SIAM J. SCI. COMPUT.
Vol. 28, No. 5, pp. 1675–1693
© 2006 Society for Industrial and Applied Mathematics

ON MASS-CONSERVING LEAST-SQUARES METHODS*

J. J. HEYS†, E. LEE‡, T. A. MANTEUFFEL‡, AND S. F. MCCORMICK‡

Abstract. Least-squares variational methods have several practical and theoretical advantages for solving elliptic partial differential equations, including symmetric positive definite discrete operators and a sharp error measure. One of the potential drawbacks, especially in three dimensions, is that mass conservation is achieved only in a least-squares sense, and underresolved solutions are especially vulnerable to poor conservation. For the stationary Navier–Stokes equations, which are typically rewritten as a larger system of first-order equations, the loss of mass in the approximate solution is strongly dependent upon the boundary conditions used. A new first-order system reformulation of the Navier–Stokes equations is presented that admits a wider range of mass-conserving boundary conditions. This new formulation is shown to provide both excellent mass conservation and excellent algebraic multigrid performance for three different problems, a square channel, backward-facing step, and branching tubes with two generations of bifurcations.

Key words. Navier–Stokes, conservation, finite elements, least squares, multigrid

AMS subject classifications. 76D05, 65F10, 65N55

DOI. 10.1137/050640928

1. Introduction.

Least-squares variational principles have been repeatedly recognized as a new tool to solve elliptic partial differential equations [6]. For the Stokes and Navier–Stokes equations in particular, a number of advantages are present [3, 4, 24], including the following:

- the weak formulations are associated with a minimization problem, resulting in *significant* computational advantages;
- the finite element spaces for each variable may be chosen independently because the inf-sup (or LBB) stability condition does not apply; and
- the functional provides a sharp, local error measure at no additional computational cost.

However, these benefits do not come without some drawbacks. In our experience, the following two are the most significant:

- when popular C^0 finite elements are used, the number of dependent variables typically increases over the original formulation (by a factor of 2 to 3 in many cases); and
- a lack of conservation of mass resulting in "ugly" solutions.

The computational advantages associated with least-squares finite element methods (LSFEM) are usually sufficient to overcome the first drawback, especially as problems become larger due to the scalability of multigrid solvers [9], which are often used in

*Received by the editors September 22, 2005; accepted for publication (in revised form) May 31, 2006; published electronically October 6, 2006. This work was sponsored by the Department of Energy under grants DE-FC02-01ER25479 and DE-FG03-99ER25217, Lawrence Livermore National Laboratory under contract B541045, Sandia National Laboratory under contract 15268, and the National Science Foundation under grants DMS-0410318. Computer time was provided by NSF ARI grant CDA-9601817, NSF MRI grant CNS-0420873, NASA AIST grant NAG2-1646, DOE SciDAC grant DE-FG02-04ER63870, NSF sponsorship of the National Center for Atmospheric Research, and a grant from the IBM Shared University Research (SUR) program.

http://www.siam.org/journals/sisc/28-5/64092.html

†Chemical and Materials Engineering, Arizona State University, Tempe, AZ 85287 (heys@asu.edu).

‡Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309 (eunjung@colorado.edu, tmanteuf@colorado.edu, steven@boulder.colorado.edu).

conjunction with least-squares methods [12]. Further, in many cases, the additional variables are physically important quantities that must be calculated at some point, regardless of the discretization method being used. It is the second drawback, the lack of conservation of mass, that is the focus of this paper.

Mass conservation between any inflow surface, Γ_i, and an outflow surface, Γ_o, which are possibly connect with impermeable boundaries, is typically measured in terms of the fractional change of mass flow, defined as

$$\frac{\int_{\Gamma_i}(\mathbf{n}\cdot\mathbf{v})d\Gamma_i - \int_{\Gamma_o}(\mathbf{n}\cdot\mathbf{v})d\Gamma_o}{\int_{\Gamma_i}(\mathbf{n}\cdot\mathbf{v})d\Gamma_i},$$

where $\mathbf{v}$ is the velocity of an incompressible fluid. The mass conservation can be measured either globally over the entire domain or locally over smaller subdomains. If the velocity is completely specified over all boundaries, it must be set to satisfy the global mass-conservation constraint, and only local mass conservation will not be exactly satisfied. Published computational results [16] and our own experience show that when global mass conservation is enforced through boundary conditions, LSFEM give approximate solutions with acceptable local mass conservation. However, a common flow problem involves the velocity being completely specified only along the inflow and wall regions but not completely specified along the outflow regions. In this case, the error in global mass conservation is approximately the accumulation of errors in local mass conservation between the inlet and outlet. It is these types of flow problems, where the outlet velocity is not fully specified, that are the subject of this paper. The goal is to improve global mass conservation while maintaining the strengths of LSFEM, specifically the multigrid convergence rates. This is in contrast to many of the existing methods to improve mass conservation, which result in slower convergence rates for standard multigrid.

Mass conservation in Galerkin formulations of the Navier–Stokes equations is enforced by the divergence-free constraint on the velocity field, which essentially uses pressure as a Lagrange multiplier. With least-squares formulations, the objective is to minimize, in a least-squares sense, the value of a functional. For example, the stationary Navier–Stokes equations in dimensionless form are

$$-Re\,(\mathbf{v}\cdot\nabla\mathbf{v}) - \nabla p + \Delta\mathbf{v} = 0 \quad \text{in } \Omega, \tag{1.1}$$

$$\nabla\cdot\mathbf{v} = 0 \quad \text{in } \Omega, \tag{1.2}$$

where p is the pressure normalized by viscosity, Re is the Reynolds number, and $\mathbf{v} = (v_x, v_y, v_z)$ is the dimensionless velocity. Equations (1.1) and (1.2) may be combined in a least-squares sense into a single functional given by

$$G(\mathbf{v},p) := \alpha\| - Re\,(\mathbf{v}\cdot\nabla\mathbf{v}) - \nabla p + \Delta\mathbf{v}\|^2_{0,\Omega} + \beta\|\nabla\cdot\mathbf{v}\|^2_{0,\Omega}\,. \tag{1.3}$$

In practice, $G(\mathbf{v},p)$ would never be minimized because it exhibits behavior similar to a biharmonic, but it is shown to illustrate that there is always a balance between equations. The first term in G represents conservation of momentum, and the second represents conservation of mass. One of the potential strengths of least-squares methods is that, by adjusting the values of constants α and β, the error can be distributed between mass and momentum conservation in a controlled manner. However, changing the values of α and β almost always affects solver performance. For example, making β larger (i.e., less error in conservation of mass) can result in faster or slower multigrid convergence, depending on the Reynolds number, Re, and the actual

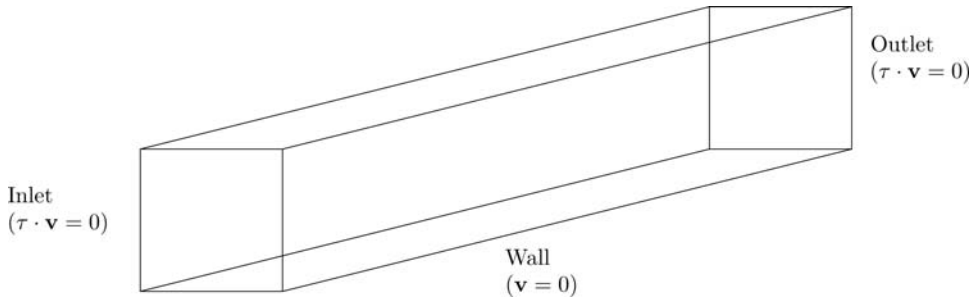

FIG. 1.1. *The test problem domain consisting of a brick with an aspect ratio of* 4:1:1.

functional used (G is not a computationally practical functional and is rarely used). The effects of weighting the functional are demonstrated later in this paper, but, even though this can be beneficial in some situations, it should not be considered a universal solution [8].

The focus in this paper is on stationary or steady-state Navier–Stokes equations, although some general statements can be made about the extension to nonstationary Navier–Stokes using implicit time stepping. The additional terms in the functional due to the discrete approximation of the time derivative makes the diagonal of the operator larger and, hence, it is usually more efficiently solved by a multigrid algorithm. Further acceleration is realized by the fact that the previous time step solution provides a good initial guess to the new time step. As a result, the methods proposed in this paper are also good methods for solving the nonstationary problem if implicit time stepping is desired.

