Gauge finite element method for incompressible flows

Weinan E[a,*,1] and Jian-Guo Liu[b,2]

[a] *Courant Institute of Mathematical Sciences, New York, NY, U.S.A.*
[b] *Temple University, Philadelphia, PA, U.S.A.*

SUMMARY

A finite element method for computing viscous incompressible flows based on the gauge formulation introduced in [Weinan E, Liu J-G. Gauge method for viscous incompressible flows. *Journal of Computational Physics* (submitted)] is presented. This formulation replaces the pressure by a gauge variable. This new gauge variable is a numerical tool and differs from the standard gauge variable that arises from decomposing a compressible velocity field. It has the advantage that an additional boundary condition can be assigned to the gauge variable, thus eliminating the issue of a pressure boundary condition associated with the original primitive variable formulation. The computational task is then reduced to solving standard heat and Poisson equations, which are approximated by straightforward, piecewise linear (or higher-order) finite elements. This method can achieve high-order accuracy at a cost comparable with that of solving standard heat and Poisson equations. It is naturally adapted to complex geometry and it is much simpler than traditional finite element methods for incompressible flows. Several numerical examples on both structured and unstructured grids are presented. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: gauge method; finite element method; incompressible flow

1. THE GAUGE FORMULATION OF THE NAVIER–STOKES EQUATION

We start with the incompressible Navier–Stokes equation

$$\begin{cases} \boldsymbol{u}_t + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p = \dfrac{1}{Re}\,\Delta\boldsymbol{u} \\ \nabla\cdot\boldsymbol{u} = 0 \end{cases} \tag{1.1}$$

* Correspondence to: Program in Applied and Computational Mathematics, Princeton University, Fine Hall, Washington Road, Princeton, NJ 08544-1000, U.S.A.
[1] E-mail: weinan@cims.nyu.edu
[2] E-mail: jliu@math.temple.edu

Received 2 October 1997
Revised 4 February 2000

on Ω, where $\boldsymbol{u} = (u, v, w)$ is the velocity and p is the pressure, with the simplest physical boundary condition:

$$\boldsymbol{u} = 0 \tag{1.2}$$

at $\Gamma = \partial\Omega$. The well-known difficulty with the numerical computation of viscous incompressible flows is the lack of an evolution equation for pressure. This is reflected in the additional boundary condition needed either explicitly or implicitly in various numerical methods for solving Equations (1.1) and (1.2). Much discussion has been devoted to this issue. We refer to Reference [1] for a review of various proposals. So far, it is generally agreed that projection method provides the best approach to tackling the issue of a pressure boundary condition.

A new approach was introduced in [Weinan E, Liu J-G. Gauge method for viscous incompressible flows. *Journal of Computational Physics* (submitted)], in which the Navier–Stokes equation was written in a form using a new variable ϕ. Instead of advecting $\boldsymbol{u}$, Reference [Weinan E, Liu J-G (submitted)] proposes to advect an auxiliary variable $\boldsymbol{a} = \boldsymbol{u} - \nabla\phi$. Indeed, the Navier–Stokes equation can be rewritten as

$$\begin{cases} \boldsymbol{a}_t - \dfrac{1}{Re}\Delta\boldsymbol{a} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = 0 \\ -\Delta\phi = \nabla\cdot\boldsymbol{a} \\ \boldsymbol{u} = \boldsymbol{a} + \nabla\phi \end{cases} \tag{1.3}$$

ϕ is referred to as the gauge variable, since writing $\boldsymbol{u} = \boldsymbol{a} + \nabla\phi$ is equivalent to choosing another gauge we view vorticity $\omega = \nabla \times \boldsymbol{u}$ as the analog of the magnetic field, and $\boldsymbol{u}$ as the analog of vector potential. Pressure has disappeared from the equations and can be recovered through

$$\phi_t - \frac{1}{Re}\Delta\phi = p \tag{1.4}$$

Replacing p by ϕ has the advantage that an additional boundary condition (as well as an initial condition) can be assigned to ϕ, since ϕ solves a parabolic equation with p as the right-hand side. Corresponding to Equation (1.2), we can either prescribe

