also made several runs where the periodic and non-periodic cases have the same physical domain, mesh, and physical time. The errors are very close, indicating that the boundary effect on accuracy is small.

TABLE 5.2
Accuracy Test, Non-uniform Meshes, Periodic Boundary Conditions

Mesh	Euler				Navier–Stokes with Re = 100			
	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
	P^1/Q^1							
16^2	8.49E-03	—	2.85E-02	—	7.77E-03	—	2.80E-02	—
32^2	1.44E-03	2.56	5.56E-03	2.36	1.16E-03	2.75	5.45E-03	2.36
64^2	2.81E-04	2.36	1.13E-03	2.29	2.17E-04	2.42	1.03E-03	2.40
128^2	5.90E-05	2.25	2.59E-04	2.13	4.13E-05	2.40	1.94E-04	2.41
	P^2/Q^2							
16^2	7.88E-04	—	2.77E-03	—	3.37E-04	—	1.18E-03	—
32^2	7.82E-05	3.33	4.11E-04	2.75	1.78E-05	4.24	6.40E-05	4.21
64^2	7.66E-06	3.35	5.15E-05	3.00	2.63E-06	2.76	6.97E-06	3.20
128^2	7.43E-07	3.37	6.11E-06	3.07	4.34E-07	2.60	1.00E-06	2.80
	P^3/Q^3							
16^2	1.03E-04	—	3.26E-04	—	1.01E-04	—	3.24E-04	—
32^2	7.18E-06	3.84	2.60E-05	3.65	6.52E-06	3.96	2.14E-05	3.92
64^2	4.60E-07	3.96	1.77E-06	3.88	4.01E-07	4.02	1.28E-06	4.06
128^2	2.86E-08	4.01	1.09E-07	4.02	2.44E-08	4.03	7.70E-08	4.06

TABLE 5.3
Accuracy Test, Uniform Meshes, Dirichlet Boundary Conditions

Mesh	Euler				Navier–Stokes with Re = 100			
	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
	P^1/Q^1							
16^2	5.92E-04	—	1.23E-03	—	5.75E-04	—	1.32E-03	—
32^2	8.19E-05	2.85	1.92E-04	2.68	7.52E-05	2.94	1.78E-04	2.89
64^2	1.06E-05	2.94	5.35E-05	1.84	9.63E-06	2.96	3.76E-05	2.25
128^2	1.35E-06	2.98	1.42E-05	1.92	1.25E-06	2.94	8.06E-06	2.22
	P^2/Q^2							
16^2	4.76E-05	—	2.57E-04	—	1.51E-05	—	4.05E-05	—
32^2	4.28E-06	3.47	3.57E-05	2.85	2.49E-06	2.60	6.09E-06	2.73
64^2	3.74E-07	3.52	4.65E-06	2.94	4.11E-07	2.60	9.42E-07	2.69
128^2	3.17E-08	3.56	5.92E-07	2.97	6.16E-08	2.74	1.34E-07	2.81
	P^3/Q^3							
16^2	6.80E-06	—	1.58E-05	—	6.34E-06	—	1.53E-05	—
32^2	4.22E-07	4.01	1.06E-06	3.90	3.90E-07	4.02	9.45E-07	4.02
64^2	2.66E-08	3.99	6.90E-08	3.94	2.38E-08	4.04	5.81E-08	4.02
128^2	1.66E-09	4.00	4.25E-09	4.02	1.46E-09	4.03	3.59E-09	4.02

We can clearly see from these tables that close to $(k + 1)$ th order of accuracy is generally achieved when k th degree polynomials are used in both the discontinuous space for ω and for the Poisson solver, both for the uniform and for the non-uniform meshes.

EXAMPLE 2. The double shear layer problem taken from [1]. We solve the Euler equation (1.1) in the domain $[0, 2\pi] \times [0, 2\pi]$ with a periodic boundary condition and an initial