1.1. Model problem and loss of mass conservation. We introduce a simple test problem to illustrate the mass-conservation issue associated with existing least-squares methods. The test problem was chosen to be as simple as possible while still capturing the key mass-conservation issues. Initially, the Reynolds number is set to zero because the important mass conservation issues exist in Stokes flow as well as laminar Navier–Stokes flow. Later, results using nonzero Reynolds numbers are given. While it is possible to demonstrate some of the mass-conservation issues in two dimensions, Pontaza and Reddy [24] recently showed that using higher-order finite elements largely alleviates the problems in two dimensions. Our own experiences agree with this finding, but, as we show, moving to higher-order elements is not as effective in addressing the mass-conservation issues in three dimensions. Therefore, the domain for the test problem is the simple "brick" shown in Figure 1.1 with an aspect ratio of 4:1:1. The long sides of the brick are walls upon which the no-slip boundary condition is imposed. Along the inlet and outlet, the tangential velocity is set to zero, and an additional boundary condition to be chosen later must also be set.

The functional, G, given by (1.3) has a higher regularity requirement and cannot be used with practical C^0 finite element spaces. Typically, the governing equations are transformed into an equivalent system of first-order equations. A number of different methods exist for generating a first-order system for the Navier–Stokes equations [3, 4]. One approach is based on defining a matrix of new variables, $\mathbf{V}$, equal to the gradient of the velocity vector. The first-order system for Navier–Stokes equations is then written as

$$\begin{aligned} &\mathbf{V} - \nabla \mathbf{v} = 0 && \text{in } \Omega, \\ &-Re(\mathbf{v} \cdot \mathbf{V}) - \nabla p + \nabla \cdot \mathbf{V} = 0 && \text{in } \Omega, \\ &\nabla \cdot \mathbf{v} = 0 && \text{in } \Omega. \end{aligned} \tag{1.4}$$

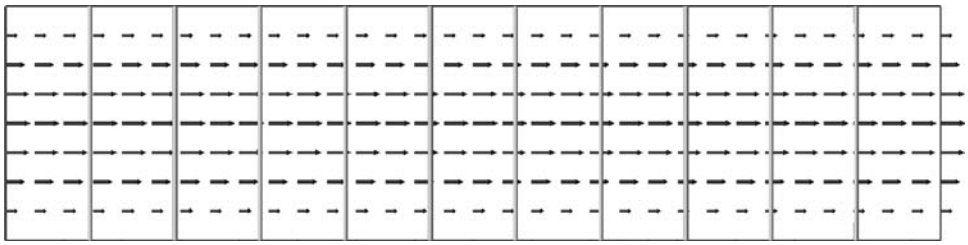

FIG. 1.2. *Cross-section of flow through the brick domain using the velocity-flux formulation of the Navier–Stokes equations. The pressure is set to* 1 *at the inlet and* 0 *at the outlet, and the contour bars represent evenly spaced pressure lines.*

Because $\mathbf{V}$ is the gradient of $\mathbf{v}$ and we are solving a minimization problem, we can augment the first-order system of equations by the following consistent equations:

$$\begin{aligned} \nabla(\mathrm{tr}(\mathbf{V})) &= 0 \quad \text{in } \Omega, \\ \nabla \times \mathbf{V} &= 0 \quad \text{in } \Omega, \end{aligned} \tag{1.5}$$

where $\mathrm{tr}(\mathbf{V}) = V_{11} + V_{22} + V_{33}$. This first-order formulation of the Navier–Stokes equations is often called the "velocity-flux" formulation, and it has a number of theoretical and practical advantages, including optimal error estimates in the H^1 norm for each variable and optimal multigrid convergence estimates [7, 4]. The corresponding functional for the velocity-flux formulation is

$$\begin{aligned} G_{v.f.}(\mathbf{V},\mathbf{v},p) := \|\mathbf{V} - \nabla\mathbf{v}\|^2_{0,\Omega} + \| - Re\,(\mathbf{v}\cdot\mathbf{V}) - \nabla p + \nabla\cdot\mathbf{V}\|^2_{0,\Omega} \\ + \|\nabla\cdot\mathbf{v}\|^2_{0,\Omega} + \|\nabla(\mathrm{tr}(\mathbf{V}))\|^2_{0,\Omega} + \|\nabla\times\mathbf{V}\|^2_{0,\Omega}. \end{aligned} \tag{1.6}$$

To minimize the functional $G_{v.f.}$ over the test domain shown in Figure 1.1, additional boundary conditions must be defined along the inlet and outlet. A common boundary condition is to set the normal stress on the normal face (i.e., the natural boundary condition) equal to a constant. For this domain with no tangential velocity at the inlet and outlet, setting the normal stress to a constant is equivalent to setting the pressure to a constant. The results of setting the inlet pressure (i.e., the normal stress) to 1 along the inlet and 0 along the outlet are shown in Figure 1.2. To generate this result, a hexahedral (cubic) mesh of $32 \times 8 \times 8$ elements was used ($h = 1/8$), the Reynolds number was set to zero (i.e., Stokes flow), and a trilinear finite element basis was used. Even though Figure 1.2 only shows the central cross-section of the brick, it is clear that there is good mass conservation based on a visual inspection. In fact, integrating the velocity field along the inlet and outlet (or any other cross-section orthogonal to the flow) shows perfect mass conservation up to 5 digits. The value of the functional at the approximate solution is 4.0×10^{-3}. If we look at the size of different terms in the functional, the divergence term, $\nabla\cdot\mathbf{v}$, is very small at 5.8×10^{-9}, and the error is largely in the momentum term, $-Re\,(\mathbf{v}\cdot\mathbf{V}) - \nabla p + \nabla\cdot\mathbf{V}$, with a value of 1.6×10^{-3}.

At this point, the least-squares formulation appears to provide excellent mass conservation. However, a change in the inlet boundary condition can have a dramatic effect. For example, the result of replacing the natural inlet condition, $p = 1$, with an essential one, in which the normal velocity is set to $v_x = y(1-y)z(1-z)$, is shown in Figure 1.3. It is important to note that the inlet condition is not set to the steady flow profile, but it is a valid boundary condition. The total value of the functional

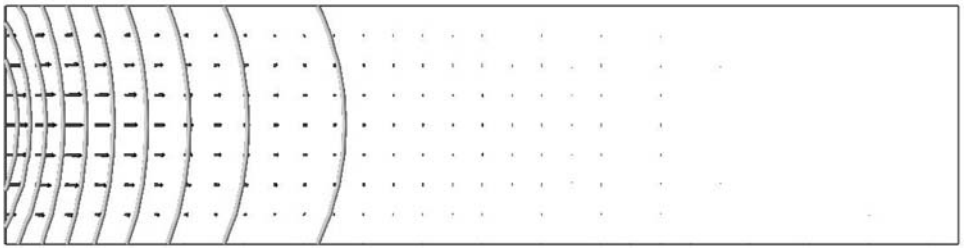

FIG. 1.3. *Cross-section of flow through the brick domain using the velocity-flux formulation of the Navier–Stokes equations. The normal velocity is set to* $v_x = y(1-y)z(1-z)$ *at the inlet and the pressure to* 0 *at the outlet, and the contour bars represent evenly spaced pressure lines.*

at the minimum is 6.0×10^{-3}, and the divergence term in the functional (5×10^{-4}), representing conservation of mass, is now nearly the same size as the momentum term (1×10^{-3}). The total flow rate into the domain is 2.7×10^{-2} and the outflow rate is 5.0×10^{-5}, a 99.8% loss of mass. The key to understanding this mass loss is to look at the pressure, which should be decreasing nearly linearly except for some small entrance effects near the inlet. Instead, the functional is a minimum when both the pressure and the flow rate are approximately zero for two-thirds of the domain lying downstream. Clearly, the functional is approximately zero for the downstream portion of the domain, which helps keep the functional small. The larger error at the entrance is not sufficient to make the overall functional value large. Remember, this method finds the solution that *minimizes* the functional over all possible solutions in the finite element space. In other words, if we do not like the solution, we asked the method to minimize a functional that does not capture our requirements for the solution.

1.2. Possible remedies. Several approaches can be taken to "fix" the conservation of mass problem illustrated in the previous example (i.e., Figure 1.3), including the following:

- reducing the mesh size, h;
- weighting the velocity-divergence term in the functional;
- using higher-order elements or some other finite element space (e.g., [10]); and
- using a different first-order system of equations that is still equivalent to the Navier–Stokes equations.

Other, more complex remedies can be used, including norms other than L^2 norms or using a FOSLL* approach [11], but only the approaches listed above are explored here. Below, each of these options is individually explored using the previous test problem. In most cases, mass conservation is improved at the cost of multigrid performance, making the remedy unacceptable.

Reducing the mesh size has the advantage that it should improve mass conservation and not make the problem worse. As the mesh size approaches zero, the approximate solution approaches the true solution at the expected rate, which depends on the finite element basis. However, because conservation of mass is currently far from being acceptable, significant refinement is required before the approximation becomes acceptable. For example, if the previous mesh ($32 \times 8 \times 8$) is refined to ($64 \times 16 \times 16$), the loss of mass is only reduced from 99.8% to 98.8%.