$$\frac{\partial\phi}{\partial\boldsymbol{n}} = 0, \qquad \boldsymbol{a}\cdot\boldsymbol{n} = 0, \qquad \boldsymbol{a}\cdot\boldsymbol{\tau} = -\frac{\partial\phi}{\partial\boldsymbol{\tau}} \tag{1.5}$$

or

$$\phi = 0, \qquad \boldsymbol{a}\cdot\boldsymbol{n} = -\frac{\partial\phi}{\partial\boldsymbol{n}}, \qquad \boldsymbol{a}\cdot\boldsymbol{\tau} = 0 \tag{1.6}$$

on Γ. Here $\boldsymbol{\tau}$ is the unit vector in the tangential direction. We will call Equation (1.5) the Neumann formulation and Equation (1.6) the Dirichlet formulation. The boundary conditions for $\boldsymbol{a}$ come from Equation (1.2) and the relation $\boldsymbol{u} = \boldsymbol{a} + \nabla\phi$.

In principle, Equation (1.3), together with Equation (1.5) or (1.6), can be solved by any reasonable numerical method for the Poisson equation and heat equations. In Reference [Weinan E, Liu J-G (submitted)], we concentrated on finite difference methods. In this paper, we present the most simple-minded finite element method. We will discretize ϕ and $\boldsymbol{a}$ using the standard C^0 elements. As we will see later, the complexity of the resulting method is not very different from that of solving standard heat and Poisson equations.

Remark 1
The idea of reformulating the Navier–Stokes equation by introducing a gauge variable appeared in Reference [2] and was used in Reference [3] in the context of the vortex method. Their formulation was shown to be numerically unstable in Reference [4]. We also refer to References [5–7] for other related work.

2. DESCRIPTION OF THE FINITE ELEMENT METHOD

There are many different ways of formulating finite element methods for Equation (1.3). Instead of presenting general procedures, we concentrate on a special scheme, which is quite likely the simplest second-order method. We first discretize time using Crank–Nicolson

$$\frac{\boldsymbol{a}^{n+1} - \boldsymbol{a}^n}{\Delta t} + \nabla\cdot(\boldsymbol{u}^{n+1/2} \otimes \boldsymbol{u}^{n+1/2}) = \frac{1}{Re}\,\Delta\,\frac{\boldsymbol{a}^{n+1} + \boldsymbol{a}^n}{\mathbf{2}} \tag{2.1}$$

The convective term can be approximated using Adams–Bashfort

$$\boldsymbol{u}^{n+1/2} \otimes \boldsymbol{u}^{n+1/2} = \frac{3}{2}\,\boldsymbol{u}^n \otimes \boldsymbol{u}^n - \frac{1}{2}\,\boldsymbol{u}^{n-1} \otimes \boldsymbol{u}^{n-1} \tag{2.2}$$

Next we describe the spatial discretization. Here we work with the Dirichlet formulation (1.6) for two reasons. The first is that the Neumann formulation was studied quite extensively in Reference [Weinan E, Liu J-G (submitted)] in the context of finite difference methods. The second is that numerically solving the Dirichlet problem of the Poisson equation is much easier than solving the Neumann problem. At each time step, we have

$$\left(I - \frac{\Delta t}{2\,Re}\,\Delta\right)\boldsymbol{a}^{n+1} = \left(I + \frac{\Delta t}{2\,Re}\,\Delta\right)\boldsymbol{a}^n - \Delta t\nabla\cdot(\boldsymbol{u}^{n+1/2} \otimes \boldsymbol{u}^{n+1/2}) \tag{2.3}$$

with boundary condition

$$\boldsymbol{a}^{n+1}\cdot\boldsymbol{n} = -\left(2\,\frac{\partial\phi^n}{\partial\boldsymbol{n}} - \frac{\partial\phi^{n-1}}{\partial\boldsymbol{n}}\right), \qquad \boldsymbol{a}^{n+1}\cdot\boldsymbol{\tau} = 0 \tag{2.4}$$

and

$$-\Delta\phi^{n+1} = \nabla\cdot\boldsymbol{a}^{n+1}, \qquad \phi^{n+1}|_{\partial\Omega} = 0 \tag{2.5}$$

These equations are discretized in space using the standard piecewise linear finite element method.