TABLE 5.4
Accuracy Test, Non-uniform Meshes, Dirichlet Boundary Conditions

Mesh	Euler				Navier–Stokes with Re = 100			
	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
	P^1/Q^1							
16^2	1.12E-03	—	4.35E-03	—	9.93E-04	—	4.25E-03	—
32^2	2.44E-04	2.20	9.79E-04	2.15	1.95E-04	2.35	8.74E-04	2.28
64^2	5.61E-05	2.12	2.39E-04	2.04	3.90E-05	2.33	1.75E-04	2.32
128^2	1.36E-05	2.04	6.29E-05	1.92	7.76E-06	2.33	3.54E-05	2.31
	P^2/Q^2							
16^2	7.54E-05	—	3.31E-04	—	1.98E-05	—	6.63E-05	—
32^2	8.15E-06	3.21	4.33E-05	2.93	2.61E-06	2.93	7.03E-06	3.24
64^2	8.46E-07	3.27	5.35E-06	3.02	4.35E-07	2.59	1.02E-06	2.79
128^2	8.31E-08	3.35	6.56E-07	3.03	6.52E-08	2.74	1.49E-07	2.78
	P^3/Q^3							
16^2	7.17E-06	—	2.49E-05	—	6.65E-06	—	2.18E-05	—
32^2	4.46E-07	4.01	1.66E-06	3.91	4.09E-07	4.02	1.31E-06	4.06
64^2	2.80E-08	3.99	1.04E-07	3.99	2.50E-08	4.03	7.86E-08	4.06
128^2	1.75E-09	4.00	6.89E-09	3.92	1.53E-09	4.02	4.77E-09	4.04

condition

$$\omega(x, y, 0) = \begin{cases} \delta \cos(x) - \frac{1}{\rho} \operatorname{sech}^2((y - \pi/2)/\rho) & y \leq \pi \\ \delta \cos(x) + \frac{1}{\rho} \operatorname{sech}^2((3\pi/2 - y)/\rho) & y > \pi \end{cases}, \quad (4.3)$$

where we take $\rho = \pi/15$ and $\delta = 0.05$.

The solution quickly develops into roll-ups with smaller and smaller scales, so on any fixed grid the full resolution is lost eventually. We use fixed uniform meshes of 64×64 and 128×128 rectangles and perform the calculation up to $t = 8$. We plot the time history of the total energy (square of the L^2 norm of velocity \mathbf{u}) and the total enstrophy (square of the L^2 norm of vorticity ω) in Fig. 5.1, as well as contours of the vorticity ω at $t = 6$ in Fig. 5.2 and at $t = 8$ in Fig. 5.3 to show the resolution. We can see from Fig. 5.1 that the numerical dissipation decreases roughly in the order of P^1/Q^1 - 64^2 , P^1/Q^1 - 128^2 , P^2/Q^2 - 64^2 , P^3/Q^3 - 64^2 , P^2/Q^2 - 128^2 , and P^3/Q^3 - 128^2 . We remark that due to the dissipation from Runge–Kutta time discretization and the choice of finite element spaces, the total energy in Fig. 5.1, left, decays rather than stays at a constant as proved in (2.4). The decay rate of total energy or total enstrophy in Fig. 5.1 is an indication of the actual resolution of the schemes for the given mesh. The higher-order methods have better resolutions and in general the resolution is quite good judging from the contours. We remark that when the numerical viscosity becomes too small with higher-order methods, since the schemes are linear, numerical oscillations are unavoidable when resolution to sharp fronts is lost, leading to instability. This is common for all linear schemes. However, the discontinuous Galerkin method we use here is able to get stable solutions for much sharper fronts with the same mesh than central type finite difference or finite element methods.

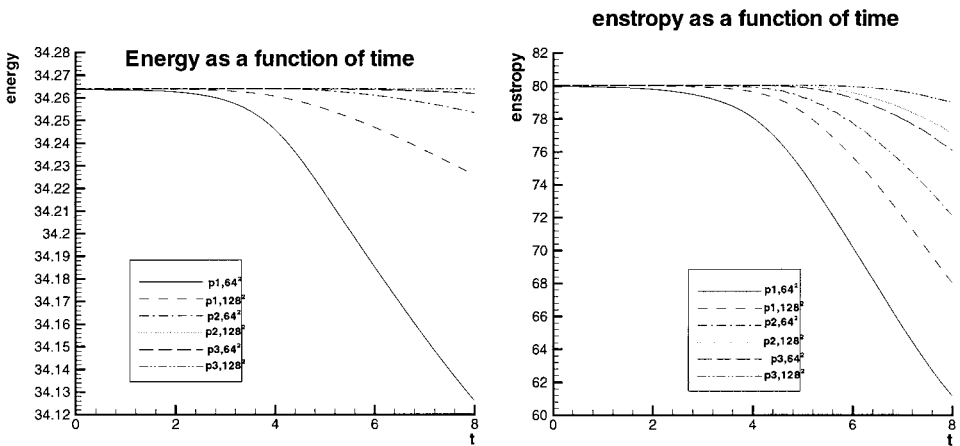


FIG. 5.1. The time history of energy (square of the L_2 norm of the velocity \mathbf{u}) and total enstrophy (square of the L^2 norm of vorticity ω). P^1/Q^1 with 64^2 mesh in solid line, P^1/Q^1 with 128^2 mesh in dashed line, P^2/Q^2 with 64^2 mesh in dash-dot line, P^2/Q^2 with 128^2 mesh in dotted line, P^3/Q^3 with 64^2 mesh in long dashed line, and P^3/Q^3 with 128^2 mesh in dash-dot-dot line.