As mentioned previously, weighting the divergence of velocity term in the functional can also improve conservation [16]. If the divergence term in the velocity-flux

formulation (i.e., $G_{v.f.}$) is weighted by 100, the mass loss is reduced from 99.8% to 0.7%. However, the multigrid convergence factor, defined as

$$\rho \approx \sqrt[m]{\frac{\|res^{(m)}\|_2}{\|res^{(0)}\|_2}}, \tag{1.7}$$

where $res^{(m)}$ is the residual (or defect) after the mth V-cycle, is increased from 0.74, with a weight of 1.0, to 0.87, with a weight of 100.0. To put that change into perspective, reducing the residual norm by a factor of 1.0×10^{-8} requires 62 V-cycles when the convergence factor is 0.74, and 132 V-cycles when the convergence factor is 0.87, a doubling of the total computational cost. Further, in our experience, larger weights are required as the domain becomes more complex, especially domains with high aspect ratios. Multigrid methods that perform well on this class of problems exist, but require special elements and special relaxation techniques (cf. [2]). A similar idea to weighting the divergence term is to enforce it through a Lagrange multiplier strategy [13]. In this case, the operator is still symmetric, but the now indefinite operator is not effectively solved using a standard multigrid method.

It has been shown by both Pontaza and Reddy [24] and Deang and Gunzburger [16] that higher-order (or spectral) elements can produce very accurate results with least-squares formulations. This has been shown extensively in two dimensions, but it has not been demonstrated in three dimensions. Further, the computational costs associated with least-squares formulations in three dimensions with high-order elements can become prohibitive. For example, in the velocity-flux formulation there are 13 variables per node in three dimensions. Taking the tensor product of 3 eighth-order nodal basis functions results in an element with 729 nodes. The cost of storing a single element stiffness matrix with double precision variables approaches 1 gigabyte. If eighth-order basis functions are too expensive to store, one may ask what is the benefit of simply refining from trilinear elements to triquadratic elements. For the test problem above with the essential condition on velocity along the inlet, the mass loss is decreased from 99.8% to 64%, a significant improvement. However, even the improved mass conservation is still not acceptable, and it came at a increased computational cost (e.g., if the one-dimensional polynomial order is p, then computational costs scale as $O(p^6)$ if exact integration is used).

Related to the idea of using alternative finite element spaces is the idea of using reduced integration methods [21, 26]. Often, the use of reduced integration results in a collocation solution [6, 24]. The net effect of using reduced integration on most problems, including the test problem, is excellent conservation of mass, loss of the functional as an error measure because it is zero at the minimum, and poor multigrid performance. For example, when using reduced integration on the brick test problem, mass was conserved up to at least 5 digits of accuracy, but thousands of algebraic multigrid V-cycles were required for marginal convergence. In this case, a direct solver would have been much more effective, but our desire is to find a solution to the mass conservation problem that maintains the optimal computational scalability of standard multigrid.

The final approach we mention is to change the form of the functional (i.e., to use a different first-order system of equations). This approach is attractive due to our previous observation that if we do not like the approximate solution that minimizes the functional, then we are using the wrong functional. The most popular first-order system that is equivalent to the Navier–Stokes equations is the vorticity system, given

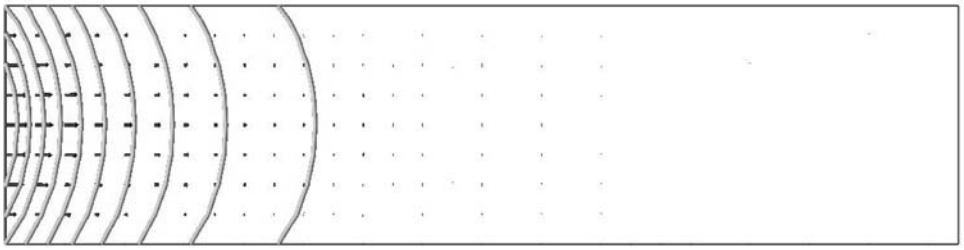

FIG. 1.4. *Cross-section of flow through the brick domain using the vorticity formulation of the Navier–Stokes equations. The normal velocity is set to* $v_x = y(1-y)z(1-z)$ *at the inlet and the pressure to* 0 *at the outlet, and the contour bars represent evenly spaced pressure lines.*

by

$$\begin{aligned} &\omega + \nabla \times \mathbf{v} = 0 && \text{in } \Omega, \\ &-Re(\mathbf{v}\cdot\nabla)\mathbf{v} - \nabla p + \nabla \times \omega = 0 && \text{in } \Omega, \\ &\nabla \cdot \mathbf{v} = 0 && \text{in } \Omega. \end{aligned} \tag{1.8}$$

Because ω, the vorticity, is the negative curl of $\mathbf{v}$ and we are solving a minimization problem, we can augment this first-order system by the following consistent equation:

$$\nabla \cdot \omega = 0 \quad \text{in } \Omega. \tag{1.9}$$

The corresponding functional for the vorticity formulation is

$$\begin{aligned} G_\omega(\omega, \mathbf{v}, p) := \|\omega + \nabla \times \mathbf{v}\|^2_{0,\Omega} + \| - Re\,(\mathbf{v}\cdot\nabla)\,\mathbf{v} - \nabla\, p + \nabla \times \omega\|^2_{0,\Omega} \\ + \|\nabla \cdot \mathbf{v}\|^2_{0,\Omega} + \|\nabla \cdot \omega\|^2_{0,\Omega}. \end{aligned} \tag{1.10}$$

One advantage of this formulation is that it contains only 7 dependent variables in three dimensions, almost half the number of the velocity-flux formulation; however, since it is not a fully H^1-coercive system [6], optimal multigrid convergence cannot be proven. Nevertheless, since the focus here is on mass conservation, we examine the effects of this formulation on mass loss through the brick domain using the same boundary conditions as Figure 1.2 (i.e., $v_x = y(1-y)z(1-z)$ along the inlet). The results are summarized by the cross-section shown in Figure 1.4, and they demonstrate extensive mass loss similar to the velocity-flux formulation. In this case, the mass loss is 99.9%, even worse than the 99.8% previously observed, and the multigrid convergence factor was significantly worse, requiring approximately twice the number of V-cycles. The final value of the functional, G_ω, was 2.7×10^{-3}, but most of the error was in the divergence-of-velocity term (1.1×10^{-3}). Clearly, the two first-order systems for the stationary Navier–Stokes equations commonly found in the literature do not provide acceptable mass conservation for this set of boundary conditions in three dimensions.

In summary, even though some of the remedies tested improved mass conservation, they did so at the cost of multigrid performance. Since the goal is to improve mass conservation without an appreciable decline in multigrid performance, none of these remedies is acceptable. The only solution shown to improve mass conservation without a decline in multigrid performance is setting boundary conditions on pressure for both the inlet and outlet. This observation first provoked an effort to find a new formulation that allows pressure boundary conditions to be set in a more flexible way.

During this effort, we developed a formulation that not only allowed new boundary conditions to be set that incorporated pressure, but the new formulation also achieves better computational performance (i.e., multigrid convergence rates) at the same. In this way, it satisfies our goal of improving mass conservation while maintaining the strengths of the LSFEM methodology.