Table I. Numerical results using the $P_1 - P_1$ method.

h	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h\|_{L_1}}{h^2}$	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h\|_{L_2}}{h^2}$	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h\|_{L_\infty}}{h^2}$
1/9	5.71	5.84	7.62
1/18	5.84	5.92	7.73
1/36	5.90	5.98	8.01

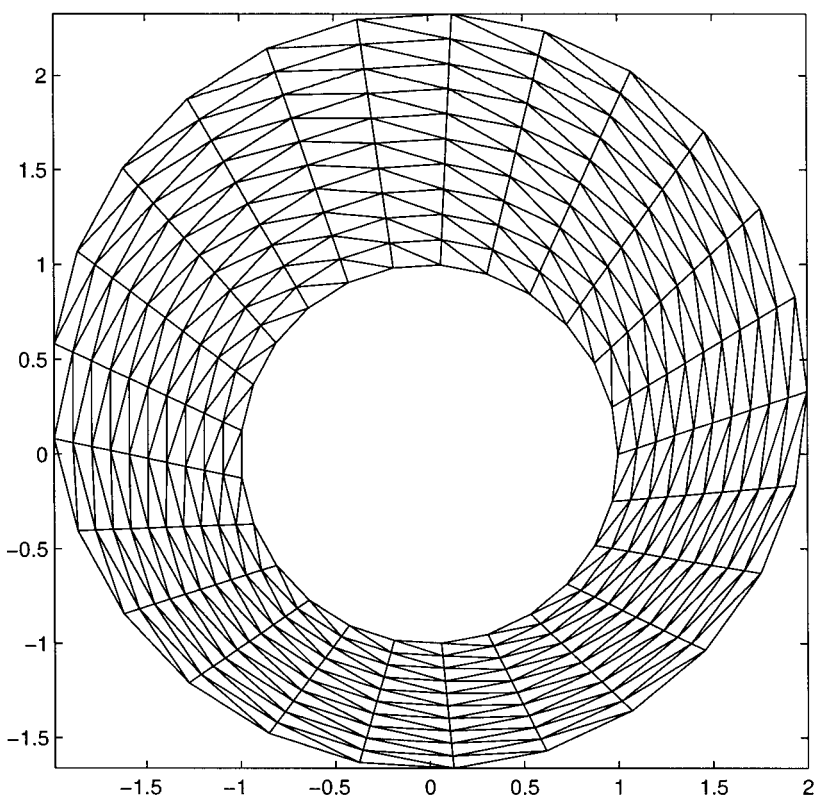

Figure 1. Numerical grid for the asymmetric annulus. Number of vertices = 500, number of elements = 275.