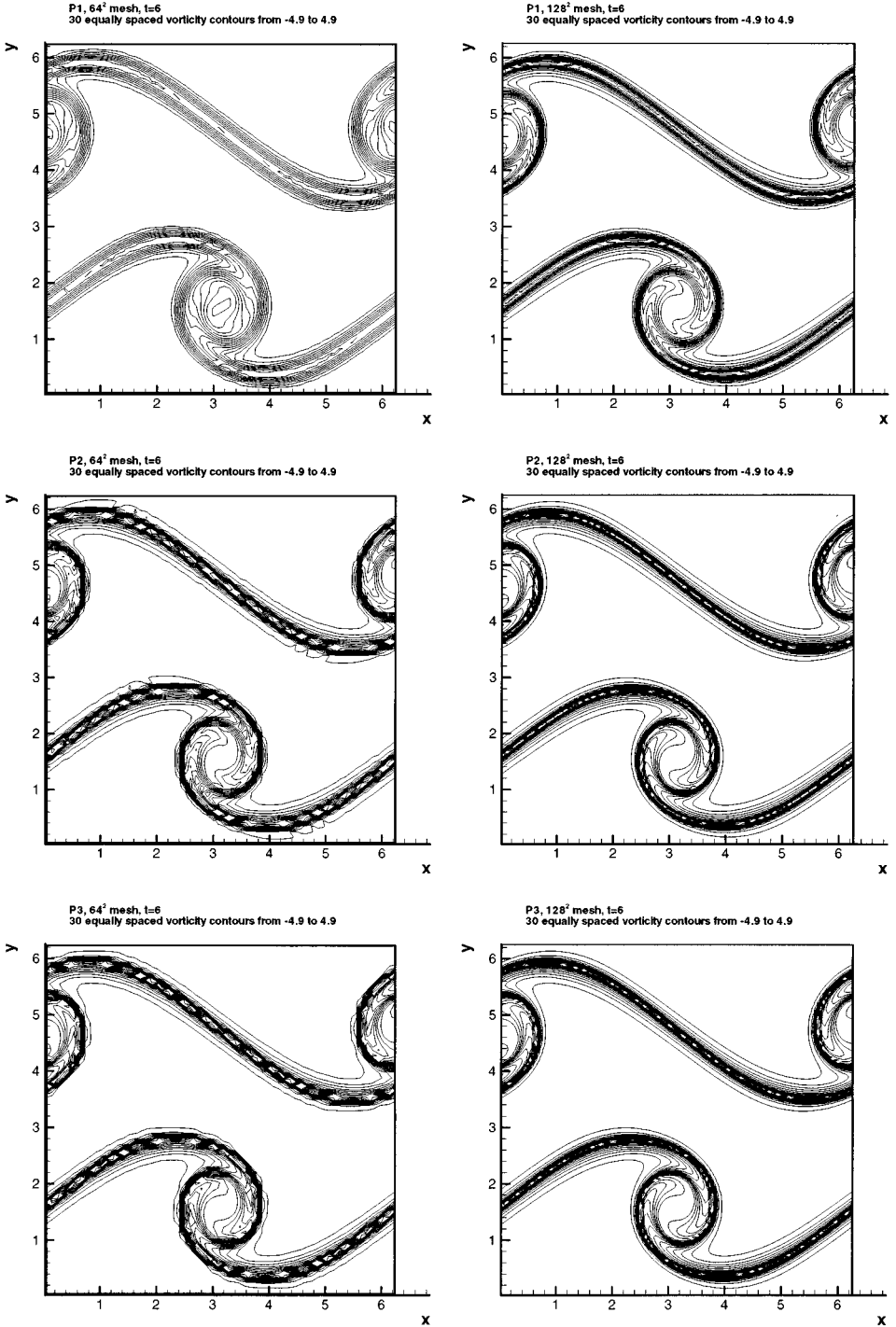


FIG. 5.2. Contour of vorticity ω at $t=6$. Thirty equally spaced contour lines between $\omega = -4.9$ and $\omega = 4.9$. Left: results with 64^2 mesh; right: results with 128^2 mesh. Top: P^1/Q^1 ; middle: P^2/Q^2 , bottom: P^3/Q^3 .