2. New formulation. Before defining the new formulation, we need the identity

$$\nabla \times (\mathbf{v} \cdot \nabla)\mathbf{v} = (\mathbf{v} \cdot \nabla)\omega - (\omega \cdot \nabla)\mathbf{v}, \tag{2.1}$$

and we define a new variable $\mathbf{r}$ as

$$\mathbf{r} = \nabla p + Re(\mathbf{v} \cdot \nabla)\mathbf{v} \ . \tag{2.2}$$

The new variable is equal to the pressure gradient plus the convective term. We now introduce the following first-order system to replace the Navier–Stokes equations:

$$\begin{array}{ll} \nabla \times \mathbf{v} + \omega = 0 & \text{in } \Omega, \\ \nabla \cdot \mathbf{v} = 0 & \text{in } \Omega, \\ \nabla \times \omega - \mathbf{r} = 0 & \text{in } \Omega, \\ \nabla \cdot \omega = 0 & \text{in } \Omega, \\ \nabla \times \mathbf{r} - Re(\mathbf{v} \cdot \nabla\omega - \omega \cdot \nabla\mathbf{v}) = 0 & \text{in } \Omega, \\ \nabla \cdot \mathbf{r} = 0 & \text{in } \Omega. \end{array} \tag{2.3}$$

The first two equations in system (2.3) are the same as those previously used in the vorticity formulation. The third equation is the momentum balance, rewritten in terms of $\mathbf{r}$ and the curl of vorticity. The fourth equation is derived by taking the divergence of the first equation. The fifth equation comes from taking the curl of (2.2), and the final equation is the result of taking the divergence of the momentum balance. The corresponding functional for the new formulation is

$$\begin{aligned} G_{new}(\omega, \mathbf{v}, \mathbf{r}) := & \|\omega + \nabla \times \mathbf{v}\|^2_{0,\Omega} + \|\nabla \cdot \mathbf{v}\|^2_{0,\Omega} \\ & + \|\mathbf{r} + \nabla \times \omega\|^2_{0,\Omega} + \|\nabla \cdot \omega\|^2_{0,\Omega} \\ & + \|\nabla \times \mathbf{r} - Re(\mathbf{v} \cdot \nabla\omega - \omega \cdot \nabla\mathbf{v})\|^2_{0,\Omega} + \|\nabla \cdot \mathbf{r}\|^2_{0,\Omega} \ . \end{aligned} \tag{2.4}$$

To further analyze this system of equations, it is useful to rewrite the system using slack variables, α, β, and δ, and moving subprincipal terms to the right-hand side, giving:

$$\begin{array}{cccccccc} \nabla \times \mathbf{v} & -\nabla\alpha & & & & & = & -\omega, \\ \nabla \cdot \mathbf{v} & & & & & & = & 0, \\ & & \nabla \times \omega & -\nabla\beta & & & = & \mathbf{r}, \\ & & \nabla \cdot \omega & & & & = & 0, \\ & & & & \nabla \times \mathbf{r} & -\nabla\delta & = & Re(\mathbf{v} \cdot \nabla\omega - \omega \cdot \nabla\mathbf{v}), \\ & & & & \nabla \cdot \mathbf{r} & & = & 0. \end{array} \tag{2.5}$$

Here, the slack variables are included to help analyze the new formulation, but the boundary conditions are set so that the slack variables, α, β, and δ, are equal to zero everywhere and, hence, are not necessary in the computation. The system of equations given by (2.5) contain a block diagonal operator with three blocks. Each block consists of a curl and divergence of a variable we are interested in approximating. With the proper boundary conditions on each block (specifically, one condition on the

slack variable and one on the original variable in the normal direction or, alternatively, conditions on the original variable in the two tangential directions), it is easy to show that the functional associated with (2.5) and $Re = 0$ is continuous and coercive in the $(H(\mathrm{div}) \cap H(\mathrm{curl})) \times H^1$ norm for each block of variables. The results are obtained by using a standard compactness argument. For the lowest block, containing the curl and divergence of $\mathbf{r}$, boundary conditions are typically set on $\mathbf{n} \cdot \mathbf{r}$, where $\mathbf{n}$ is the normal vector, and on slack variable δ. For the central block, boundary conditions are set on the normal vorticity and on slack variable β. For the upper block, boundary conditions only need to be set on the normal velocity and α because the tangential velocity is related to the normal vorticity, but, in practice, boundary conditions are set on two or all three of the velocity components.

If the Reynolds number is set to zero, the system of equations (2.5) can be solved by inverting the individual blocks of the matrix. First, the bottom block is inverted to calculate $\mathbf{r}$, then $\mathbf{r}$ is used on the right-hand side of the central block, which is inverted to calculate ω. Finally, ω is used to calculate the velocity, $\mathbf{v}$. In this way, storage costs can be reduced by requiring only that the blocks of the operator be stored. Further, these div/curl blocks are solved very efficiently by multigrid solvers. If the Reynolds number is not equal to zero, the block strategy can still be used, but an initial guess for $\mathbf{v}$ and ω is required followed by multiple iterations. In practice, we assemble the full operator because it is efficiently handled by multigrid and makes the implementation more straightforward.

The boundary conditions for the new formulation are similar to those typically used for the traditional vorticity formulation except for those on $\mathbf{r}$. An excellent description and classification of possible boundary conditions on $\mathbf{v}$ and ω can be found in the work of Bochev [3]. Fully specifying the velocity along all boundaries is sufficient to obtain a well-posed problem having a unique solution. However, the inclusion of boundary conditions on $\mathbf{n}\cdot\omega$ and $\mathbf{n}\cdot\mathbf{r}$ results in the operator being coercive in H^1 and optimal multigrid performance on the linear system. Before the boundary conditions are described in detail, it is important to note that all boundary conditions can be imposed either strongly on the finite element space or weakly by including the boundary conditions in the least-squares functional. This flexibility is useful since we want to strongly impose some boundary conditions, like the no-slip condition on velocity, while others are typically imposed weakly.

Along the walls or the no-slip boundaries, all velocity components are set to zero, the normal vorticity is set to zero, and $\mathbf{n} \cdot \mathbf{r}$ is set to zero (see Table 2.1). The last boundary condition, which is set on $\mathbf{r}$, can be explained by the equation

$$\mathbf{n} \cdot \mathbf{r} = \mathbf{n} \cdot \nabla p + \mathbf{n} \cdot (Re(\mathbf{v} \cdot \nabla)\mathbf{v}) = \mathbf{n} \cdot \Delta \mathbf{v}, \tag{2.6}$$

where $\Delta \mathbf{v} \equiv (\nabla \cdot \nabla)\mathbf{v}$. Therefore, setting $\mathbf{n} \cdot \mathbf{r}$ to zero along the wall can be viewed physically in two different ways. First, it is equivalent to setting the second derivative of the normal velocity in the normal direction to zero (i.e., the normal velocity must be zero near the wall as well as at the wall). Second, since the normal velocity is zero at the wall, it is equivalent to setting the normal derivative on pressure to zero, and this is consistent with the basic assumptions of Prandtl's boundary layer theory [28].

Inflow and outflow boundary conditions are a well-known difficulty in computational fluid dynamics [18]. For the new formulation, boundary conditions are typically set on two or three of the velocity components (see Table 2.1), although it is possible to set a condition on just the normal velocity. A condition should be set only on the normal vorticity because fully specifying the vorticity usually results in a problem

TABLE 2.1

Boundary conditions used for the the new formulation. τ_1 and τ_2 are the two tangential vectors along the inlet and c is an unknown constant that approximates $\frac{\partial^2 \mathbf{v}_n}{\partial \mathbf{n}^2}$.

Boundary	Condition	Enforcement
Wall	$\mathbf{v} = 0$	strong
	$\mathbf{n} \cdot \omega = 0$	weak
	$\mathbf{n} \cdot \mathbf{r} = 0$	weak
Inflow	$\mathbf{n} \cdot \mathbf{v} = g$	strong
	$\mathbf{n} \times \mathbf{v} = 0$	strong
	$\mathbf{n} \cdot \omega = 0$	weak
	$\mathbf{n} \cdot \mathbf{r} = \frac{\partial^2 \mathbf{v}_n}{\partial \tau_1^2} + \frac{\partial^2 \mathbf{v}_n}{\partial \tau_2^2} + c$	weak
Outflow	$\mathbf{n} \times \omega = 0$	strong
	$\mathbf{n} \cdot \omega = 0$	weak
	$\mathbf{n} \times \mathbf{r} = 0$	weak

that is not well posed [3]. The final boundary condition to be set is on $\mathbf{r}$. In the test problem, for example, a boundary condition setting the pressure equal to zero was used. An equivalent boundary condition here would be to set the value of $\mathbf{r}$ in the tangential direction(s) to zero, assuming that the tangential velocity is zero. If a steady flow boundary condition is desired, then $\mathbf{n} \cdot \mathbf{r}$ is set to $\mathbf{n} \cdot \Delta\mathbf{v}$, which is fully known since the second derivative in the normal direction is zero under steady flow conditions. The most difficult boundary condition to handle is when the velocity is fully specified, but the velocity profile is not equal to the steady profile, i.e., $\mathbf{n} \cdot \Delta\mathbf{v}$ is not fully known. Clearly, we know the second derivatives of the normal velocity in the tangential directions, but the second derivative in the normal direction of the normal velocity is not known and it is generally nonzero. One possible mechanism for solving for this unknown is to (1) assume the unknown is zero, (2) impose the boundary condition weakly, (3) solve the minimization problem, (4) check the value of the unknown quantity, and (5) repeat using the new value until convergence. This method has potentially slow convergence, and we have developed an alternative approach. Recalling that one of our primary goals is global mass conservation, we propose setting the second derivative of $\mathbf{n} \cdot \mathbf{v}$ in the normal direction equal to an unknown constant and solving for the constant using the constraint of global mass conservation.