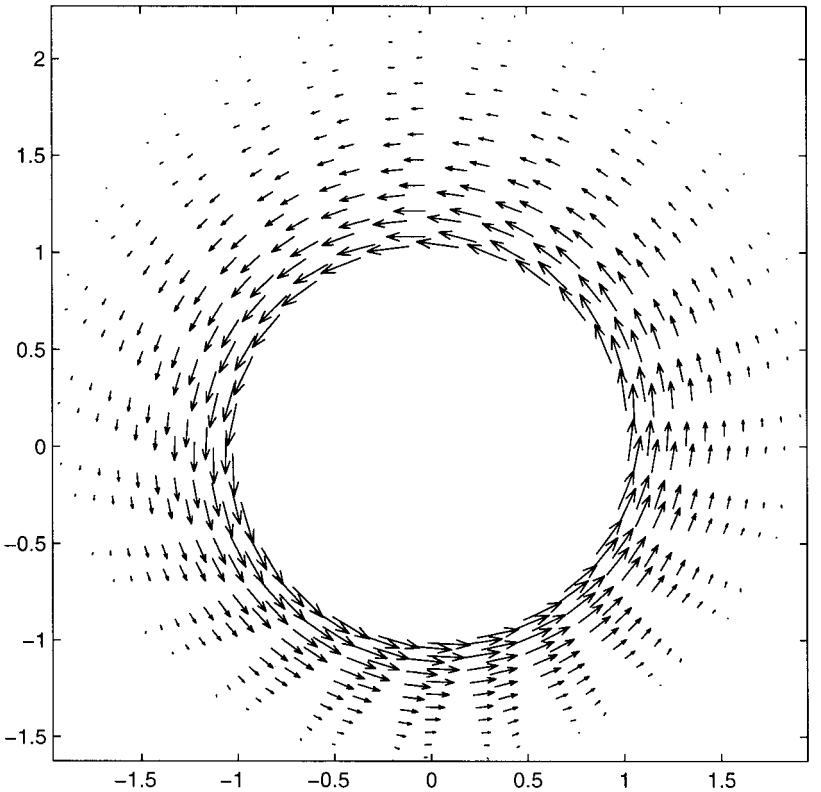

Figure 2. Computed steady state velocity field on the grid shown in Figure 1. Reynolds number = 100.

Having $\boldsymbol{a}^{n+1}$ and ϕ^{n+1}, we compute $\boldsymbol{u}^{n+1}$ by $\boldsymbol{u}^{n+1} = \boldsymbol{a}^{n+1} - \nabla\phi^{n+1}$. In general, $\boldsymbol{u}^{n+1}$ is discontinuous across cell boundaries. We therefore compute the convective term using a conservative form

$$((\boldsymbol{u}^n \cdot \nabla)\boldsymbol{u}^n, \psi) = -\int_\Omega (\nabla\psi)(\boldsymbol{u}^n \otimes \boldsymbol{u}^n)\, \mathrm{d}x \tag{2.6}$$

for all $\psi \in V_h \times V_h$, where V_h is the space of continuous piecewise linear functions that vanishes at the boundary.

The method described above is second-order accurate on regular grids, as shown in Table I, but degenerates to first-order accuracy on irregular grids. This is due to the involvement of gradient terms in the relation $\boldsymbol{u} = \boldsymbol{a} + \nabla\phi$. To achieve second-order accuracy on general irregular grids, we use piecewise quadratic polynomials to approximate ϕ. We call this the $P_1 - P_2$ method, and the method described earlier, the $P_1 - P_1$ method. There is an obvious generalization to $P_k - P_{k-1}$ methods for $k \leq 1$.

Remark 2

The main difference between the gauge method and the projection method [8–15] is that the gauge method requires solving *standard* Poisson equations for the gauge, whereas the projection method requires numerically performing the Helmholtz decomposition at the projection step. This amounts to solving the pressure Poisson equation via a mixed formulation. We demonstrated in Reference [Weinan E, Liu J-G (submitted)] that one cannot use standard Poisson solvers for the pressure Poisson equation because of difficulties at the boundary.

Remark 3

The time lag (also called vertical extrapolation) in Equation (2.4) is necessary to decouple the computation of $\boldsymbol{a}^{n+1}$ and ϕ^{n+1}. We showed in Reference [Weinan E, Liu J-G (submitted)] that this does not affect the stability and accuracy of the overall method.

Remark 4

Rigorous error estimates of this method were proved in Reference [16]. Basically, for smooth solutions this method achieves the expected accuracy. In principle, the regularity requirement