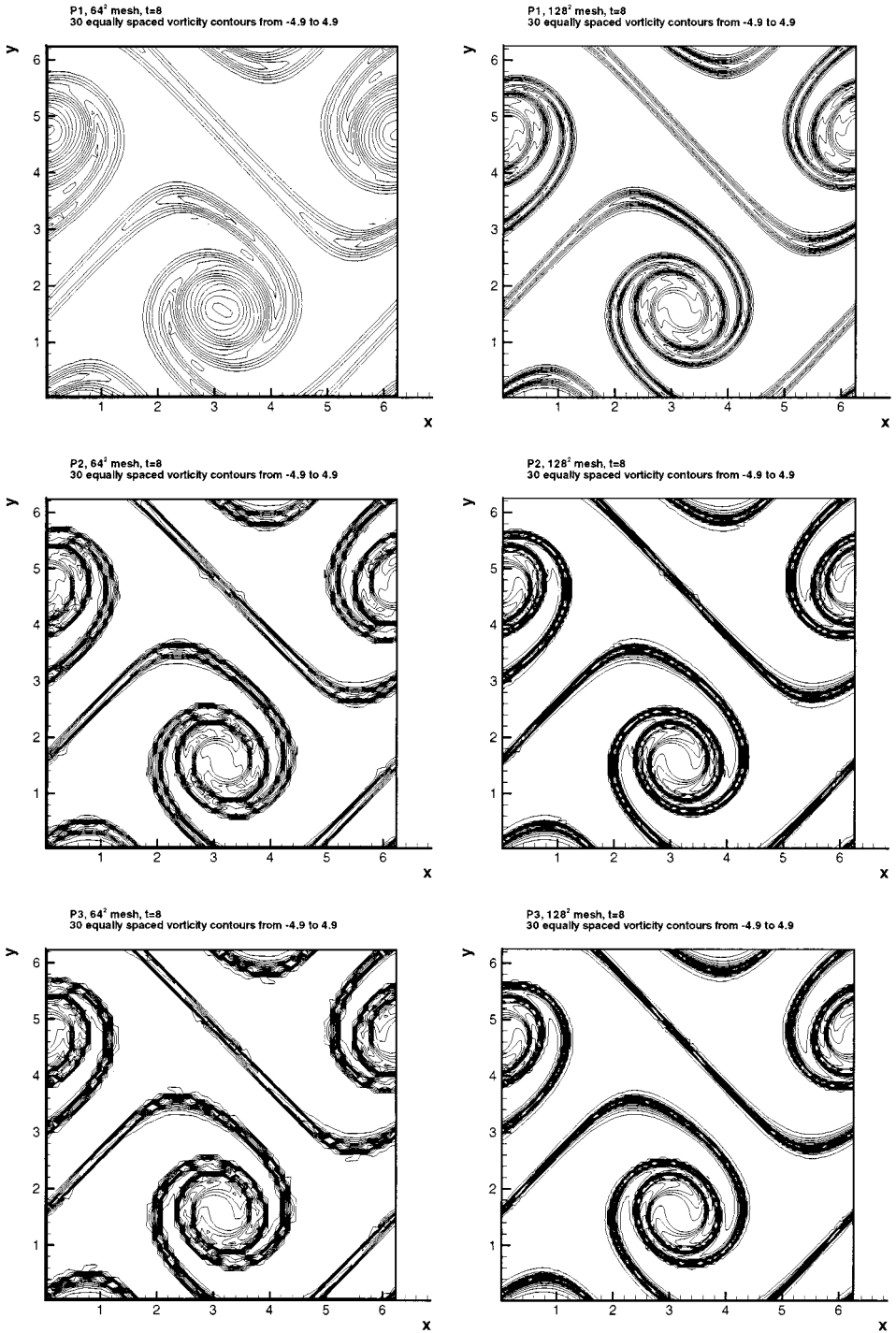


FIG. 5.3. Contour of vorticity ω at $t = 8$. Thirty equally spaced contour lines between $\omega = -4.9$ and $\omega = 4.9$. Left: results with 64^2 mesh; right: results with 128^2 mesh. Top: P^1/Q^1 ; middle: P^2/Q^2 , bottom: P^3/Q^3 .