Solving for the unknown constant in the $\mathbf{n} \cdot \mathbf{r}$ boundary condition can be accomplished easily if the Reynolds number is zero. In this case, two different guesses of the value of the constant are made, and the problem is solved with each guess. Unless one of the guesses is exact, the inflow rate will not exactly equal the outflow rate. However, based on the different mass conservation results, linear interpolation can be used to determine the correct value of the unknown constant, and a linear combination of the two solutions gives the final solution. For the case of the nonlinear Navier–Stokes equations, the same basic procedure of guessing different values of the constant may be used. In this case, however, more than two solutions may be required to get acceptably close to global conservation. This process often does not add any additional computational cost, in the case of a nonzero Reynolds number, because multiple iterations are already required by the nonlinear operator.

To solve the least-squares problem, the equations that are used in the functional (G_{new}) are first linearized so that the solution can be found using a Gauss–Newton approach. The value of the nonlinear functional is calculated after each Gauss–Newton step to ensure that the nonlinear functional is decreasing to a minimum. A line search in the direction of the Gauss–Newton step could be used to ensure that the nonlinear

functional decreases at each step. However, for all the problems that we tested, no line search was necessary. The functional for the linearized equations is minimized using standard techniques from the calculus of variations to obtain the weak form (e.g., [5, 3]). A finite element basis is then chosen so that the weak form generates a matrix problem. An alternative approach is to linearize the weak form instead of linearizing the original equations [14, 15], but we found better convergence by first linearizing the equations. All of the simulations presented in the results section utilized a trilinear finite element basis for all of the variables unless otherwise noted. The FOSLS formulation allows the solution spaces for the variables to be chosen independently, and there is no restrictive stability condition (i.e., inf-sup condition) to satisfy. As a result, both the pressure and velocity in the Navier–Stokes equations can be approximated with the same basis.

All simulations, including those in the introduction section, were performed using the ParaFOS code, written by the authors. The code imports hexahedral meshes from the Cubit mesh generation package (Sandia National Laboratory). The finite element meshes are then partitioned using the Metis graph partitioning library [23]. The software is designed to run on distributed memory clusters using the MPI library for communication. The linear matrix problem generated during each Gauss–Newton step is solved using the *hypre* library of solvers ([17], Center for Applied Scientific Computing, Lawrence Livermore National Laboratory). Specifically, the Boomer-AMG parallel algebraic multigrid (AMG) solver is used as a preconditioner for a conjugate gradient solver. Most of the solver parameters used were the default settings except for the coarse grid selection algorithm, which was set to the parallel modified independent set algorithm. This algorithm was chosen because it chooses coarse grids with fewer nodes, thus reducing storage requirements.

3. Results. The brick test problem with all components of the inlet velocity specified ($v_x = y(1-y)z(1-z)$, $v_y = 0$, $v_z = 0$) is complicated because the value of $\mathbf{n} \cdot \Delta\mathbf{v}$ is not completely known over the inlet. Specifically, the value of $\frac{\partial^2 v_x}{\partial x^2}$, which we refer to as C_1, is unknown. While this value may be a function of (y, z) along the inlet, we assume it is a constant and solve for it using the constraint of global mass conservation. Since the Reynolds number is also set to zero, this constant is equal to the deviation of the normal pressure gradient from the steady flow value. In other words, if the inlet velocity profile was the steady velocity profile, the pressure gradient would be a constant (i.e., a linear pressure drop), and the value of C_1 would be zero. The appropriateness of our assumption that C_1 is a constant and not a function of (x, y) is going to depend upon how close the specified inlet velocity profile is to the steady velocity profile. If the specified profile is close in some regions and deviates greatly in other regions, the assumption will not be appropriate. If the specified profile is close to the steady profile, the assumption is not a considerable source of error. A more detailed analysis of the error resulting from this assumption is the subject of a future paper. For now, we just show numerically that setting C_1 to a constant is acceptable for the chosen test problems.

We begin by setting $C_1 = 0$ and minimizing the functional, G_{new}, over the brick-shaped domain. Clearly, $C_1 \neq 0$ and the result is that the inflow rate is 2.7×10^{-2} and the outflow rate is 1.9×10^{-2}. It is well known from computational and experimental fluid dynamics that the pressure drop in the normal direction at the inlet must be larger than the steady-state pressure drop due to entrance effects. In other words, $C_1 > 0$ due to the extra pressure drop to rearrange the flow profile into the steady-state profile. For the second run, we set $C_1 = 1$ and minimize the functional again.

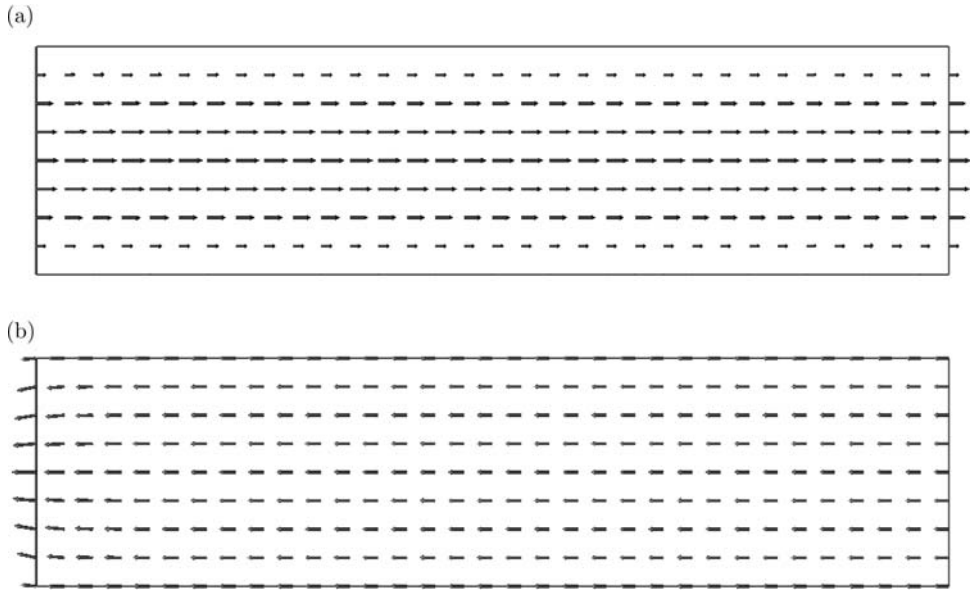

FIG. 3.1. *Central cross-section of flow through the brick domain using the new formulation of the Navier–Stokes equations. The normal velocity is set to* $v_x = y(1-y)z(1-z)$ *at the inlet and the pressure to* 0 *at the outlet. Velocity vectors are shown in* (a) *and the pressure gradient is shown in* (b).

The inflow rate remains unchanged at 2.7×10^{-2} (this essential boundary condition was strongly enforced), but now the outflow rate is 4.8×10^{-2}. Instead of losing mass, we are now "creating" mass. The true value for C_1 can be determined by linear interpolation to be 0.28 to give discrete mass conservation. To get the final solution shown in Figure 3.1, we can either use the appropriate linear combination of previous solutions or simply resolve the problem using $C_1 = 0.28$. Both the global and local conservation of mass using the final value of C_1 are excellent as mass is conserved exactly to the 5 digits calculated for any cross-section. Further, the average AMG convergence factor is also excellent at 0.3, indicating that only 15 V-cycles are required to reduce the residual by a factor of 1×10^{-8}. A comparison of the 3 different functionals, $G_{v.f.}$, G_ω, and G_{new}, is given in Table 3.1. The new formulation provides excellent global mass conservation and significantly better computational performance while maintaining computational scalability. It is not appropriate to compare the minimums of the different functionals, and they are only given for reference. Under steady flow conditions, the pressure drop in the normal direction is 1.0 and, now, for this inlet flow profile, the pressure drop at the center of the entrance is approximately 1.28. Figure 3.1(b) shows the pressure gradient throughout the domain. Note that it is constant at the outlet, but there are some entrance effects near the inlet.

Similar results can be achieved for problems with nonzero Reynolds number. Repeating the previous problem with $Re = 40$ results in $C_1 = 0.46$ being required for global mass conservation. In this case, the average AMG convergence factor deteriorates slightly to between 0.36 and 0.5, depending on the Gauss–Newton step, but it remains much improved over the factors of the other two formulations, which are between 0.9 and 0.95. As the Reynolds number becomes larger and larger, the entrance effects extend further downstream. For laminar flow into a round pipe, the

TABLE 3.1
Comparison of the different functionals for solving Stokes flow through the brick-shaped domain with the velocity fully specified along the inlet.