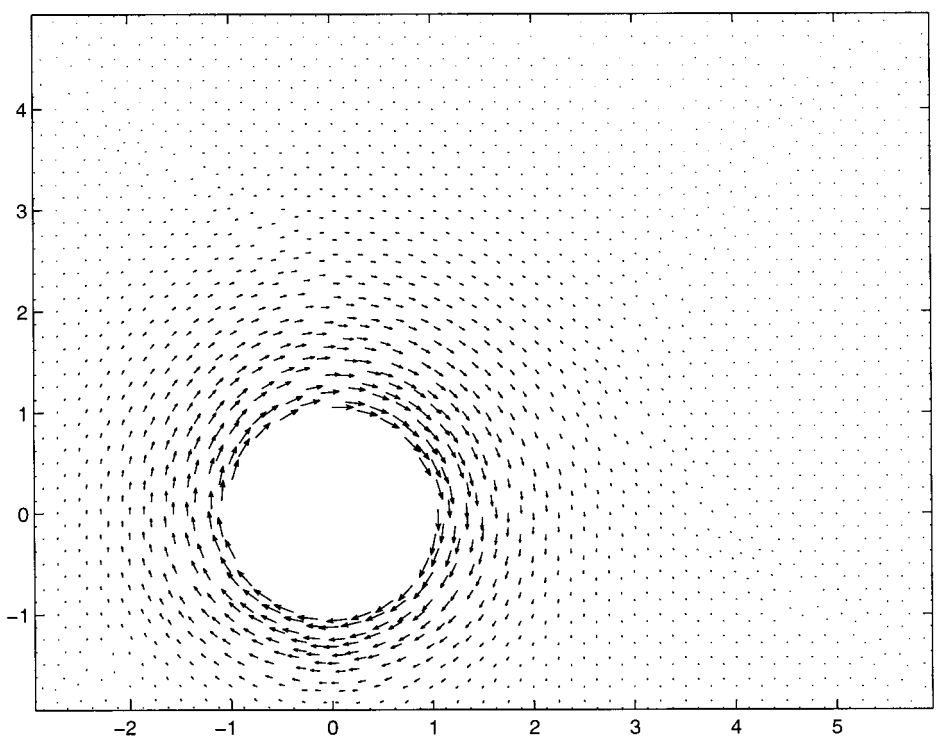

Figure 3. Computed velocity field at $t = 2.8$, with impulsive start-up of the circle. Other parameters are: number of elements = 1193, Reynolds number = 1.

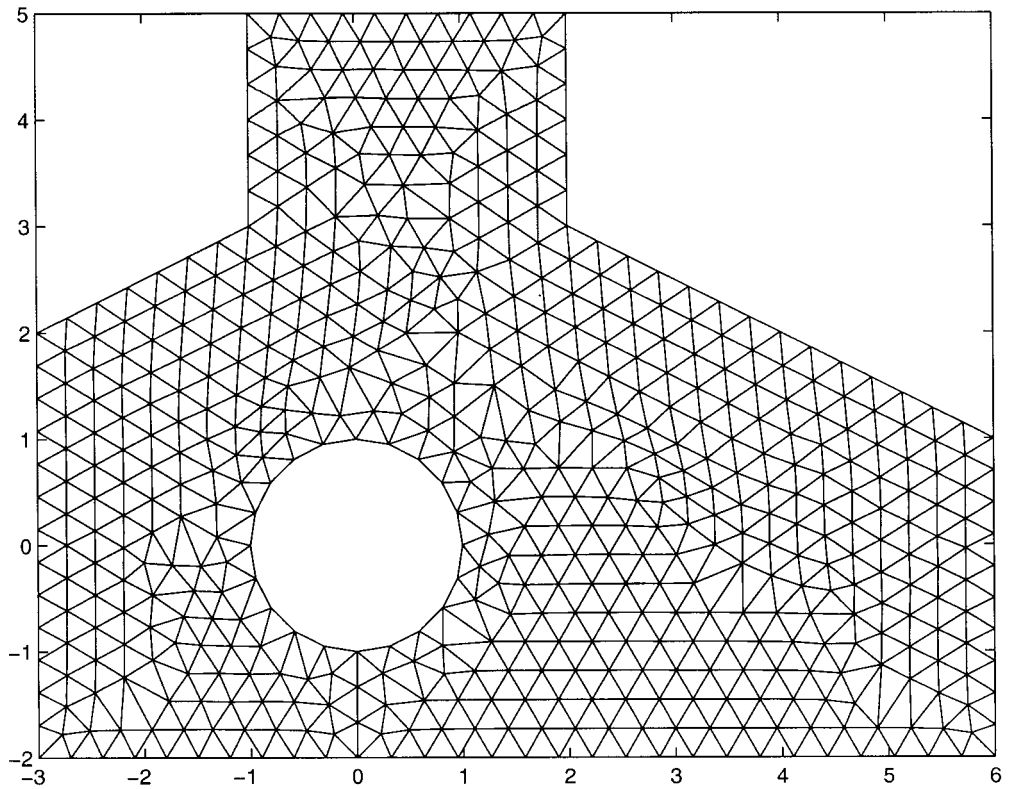

Figure 4. Numerical example for Example 3. Number of vertices = 1021, number of elements = 568.

on the solution can be relaxed, especially at $t = 0$, as in Reference [17]. However, this is a much more involved analysis and it is not yet done.