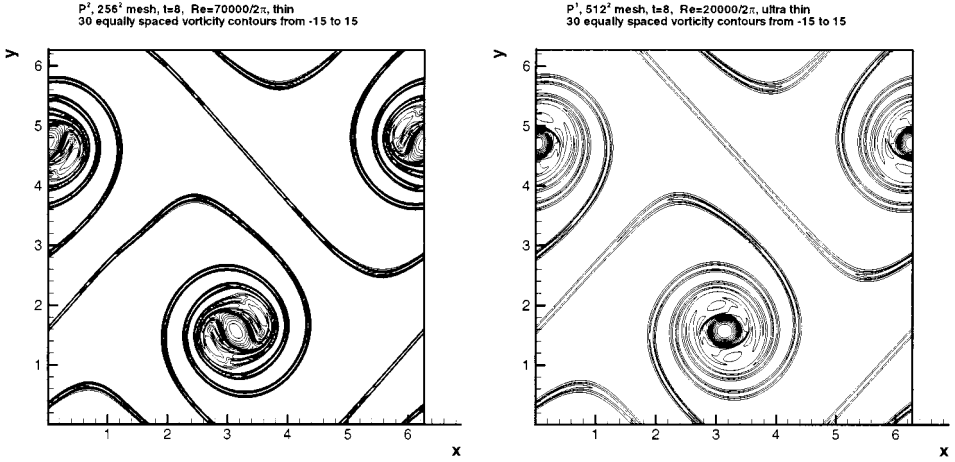


FIG. 5.4. Contour of vorticity ω at $t = 8$. Thirty equally spaced contour lines between $\omega = -15$ and $\omega = 15$. Left: results of the thin shear layer, $\rho = \pi/50$, $\text{Re} = 70,000/2\pi$, P^2/Q^2 method with a 256^2 mesh; right: results of the ultra thin shear layer, $\rho = \pi/100$, $\text{Re} = 20,000/2\pi$, P^1/Q^1 method with a 512^2 mesh.

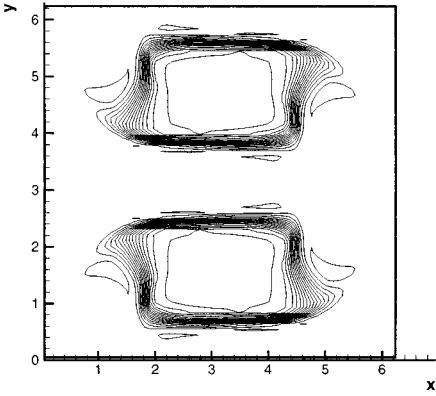
In [2], shear layers are distinguished between “thick” and “thin” ones, with the latter producing spurious vortices. The result above corresponds to the “thick” shear layer in [2]. We show in Fig. 5.4, left, a “thin” shear layer case as defined in [2], corresponding to $\rho = \pi/50$ with a Reynolds number $\text{Re} = 70,000/2\pi$, simulated with a uniform rectangular mesh of 256×256 cells with P^2/Q^2 method at $t = 8$. Notice that this is at a much higher Reynolds number than that used in [2], where the Reynolds number is $\text{Re} = 2000/2\pi$ and a second-order Godunov upwind projection method with 256×256 points produces spurious non-physical vortices. We also compute a ultrathin shear layer with $\rho = \pi/100$ with Reynolds number $\text{Re} = 20,000/2\pi$. The simulation result with a uniform rectangular mesh of 512×512 cells with P^1/Q^1 method at $t = 8$ is shown in Fig. 5.4, right. More extensive numerical resolution study for this example can be found in [24], where we explore thoroughly the resolution both for the “thick” and for the “thin” shear layers. In [24] we also plot the time history for the energy and enstrophy during a mesh refinement to show that the physical viscosity is dominating the numerics at such high Reynolds numbers, according to the decay of energy and enstrophy. This indicates that the built-in numerical viscosity of the methods is very small. We refer the reader to [24] for details. For a comparison with nonlinear ENO schemes, we refer to [16].