Method	Mesh	Mass loss	Functional	Avg. C.F.	CPU time
Velocity-flux	$32 \times 8 \times 8$	99.8%	6×10^{-3}	0.74	60 sec.
Vorticity	$32 \times 8 \times 8$	99.9%	2.7×10^{-3}	0.88	16 sec.
New	$16 \times 4 \times 4$	$< 0.01\%$	2.3×10^{-1}	0.25	1.6 sec.
New	$32 \times 8 \times 8$	$< 0.01\%$	9.6×10^{-2}	0.3	15 sec.
New	$64 \times 16 \times 16$	$< 0.01\%$	3.0×10^{-2}	0.36	150 sec.

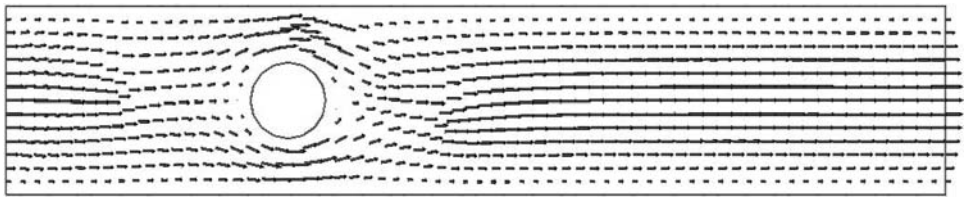

FIG. 3.2. *Central cross-section of flow through the brick domain* $(5 \times 1 \times 1)$ *with a cylindrical obstruction with a diameter of* 0.4. *The normal velocity is set to* $v_x = y(1-y)z(1-z)$ *at the inlet, and the nodal velocity vectors are shown.*

accepted correlation [27] is $L \approx 0.06 \cdot Re \cdot d$, where d is the diameter and L is the length of the entrance effects. The current assumption of steady flow at the outlet is probably appropriate since the inlet profile is close to the steady profile, but the assumption will break down at higher Reynolds numbers.

An obstruction can be included in this first test problem by removing a cylindrical portion of the domain. The results for $Re = 100$ and an obstruction with a diameter of 0.4 are shown in Figure 3.2. The boundary conditions along the walls of the obstruction are the same as those along the outer walls of the domain (i.e., $\mathbf{v} = 0$, $\mathbf{n} \cdot \omega = 0$, and $\mathbf{n} \cdot \mathbf{r} = 0$), and, in this case, a triquadratic finite element basis is used to more accurately capture the cylindrical geometry. Interestingly, to achieve global mass conservation C_1 is set to 0.20, a smaller value than the unobstructed case, but this is likely due to the more accurate triquadratic basis. The finite element mesh for this problem contains 1290 hexahedral elements and is unstructured, which contributes to some of the deterioration in the average AMG convergence factor to 0.74. For this problem, the domain was extended to a dimensionless length of 5 to ensure that the outlet profile is the steady profile.

The geometry and boundary conditions for the second test problem are somewhat similar to the first. In this case, however, the domain is longer, having an aspect ratio of 8:1:1, and there is a 0.5:0.5:1 notch removed from the lower half of the inlet. In other words, this is the classic flow over a backward-facing step in three dimensions. The inlet velocity is once again set to a tensor product of one-dimensional parabolas, $v_x = y(0.5-y)(0.5+z)(0.5-z), v_y = 0, v_z = 0$, and the tangential velocity is set to zero along the outlet. Along the inlet, $\mathbf{n} \cdot \mathbf{r} = -2(0.5+z)(0.5-z) - 2y(0.5-y) - C_2$, where C_2 is a constant that is set so that global conservation of mass is achieved, and $\tau \cdot \mathbf{r}$ is set to zero along the outlet. Finally, the no-slip boundary conditions along the wall are set the same as before, with the notable exception of the condition on $\mathbf{n} \cdot \mathbf{r} = \mathbf{n} \cdot \Delta\mathbf{v}$, which is *not* set to zero near the reentrant corner of the step. This is because the physical assumption that allowed us to set this to zero along a smooth wall

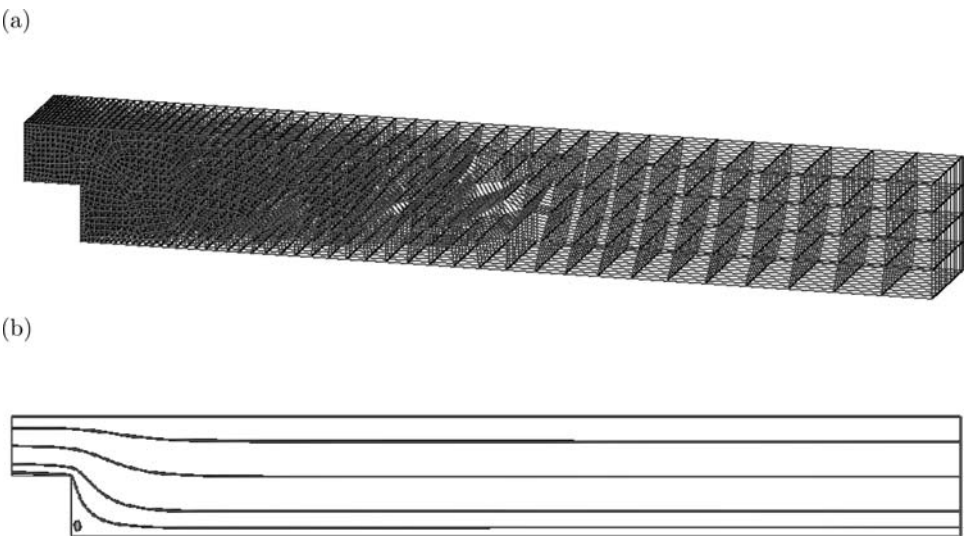

FIG. 3.3. *Central cross-section of flow through the backward-facing step domain using the new formulation of the Navier–Stokes equations at* $Re = 50$. *The finite element mesh are shown in* (a) *and streamlines are shown in* (b).

no longer holds. In this problem, there is a boundary layer separation at the corner, and there is an infinite number of normal vectors at the corner. Numerical experiments were performed on different methods for removing this boundary condition from the corner. First, the $\mathbf{n}\cdot\mathbf{r}$ condition, which is typically enforced weakly, can be completely removed from the functional along the two walls that contact at the corner. Second, the $\mathbf{n}\cdot\mathbf{r}$ term can be multiplied by a weight function that is zero at the corner and one at some radius away from the corner [19]. A typical weight function has the form $w(r) = (r/R)^n$, where r is the distance from the corner, R is the radius of the weight function (typically $O(0.1)$), and n is the polynomial order of the weight function (typically 3 or 4). Acceptable (and indistinguishable) results were obtained with either method as long as the radius of the weight function was sufficiently large.

The finite element mesh used for the backward-facing step geometry is shown in Figure 3.3(a). The mesh is unstructured and consists of 6510 elements. A triquadratic finite element basis was used to improve the accuracy of the method. An analysis of the accuracy of LSFEM for the two-dimensional backward-facing step problem can be found in Röhrle [25] and for the three-dimensional problem in Jiang et al. [22]. A Reynolds number of 50 was used to obtain the streamlines in a central cross-section of the approximate solution shown in Figure 3.3(b). At this low Reynolds number, only a small recirculation is found downstream of the step; this is consistent with the two-dimensional experimental results of Armaly et al. [1]. The global conservation of mass was exact, since it was enforced by setting C_2, and the behavior of the pressure field was also consistent with experimental predictions in that the pressure gradient in the flow direction was greater for the small channel, upstream from the step, than for the larger channel downstream from the step. The AMG convergence factor was 0.49 for a trilinear basis and 0.72 for a triquadratic basis. Both factors are acceptable considering the higher-order finite element basis, the partitioning of the problem among 16 processors, and absence of a boundary condition on $\mathbf{r}$ along the step.