3. NUMERICAL RESULTS

3.1. Flow between concentric circles: an accuracy check

The method described above is expected to be second-order accurate. To check that this is actually the case, we performed a detailed accuracy check on a simple model problem: flow between two concentric circles. The radii of the inner and outer circle are 1 and 2 respectively. The outer circle remains stationary and the inner circle rotates in the counter-clockwise direction with unit speed. A standard regular polar co-ordinate grid is used. Table I shows the numerical results using the $P_1 - P_1$ method. We obtain second-order accuracy as a result of the regularity of the grid.

3.2. Other examples

We present several examples involving increasingly complex geometries and demonstrate the flexibility of the gauge finite element method for such geometries. In all these examples, the flow is driven by the impulsively started rotation of the inner circle. In the first example the computational domain is all annulus with the inner circle displaced. The computational grid is displayed in Figure 1. The computed velocity field using the $P_1 - P_1$ method is presented in Figure 2. In the second example, the computational domain is a rectangle with a circular hole at the lower left corner. The computed velocity field is shown in Figure 3. Finally, the numerical grid for the third example is shown in Figure 4, with the computed velocity field in Figure 5.

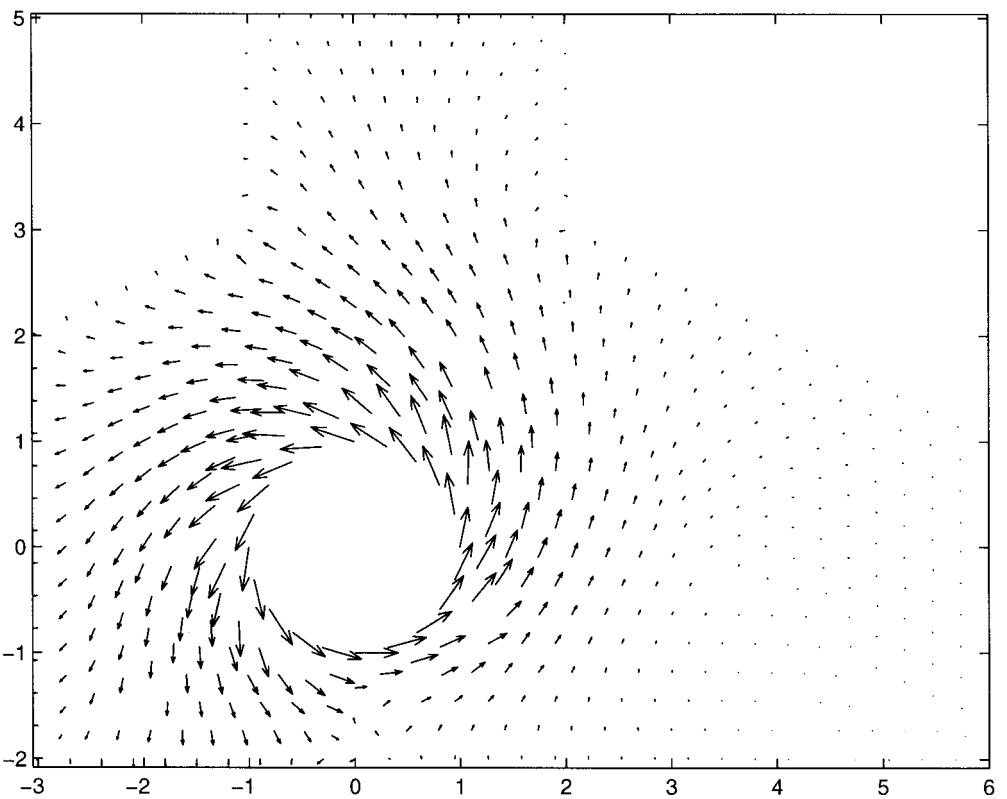

Figure 5. Computed velocity field at $t = 11$ on the grid shown in Figure 4. Reynolds number = 10.