EXAMPLE 3. The vortex patch problem. We solve the Euler equation (1.1) in $[0, 2\pi] \times [0, 2\pi]$ with the initial condition

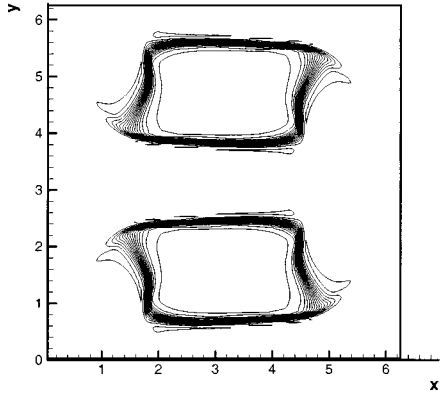
$$\omega(x, y, 0) = \begin{cases} -1, & \frac{\pi}{2} \leq x \leq \frac{3\pi}{2}, \quad \frac{\pi}{4} \leq y \leq \frac{3\pi}{4}; \\ 1, & \frac{\pi}{2} \leq x \leq \frac{3\pi}{2}, \quad \frac{5\pi}{4} \leq y \leq \frac{7\pi}{4}; \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

and periodic boundary conditions. The contour plots of vorticity ω , with 30 equally spaced contour lines between $\omega = -1.1$ and $\omega = 1.1$, are given in Fig. 5.5 for $t = 5$ and in Fig. 5.6 for $t = 10$. We can see that the scheme gives stable results for all runs, and higher-order schemes give better resolutions for vorticity.

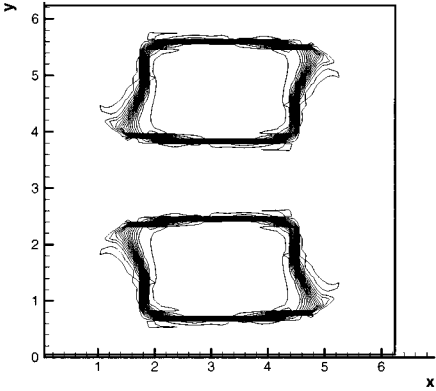
Vorticity at $t=5$, P_1 , 64^2 mesh
30 equally spaced contours from -1.1 to 1.1



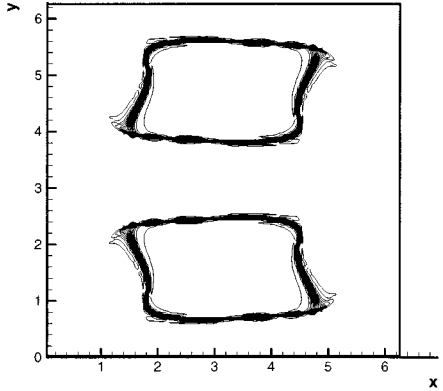
Vorticity at $t=5$, P_1 , 128^2 mesh
30 equally spaced contours from -1.1 to 1.1



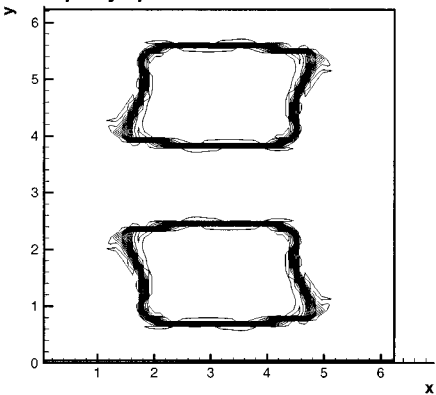
Vorticity at $t=5$, P_2 , 64^2 mesh
30 equally spaced contours from -1.1 to 1.1



Vorticity at $t=5$, P_2 , 128^2 mesh
30 equally spaced contours from -1.1 to 1.1



Vorticity at $t=5$, P_3 , 64^2 mesh
30 equally spaced contours from -1.1 to 1.1



Vorticity at $t=5$, P_3 , 128^2 mesh
30 equally spaced contours from -1.1 to 1.1

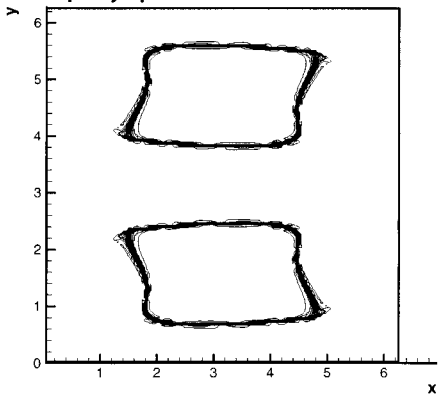


FIG. 5.5. Contour of vorticity ω at $t = 5$. Thirty equally spaced contour lines between $\omega = -1.1$ and $\omega = 1.1$. Left: results with 64^2 mesh; right: results with 128^2 mesh. Top: P^1/Q^1 ; middle: P^2/Q^2 , bottom: P^3/Q^3 .