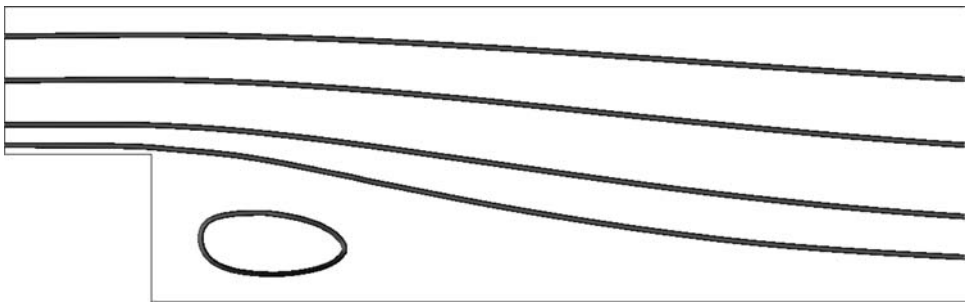

FIG. 3.4. *Central cross-section of flow through the backward-facing step domain using the functional $G_{new,b}$ at $Re = 200$. Only the upstream half of the total domain is shown for clarity.*

For greater values of Re, it is potentially helpful to rescale the momentum equation by dividing by Re, giving a new functional

$$
\begin{aligned}
G_{new,b}(\omega, \mathbf{v}, \mathbf{r}) := {} & \|\omega + \nabla \times \mathbf{v}\|_{0,\Omega}^2 + \|\nabla \cdot \mathbf{v}\|_{0,\Omega}^2 \\
& + \|\mathbf{r} + \frac{1}{Re}\nabla \times \omega\|_{0,\Omega}^2 + \|\nabla \cdot \omega\|_{0,\Omega}^2 \\
& + \|\nabla \times \mathbf{r} - (\mathbf{v} \cdot \nabla\omega - \omega \cdot \nabla\mathbf{v})\|_{0,\Omega}^2 + \|\nabla \cdot \mathbf{r}\|_{0,\Omega}^2 \ ,
\end{aligned} \tag{3.1}
$$

in which the diffusive term becomes small instead of the convective term becoming large and dominating the other terms in the functional. Figure 3.4 shows streamlines in the upstream half of the domain for flow over the backward-facing step at a Reynolds number of 200. Interestingly, when triquartic elements were used, excellent conservation ($> 99\%$ of mass was conserved) is achieved even though the boundary conditions on $\mathbf{n} \cdot \mathbf{r}$ were not set along the inflow. Consequently, in this case, we did not have to solve for the value of an unknown constant. Unfortunately, the lack of a boundary condition on $\mathbf{r}$ means that the operator is no longer elliptic, and the convergence rate of the linear solver is slower. The excellent conservation without a boundary condition on $\mathbf{r}$ along the inlet and outlet is not observed if a trilinear basis is used or if the Reynolds number is small (i.e., less than approximately 100 in this case). Regardless, this is an encouraging result because the requirement of an inflow boundary condition on $\mathbf{n} \cdot \mathbf{r}$ is one of the few drawbacks of the new formulation.

The final test problem is flow with $Re = 40$ through a series of bifurcations, as shown in Figure 3.5. The inlet is the single tube at the top, and the flow into the initial tube is the steady flow profile, which has a known analytic form. Boundary conditions along the inlet are very straightforward as a result of the steady profile. Velocity is set to $v_x = 0$, $v_y = 0$, and $v_z = -1 + (x^2 + y^2)$, normal vorticity is set to zero, and $\mathbf{n} \cdot \mathbf{r} = \frac{\partial^2 v_z}{\partial x^2} + \frac{\partial^2 v_z}{\partial y^2} = 4$. Because we wish to accurately capture the inflow boundary condition on velocity, a *triquartic* finite element basis is used for this problem. The no-slip conditions, specified previously, are set along the walls. The main inlet tube branches into two tubes, and each of these tubes branch, giving a total of four outlet surfaces. The outflow boundary condition is, as always in computational fluid dynamics, difficult to determine. We begin by setting the tangential velocity to zero, which implies that the normal vorticity is also zero. Another potential boundary condition is setting the pressure to a constant by setting $\tau \cdot \mathbf{r} = 0$. However, better accuracy is usually found by setting a boundary condition on $\mathbf{n} \cdot \mathbf{r}$. If the flow is steady (i.e., fully developed), then $\mathbf{n} \cdot \mathbf{r} = \textit{constant}$, and the value of the *constant* can

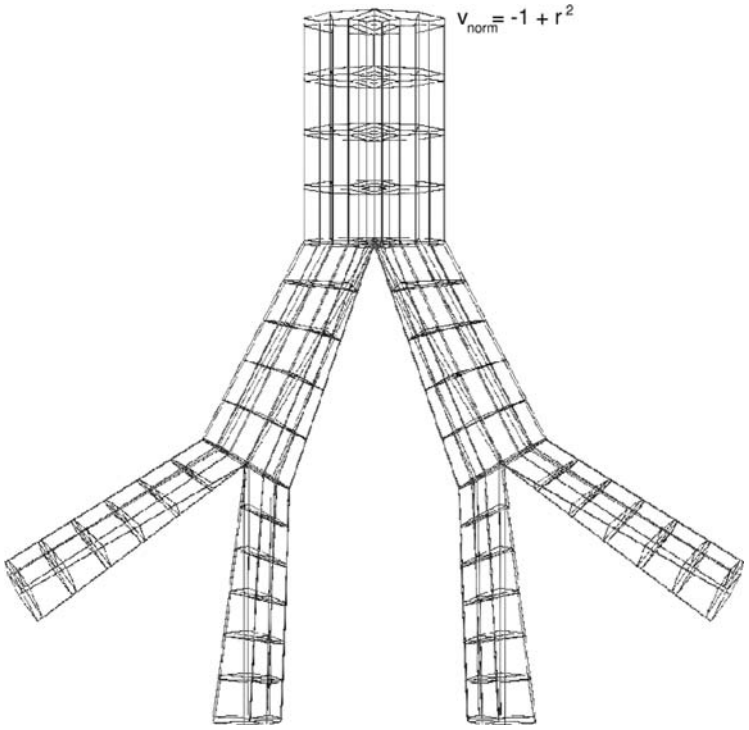

FIG. 3.5. *The hexahedral mesh used for the bifurcating flow domain. The inlet is the single tube at the top, and the four tubes at the bottom contain the outflow surfaces.*

be determined based on the geometry and setting the inflow rate equal to the outflow rate. However, because the tubes are not exact cylinders due to the bifurcations, it is unlikely that the outflow is fully developed. A third possibility is to assume that the flow has not developed fully (i.e., $\frac{\partial^2 v_n}{\partial n^2} \neq 0$), but well enough that it is a paraboloid at the outlet, albeit an unsteady one due to small changes in tube geometry. In this case, $\mathbf{n} \cdot \mathbf{r}$ is equal to an unknown constant, C_3. We solve for this unknown constant using the constraint that mass be conserved globally.

Figure 3.6 shows streamlines for the central cross-section of the bifurcating tubes domain. Also shown in the inset images are velocity vectors for specific branches. Because the triquartic finite element basis was based on Chebyshev–Lobatto node positions, which have been previously shown to work well with AMG solvers [20], the nodes are not evenly distributed but are somewhat clustered near the corners of elements. The global conservation of mass is approximately exact because C_3 was set to enforce this condition, but it is useful to examine the pressure gradient to determine the accuracy of the solution. For laminar flow in a straight, cylindrical tube, the relationship between the pressure gradient, flow rate (Q), and tube radius (R) is given by [27]

$$\frac{\partial p}{\partial n} \propto \frac{Q}{R^4}. \tag{3.2}$$

If the flow rate into the domain is Q, the flow rate through each of the outlets is approximately $Q/4$ due to symmetry. Further, for this geometry, the radius of the inlet is R, and the radius of each outlet tube is $R/2$. Combining these relationships

