Start-up flow in a three-dimensional lid-driven cavity by means of a massively parallel direction splitting algorithm

J. L. Guermond[1,*,†] and P. D. Minev[2]

[1]*Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA*
[2]*Department of Mathematical and Statistical Sciences, University of Alberta, Edmonton, Alberta, T6G 2G1, Canada*

SUMMARY

The purpose of this paper is to validate a new highly parallelizable direction splitting algorithm. The parallelization capabilities of this algorithm are illustrated by providing a highly accurate solution for the start-up flow in a three-dimensional impulsively started lid-driven cavity of aspect ratio $1 \times 1 \times 2$ at Reynolds numbers 1000 and 5000. The computations are done in parallel (up to 1024 processors) on adapted grids of up to 2 billion nodes in three space dimensions. Velocity profiles are given at dimensionless times $t = 4, 8$, and 12; at least four digits are expected to be correct at $Re = 1000$. Copyright © 2011 John Wiley & Sons, Ltd.

1. INTRODUCTION

The purpose of this paper is to validate a new highly parallelizable direction splitting algorithm for solving the time-dependent incompressible Navier–Stokes equations. The central idea behind this new algorithm is to replace the standard Poisson problem for the pressure correction by a series of one-dimensional second-order boundary value problems. This algorithm has been announced in [1], but the performance of the algorithm has not yet been documented anywhere. The intent of the present paper is to illustrate the capabilities of this new method for massive parallelization by computing a highly accurate solution for the start-up flow in a three-dimensional impulsively started lid-driven cavity of aspect ratio $1 \times 1 \times 2$ at Reynolds numbers 1000 and 5000 using grids consisting of up to 2×10^9 nodes. The set of data that we provide herein could constitute a helpful benchmark for developers of time-dependent three-dimensional incompressible Navier–Stokes codes.

The paper is organized as follows. A description of the new numerical method is provided in Section 2. We pursue further the ideas introduced/announced in [1] in the sense that in addition to splitting the pressure correction, we also apply a direction splitting technique to the momentum equation, thus further reducing the overall computational cost of the method. The method is validated in Section 3 by making comparisons with two-dimensional steady benchmark results from [2, 3]. Our results coincide with the highly accurate computations of [2] up to the fifth digit. The method is also validated by making comparisons with steady three-dimensional results from [4]. In Section 4, we provide time-dependent benchmark results for the three-dimensional flow at $Re = 1000$ and $Re = 5000$ and times $t = 4, 8$, and 12. The three-dimensional computations are done on a non-uniform grid composed of $1000 \times 1000 \times 2000$ grid points. The method presented in Section 2 is new, and to the best of knowledge, the results presented in Section 4 are original.

---

*Correspondence to: Jean L. Guermond, Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA.
†E-mail: guermond@math.tamu.edu

2. NUMERICAL PROCEDURE

The purpose of this section is to formulate the problem and to describe the new direction splitting algorithm.

2.1. The continuous problem

We consider the time-dependent non-dimensional incompressible Navier–Stokes equations in the dimensionless box $\Omega = [0, L_x] \times [0, L_y] \times [0, L_z] \subset \mathbb{R}^3$:

$$\begin{cases} \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla \mathbf{u}) - \dfrac{1}{Re} \Delta \mathbf{u} + \nabla \mathrm{p} = \boldsymbol{f}, & \text{in } \Omega \times (0, T], \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \times [0, T], \\ \mathbf{u}|_{\partial\Omega} = \boldsymbol{a} & \text{in } \partial\Omega \times (0, T], \\ \mathbf{u}|_{t=0} = \mathbf{u}_0 & \text{in } \Omega. \end{cases} \quad (1)$$

The quantity $\boldsymbol{f}$ is a source term, $\boldsymbol{a}$ is the boundary data, and $\mathbf{u}_0$ is the initial condition.

2.2. Heuristics

Virtually, all currently known incremental pressure-correction schemes are more or less semi-discrete versions of the following singular perturbation of the linearized version of (1):

$$\begin{cases} \partial_t \mathbf{u}_\epsilon - \dfrac{1}{Re} \Delta \mathbf{u}_\epsilon + \nabla \mathrm{p}_\epsilon = \boldsymbol{f} & \text{in } \Omega \times (0, T], \quad \mathbf{u}_\epsilon|_{\partial\Omega \times (0,T]} = \boldsymbol{a}, \quad \mathbf{u}_\epsilon|_{t=0} = 0 \\ -\Delta t \, \Delta \phi_\epsilon + \nabla \cdot \mathbf{u}_\epsilon = 0 & \text{in } \Omega \times (0, T], \qquad \partial_n \phi_\epsilon|_{\partial\Omega \times (0,T]} = 0 \\ \Delta t \, \partial_t \mathrm{p}_\epsilon = \phi_\epsilon - \dfrac{\chi}{Re} \nabla \cdot \mathbf{u}_\epsilon & \mathrm{p}_\epsilon|_{t=0} = \mathrm{p}_0 \end{cases} \quad (2)$$

where $\mathrm{p}_0 = \mathrm{p}|_{t=0}$, Δt is the perturbation parameter (i.e., $\epsilon := \Delta t$), and $\chi \in [0, 1]$ is an adjustable parameter. The convergence properties of this singular perturbation have been analyzed in [5, §3.3] and [6, §3.1], and $\boldsymbol{u}_\epsilon$ has been shown therein to be a $\mathcal{O}(\Delta t^2)$ perturbation of $\boldsymbol{u}$ in the $\boldsymbol{L}^2$-norm and a $\mathcal{O}(\Delta t^{\frac{3}{2}})$ perturbation in the $\boldsymbol{H}^1$-norm for all $0 < \chi \le 1$.