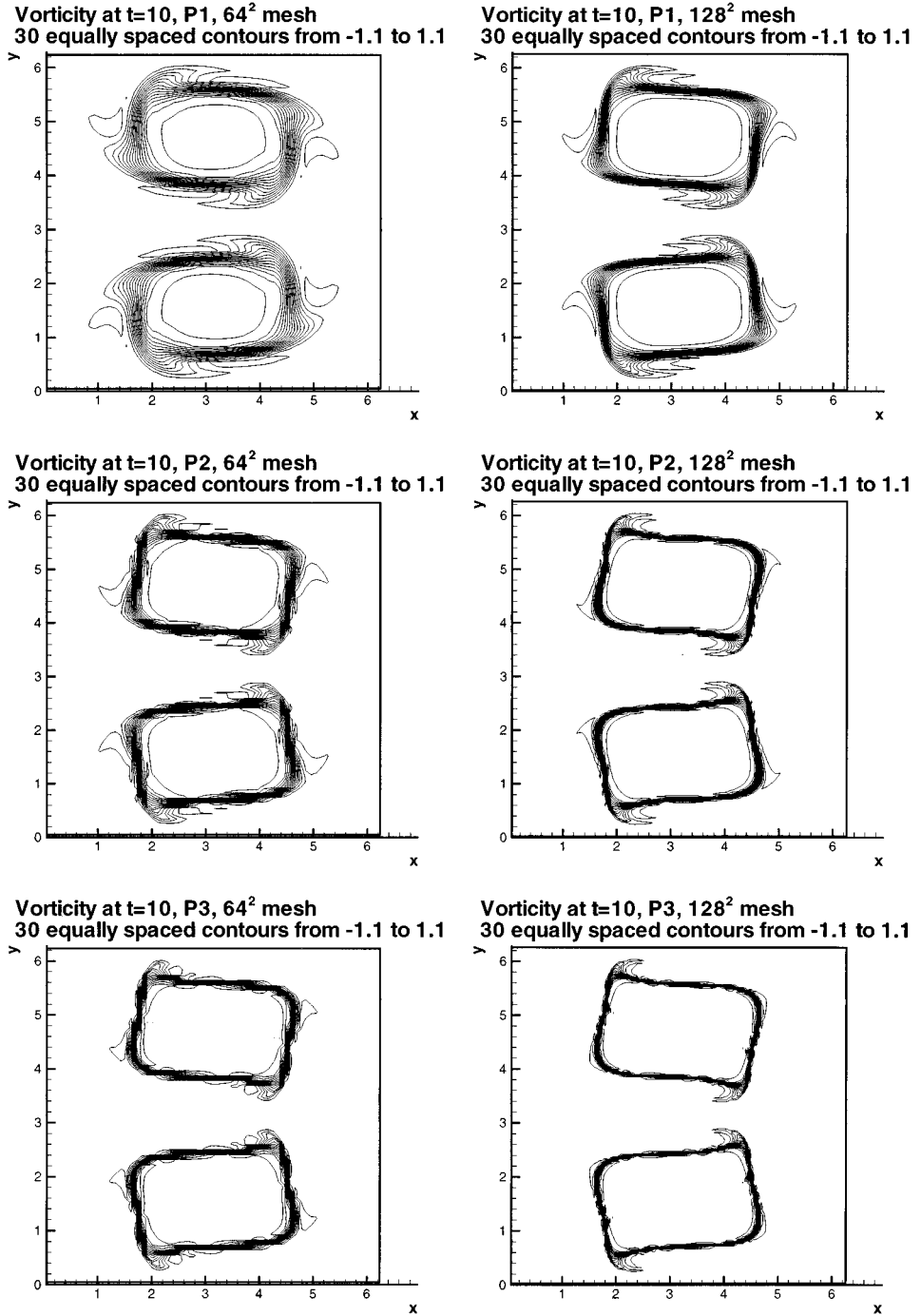


FIG. 5.6. Contour of vorticity ω at $t = 10$. Thirty equally spaced contour lines between $\omega = -1.1$ and $\omega = 1.1$. Left: results with 64^2 mesh; right: results with 128^2 mesh. Top: P^1/Q^1 ; middle: P^2/Q^2 , bottom: P^3/Q^3 .

5. CONCLUDING REMARKS

We have developed a high order discontinuous Galerkin method for the two-dimensional incompressible Euler and Navier–Stokes equations in the vorticity stream-function formulation, coupled with a standard continuous finite element solution of the Poisson equation for the stream function. A natural matching between the two finite element spaces allows us to obtain total energy conservation and total enstrophy stability. Numerical examples are shown to demonstrate the accuracy and resolution of the methods.

ACKNOWLEDGMENTS

This work was initialized when both authors were visiting the Mittag-Leffler Institute in Sweden. We thank the faculty and staff there for their warm hospitality.