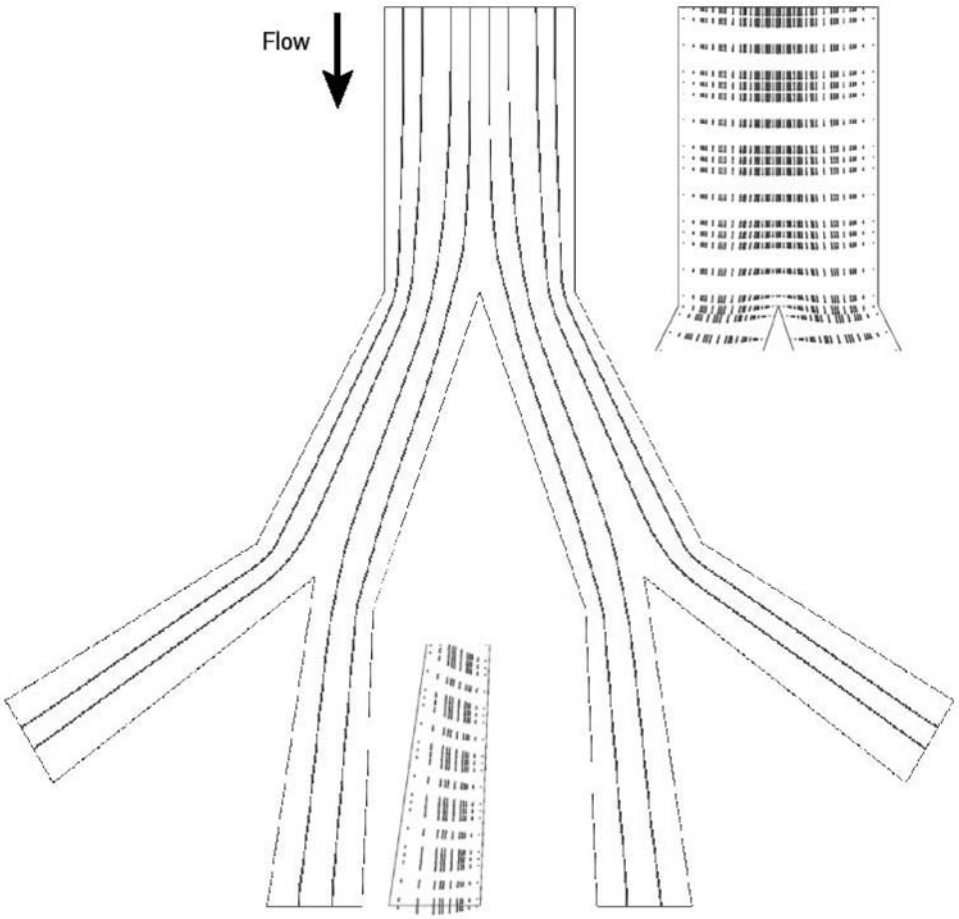

FIG. 3.6. *The central cross-section of the bifurcating tube domain showing streamlines through the bifurcations. The inset images show velocity vectors for the nearby tubes. The inlet is the single tube at the top, and the four tubes at the bottom contain the outflow surfaces.*

and using (3.2) results in the prediction that the normal pressure gradient at the outlet is roughly 4 times the normal pressure gradient at the inlet. Since the normal pressure gradient at the inlet is equal to 4, we expect that the normal pressure gradient at the outlet is roughly 16, but probably slightly higher due to the multiple bifurcations and entrance effects. For the result shown in Figure 3.6, the normal pressure gradient at the outlet was 17.4, indicating good agreement with our straight tube heuristic. Further, the approximation just presented can be used to obtain a good initial guess for the value of C_3.

Conclusions. Poor mass conservation can be observed when reformulating the Navier–Stokes equations as a first-order system using either the velocity-flux or vorticity formulations. The problem occurs primarily when the pressure (or normal stress) is not specified on both ends of the domain. To help address this problem, we are proposing a new first-order system of equations that allows boundary conditions to be set on the pressure gradient. In many cases, it is easier to specify boundary conditions on the pressure gradient because much of the information can be obtained from the

boundary conditions on velocity. Further, when using the new formulation in conjunction with AMG, significant computational improvements can also be observed. In this way, the new formulation achieves the objective of improved mass conservation on coarse grids while maintaining AMG convergence rates. The new formulation achieves only local mass conservation asymptotically as the mesh size goes to zero, like other least-squares formulations, but overall conservation is much improved for coarser meshes due to the boundary condition on $\mathbf{n}\cdot\mathbf{r}$. For each of the three test problems presented here—the square channel, the backward-facing step, and the series of bifurcations—excellent mass conservation was possible for both Stokes flow and low to moderate Reynolds numbers. Traditionally, high Reynolds number flows have been a strength of least-squares methods because of their excellent stability properties. However, high Reynolds numbers also imply that flow through the outflow boundary is unlikely to be steady, a difficult condition to handle with the new formulation. However, we are confident that as experience is gained with the new formulation, the proper boundary conditions can be found for a much wider set of problems.

REFERENCES

[1] B. F. Armaly, F. Durst, J. C. F. Pereira, and B. Schönung, *Experimental and theoretical investigation of backward-facing step flow*, J. Fluid Mech., 127 (1983), pp. 473–496.

[2] T. Austin, T. Manteuffel, and S. McCormick, *A robust approach to minimizing h(div) dominated functionals in an h^1-conforming finite element space*, J. Numer. Linear Algebra Appl., 11 (2004), pp. 115–140.

[3] P. B. Bochev, *Analysis of least-squares finite element methods for the Navier–Stokes equations*, SIAM J. Numer. Anal., 34 (1997), pp. 1817–1844.

[4] P. B. Bochev, Z. Cai, T. M. Manteuffel, and S. F. McCormick, *Analysis of velocity-flux first-order system least-squares principles for the Navier–Stokes equations: Part* I, SIAM J. Numer. Anal., 35 (1998), pp. 990–1009.

[5] P. B. Bochev and M. D. Gunzburger, *Accuracy of least-squares methods for the Navier-Stokes equations*, Comput. Fluids, 22 (1993), pp. 549–563.

[6] P. B. Bochev and M. D. Gunzburger, *Finite element methods of least-squares type*, SIAM Rev., 40 (1998), pp. 789–837.

[7] P. B. Bochev, T. A. Manteuffel, and S. F. McCormick, *Analysis of velocity-flux least-squares principles for the Navier–Stokes equations: Part* II, SIAM J. Numer. Anal., 36 (1999), pp. 1125–1144.

[8] P. Bolton and R. W. Thatcher, *On mass conservation in least-squares methods*, J. Comput. Phys., 201 (2004), pp. 110–118.

[9] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.

[10] Z. Cai, B. Lee, and P. Wang, *Least-squares methods for incompressible Newtonian fluid flow: Linear stationary problems*, SIAM J. Numer. Anal., 42 (2004), pp. 843–859.

[11] Z. Cai, T. A. Manteuffel, S. F. McCormick, and J. Ruge, *First-order system $\mathcal{LL}^*$ (FOSLL*): Scalar elliptic partial differential equations*, SIAM J. Numer. Anal., 39 (2001), pp. 1418–1445.

[12] Z. Cai, T. A. Manteuffel, and S. F. McCormick, *First-order system least squares for second-order partial differential equations: Part* II, SIAM J. Numer. Anal., 34 (1997), pp. 425–545.

[13] C. L. Chang and J. J. Nelson, *Least-squares finite element method for the Stokes problem with zero residual of mass conservation*, SIAM J. Numer. Anal., 34 (1997), pp. 480–489.

[14] A. L. Codd, T. A. Manteuffel, and S. F. McCormick, *Multilevel first-order system least squares for nonlinear partial differential equations*, SIAM J. Numer. Anal., 41 (2003), pp. 2197–2209.

[15] A. L. Codd, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge, *Multilevel first-order system least squares for elliptic grid generation*, SIAM J. Numer. Anal., 41 (2003), pp. 2210–2232.

[16] J. M. Deang and M. D. Gunzburger, *Issues related to least-squares finite element methods for the Stokes equations*, SIAM J. Sci. Comput., 20 (1998), pp. 878–906.

[17] R. D. Falgout and U. M. Yang, *hypre: A library of high performance preconditioners*, in Computational Science—ICCS 2002, Part III, Lecture Notes in Comput. Sci. 2331, Springer-Verlag, Berlin, 2002, pp. 632–641.
[18] P. M. Gresho and R. L. Sani, *Introducing four benchmark solutions*, Internat. J. Numer. Methods Fluids, 11 (1990), pp. 951–952.
[19] J. J. Heys, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge, *First-order systems least squares (FOSLS) for coupled fluid-elastic problems*, J. Comput. Phys, 195 (2004), pp. 560–575.
[20] J. J. Heys, T. A. Manteuffel, S. F. McCormick, and L. N. Olson, *Algebraic multigrid for higher-order finite elements*, J. Comput. Phys, 204 (2005), pp. 520–532.
[21] B. N. Jiang, *The Least-Squares Finite Element Method*, Springer-Verlag, Berlin, 1998.
[22] B. N. Jiang, L. J. Hou, T. L. Lin, and L. A. Povinelli, *Least-squares finite element solutions for three-dimensional backward-facing step flow*, Internat. J. Comput. Fluid Dynamics, 4 (1995), pp. 1–19.
[23] G. Karypis and V. Kumar, *A fast and high quality multilevel scheme for partitioning irregular graphs*, SIAM J. Sci. Comput., 20 (1998), pp. 359–392.
[24] J. P. Pontaza and J. N. Reddy, *Space-time coupled spectral/hp least-squares finite element formulations for the incompressible Navier-Stokes equations*, J. Comput. Phys., 197 (2004), pp. 418–459.
[25] O. Röhrle, *Multilevel First-Order System Least Squares for Quasilinear Elliptic Partial Differential Equations*, Ph.D. thesis, Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO, 2004.
[26] L. Q. Tang and T. H. Tsang, *A least-squares finite element method for time-dependent incompressible flows with thermal convections*, Internat. J. Numer. Methods Fluids, 17 (1993), pp. 271–289.
[27] F. M. White, *Viscous Fluid Flow*, 2nd ed., McGraw-Hill, New York, 1991.
[28] F. M. White, *Fluid Mechanics*, 5th ed., McGraw-Hill, Boston, MA, 2003.