4. CONCLUDING REMARKS

As we demonstrated in Reference [Weinan E, Liu J-G (submitted)], and here also, the main advantage of the gauge method is its simplicity and flexibility for computing low-Reynolds number flows. The gauge finite element method has the additional advantage of handling complicated geometries easily.

The method we presented here can be generalized in many different ways. Higher-order methods can be obtained by using higher-order backward differentiation formulas in time, and higher-order C^0 finite elements in space. Extension to three space dimensions is straightforward. At the present time, we have developed a program for computing two-dimensional viscous flows with any specified geometry and work on extending it to three dimensions is underway.

Because of its simplicity, the gauge method also opens up new ways of handling more complicated physical problems, such as visco-elastic flows. Work in this direction is also in progress.

ACKNOWLEDGMENTS

The authors thank Bjorn Engquist for suggesting the idea of finite element methods in the gauge formulation. They also thank M.K. Shin for his help in the initial stages of implementing the gauge finite element method. The work of Liu was supported in part by ONR grant N00014-96-1-1013 and NSF grant DMS-9805621.

REFERENCES

1. Gresho PM, Sani RL. On pressure boundary conditions for the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1987; **7**: 1111–1145.
2. Oseledets VI. On a new way of writing the Navier–Stokes equation: the Hamiltonian formalism. *Russian Mathematical Surveys* 1989; **44**: 210.
3. Buttke T. Velicity methods: Lagrangian numerical methods which preserve the Hamiltonian structure of incompressible fluid flow. In *Vortex Flows and Related Numerical Methods*, Beale JT, Cottet GH, Huberson S (eds). Kluwer: Dordrecht, 1993.
4. Weinan E, Liu J-C. Finite difference schemes for incompressible flows in the velocity impulse density formulation. *Journal of Computational Physics* 1997; **130**: 67–76.
5. Maddocks JH, Pego RL. An unconstrained Hamiltonian, formulation for incompressible fluid flow. *Communications in Mathematics and Physics* 1995; **170**: 207.
6. Russo G, Smereka P. Impulse formulation of the Euler equations: general properties and numerical methods. *Journal of Fluid Mechanics*, in press.
7. Summers D, Chorin AJ. Numerical vorticity creation based on impulse conservation. *Proceedings of the National Academy of Science* 1996; **93**: 1881–1885.
8. Bell JB, Colella P, Glaz HM. A second-order projection method for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 1989; **85**: 257–283.
9. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computations* 1968; **22**: 745–762.
10. Weinan E, Liu J-C. Projection method I: convergence and numerical boundary layers. *SIAM Journal of Numerical Analysis* 1995; **32**: 1017–1057.
11. Weinan E, Liu J-C. Projection method II: Godunov–Ryabenki analysis. *SIAM Journal of Numerical Analysis* 1996; **33**: 1597–1621.
12. Fortin M, Peyret R, Temam R. *Journal de Mecanique* 1971; **10**: 357–390.
13. Kim J, Moin P. Application of a fractional-step method to incompressible Navier–Stokes equations. *Journal of Computational Physics* 1985; **59**: 308–323.
14. Temam R. Sur l'approximation de la solution des equations de Navier–Stokes par to méthode des fractionnarires II. *Archives of Rational Mechanical Analysis* 1969; **33**: 377–385.

15. van Kan J. A second-order accurate pressure-correction scheme for viscous incompressible flow. *SIAM Journal of Science and Statistical Computing* 1986; **7**: 870–891.
16. Wang C, Liu J-G. Convergence of the gauge method for incompressible flow. *Mathematics of Computations*, in press.
17. Heywood J, Rannacher R. Finite element approximations of the non-stationary Navier–Stokes problem, part III. Smoothing property and higher-order error estimates for spatial discretizations. *SIAM Journal of Numerical Analysis* 1988; **25**: 490–512.