REFERENCES

1. J. Bell, P. Colella, and H. Glaz, A second order projection method for the incompressible Navier–Stokes equations, *J. Comput. Phys.* **85**, 257 (1989).
2. D. L. Brown and M. Minion, Performances of under-resolved two dimensional incompressible flow simulations, *J. Comput. Phys.* **122**, 165 (1995).
3. R. Biswas, K. D. Devine, and J. Flaherty, Parallel, adaptive finite element methods for conservation laws, *Appl. Numer. Math.* **14**, 255 (1994).
4. A. Chorin, A numerical method for solving incompressible viscous flow problems, *J. Comput. Phys.* **2**, 12 (1967).
5. A. Chorin, Numerical solution of the Navier–Stokes equations, *Math. Comp.* **22**, 745 (1968).
6. P. Ciarlet, *The Finite Element Method for Elliptic Problems* (North-Holland, Amsterdam, 1975).
7. B. Cockburn, An introduction to the discontinuous Galerkin method for convection-dominated problems, in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, edited by B. Cockburn, C. Johnson, C.-W. Shu, and E. Tadmor, Lecture Notes in Mathematics, Vol. 1697 (Springer-Verlag, Berlin/New York, 1998).
8. B. Cockburn and C.-W. Shu, The Runge–Kutta local projection P^1 -discontinuous Galerkin method for scalar conservation laws, *Math. Model. Numer. Anal.* **25**, 337 (1991).
9. B. Cockburn and C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws. II. General framework, *Math. Comp.* **52**, 411 (1989).
10. B. Cockburn, S.-Y. Lin, and C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. III. One dimensional systems, *J. Comput. Phys.* **84**, 90 (1989).
11. B. Cockburn, S. Hou, and C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case, *Math. Comp.* **54**, 545 (1990).
12. B. Cockburn and C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws. V. Multidimensional systems, *J. Comput. Phys.* **141**, 199 (1998).
13. B. Cockburn and C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection diffusion systems, *SIAM J. Numer. Anal.* **35**, 2440 (1998).
14. W. E and J.-G. Liu, Vorticity boundary condition and related issues for finite difference schemes, *J. Comput. Phys.* **124**, 368 (1996).
15. W. E and J.-G. Liu, Essentially compact schemes for unsteady viscous incompressible flows, *J. Comput. Phys.* **126**, 122 (1996).
16. W. E and C.-W. Shu, A numerical resolution study of high order essentially nonoscillatory schemes applied to incompressible flow, *J. Comput. Phys.* **110**, 39 (1994).
17. V. Girault and P. A. Raviart, *Finite Element Methods for Navier–Stokes Equations, Theory and Algorithms* (Springer-Verlag, Berlin, 1986).

18. P. M. Gresho, Some interesting issues in incompressible fluid dynamics, both in the continuum and in numerical simulation, *Adv. Appl. Mech.* **28**, 45 (1992).
19. M. Gunzburger, *Finite Element Methods for Viscous Incompressible Flows* (Academic Press, Boston, 1989).
20. G. Jiang and C.-W. Shu, On cell entropy inequality for discontinuous Galerkin methods, *Math. Comp.* **62**, 531 (1994).
21. C. Johnson and J. Pitkäranta, An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation, *Math. Comp.* **46**, 1 (1986).
22. D. Levy and E. Tadmor, Non-oscillatory central schemes for the incompressible 2-D Euler equations, *Math. Res. Lett.* **4**, 1 (1997).
23. J.-G. Liu and W. E, Simple finite element method in vorticity formulation for incompressible flows, *Math. Comp.*, to appear .
24. J.-G. Liu and C.-W. Shu, A numerical example on the performance of high order discontinuous Galerkin method for 2D incompressible flows, in *Discontinuous Galerkin Methods: Theory, Computation and Applications*, B. Cockburn, G. Karniadakis, and C.-W. Shu, Editors, Lecture Notes in Computational Science and Engineering, volume 11, 2000, Springer-Verlag, Berlin/New York, pp. 369–374.
25. O. A. Pironneau, *Finite Element Methods for Fluids* (Wiley, New York, 1989).
26. L. Quartapelle, *Numerical Solution of the Incompressible Navier–Stokes Equations* (Birkhäuser, Berlin, 1993).
27. C.-W. Shu and S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.* **77**, 439 (1988).
28. R. Temam, *Navier–Stokes Equations, Theory and Numerical Analysis* (North-Holland, New York, 1979).