We now introduce a generalization of (2) that allows for direction splitting by considering the following alternative $\mathcal{O}(\Delta t^2)$ perturbation of the linearized version of (1):

$$\begin{cases} \partial_t \mathbf{u}_\epsilon - \dfrac{1}{Re} \Delta \mathbf{u}_\epsilon + \nabla \mathrm{p}_\epsilon = \boldsymbol{f} & \text{in } \Omega \times (0, T], \quad \mathbf{u}_\epsilon|_{\partial\Omega \times (0,T]} = \boldsymbol{a}, \quad \mathbf{u}_\epsilon|_{t=0} = \mathbf{u}_0 \\ \Delta t \, A\phi_\epsilon + \nabla \cdot \mathbf{u}_\epsilon = 0 & \text{in } \Omega \times (0, T], \qquad \phi_\epsilon \in D(A), \\ \Delta t \, \partial_t \mathrm{p}_\epsilon = \phi_\epsilon - \dfrac{\chi}{Re} \nabla \cdot \mathbf{u}_\epsilon & \mathrm{p}_\epsilon|_{t=0} = \mathrm{p}_0, \end{cases} \quad (3)$$

where the operator A and its domain $D(A)$ are such that the bilinear form $a(p, q) := \int_\Omega q A p \, \mathrm{d}\boldsymbol{x}$ satisfies the following properties:

$$a \text{ is symmetric,} \quad \text{and} \quad \|\nabla q\|_{L^2}^2 \le a(q, q), \quad \forall q \in D(A). \quad (4)$$

It can be shown that this new singular perturbation has similar approximation properties as (2) provided that the limit solution is smooth enough. The convergence analysis involves regularity of the pressure in the norm $\|\mathrm{p}\|_{D(A)} := a(\mathrm{p}, \mathrm{p})^{\frac{1}{2}}$. There are many admissible choices for the operator A. For instance, one recovers the usual pressure-correction algorithm in incremental form by using $A = -\Delta_N$, where $-\Delta_N$ is the Laplace operator supplemented with homogeneous Neumann boundary conditions. One could also use $A = I - \Delta_N$, where I is the identity operator. The key to the method presented in this paper is that the operator $A := (1 - \partial_{xx})(1 - \partial_{yy})(1 - \partial_{zz})$, with appropriate boundary conditions, satisfies the requirement (4) in three space dimensions. In two space dimensions, the operator $A := (1 - \partial_{xx})(1 - \partial_{yy})$ with appropriate boundary conditions also satisfies (4).

2.3. The direction splitting algorithm

Let us now describe the new algorithm (announced in [1]). The main idea consists of proceeding as in traditional pressure-correction algorithms by uncoupling the velocity and the pressure. This is done by solving (3) using a splitting technique that uncouples the momentum equation and the mass conservation equation. The momentum equation is solved by means of the second-order accurate direction splitting of Douglas [7], and the pressure correction is computed by using the operator $A := (1 - \partial_{xx})(1 - \partial_{yy})(1 - \partial_{zz})$. The complete algorithm is as follows:

- *Pressure predictor:* Denoting by p_0 the pressure field at $t = 0$, the algorithm is initialized by setting $p^{-\frac{1}{2}} = p_0$ and $\phi^{-\frac{1}{2}} = 0$. Then, for all $n \geq 0$, a pressure predictor is computed as follows:

$$p^{*,n+\frac{1}{2}} = p^{n-\frac{1}{2}} + \phi^{n-\frac{1}{2}}. \tag{5}$$

- *Velocity update:* The velocity field is initialized by setting $u^0 = u_0$, and for all $n \geq 0$, the velocity update is computed by solving the following series of one-dimensional problems:

$$\frac{\xi^{n+1} - u^n}{\Delta t} - \frac{1}{Re}\Delta u^n + \nabla p^{*,n+\frac{1}{2}} + \mathbf{NL}^{n+1}(u^n, u^{n-1}) = 0, \qquad \xi^{n+1}|_{\partial\Omega} = a.$$

$$\frac{\eta^{n+1} - \xi^{n+1}}{\Delta t} - \frac{1}{2Re}\partial_{xx}(\eta^{n+1} - u^n) = 0, \quad \eta^{n+1}|_{x=0,L_x} = a,$$

$$\frac{\zeta^{n+1} - \eta^{n+1}}{\Delta t} - \frac{1}{2Re}\partial_{yy}(\zeta^{n+1} - u^n) = 0, \quad \zeta^{n+1}|_{y=0,L_y} = a, \tag{6}$$

$$\frac{u^{n+1} - \zeta^{n+1}}{\Delta t} - \frac{1}{2Re}\partial_{zz}(u^{n+1} - u^n) = 0, \quad u^{n+1}|_{z=0,L_z} = a.$$

where $\mathbf{NL}^{n+1}(u^n, u^{n-1}) = \frac{3}{2}(u^n \cdot \nabla)u^n - \frac{1}{2}(u^{n-1} \cdot \nabla)u^{n-1}$.

- *Penalty step:* The pressure increment $\phi^{n+\frac{1}{2}}$ is computed for all $n \geq 0$ via the following sequence of one-dimensional problems:

$$\begin{aligned}
\psi - \partial_{xx}\psi &= -\frac{1}{\Delta t}\nabla \cdot u^{n+1}, & \partial_x\psi|_{x=0,L_x} &= 0; \\
\varphi - \partial_{yy}\varphi &= \psi, & \partial_y\varphi|_{y=0,L_y} &= 0. \\
\phi^{n+\frac{1}{2}} - \partial_{zz}\phi^{n+\frac{1}{2}} &= \varphi, & \partial_z\phi^{n+\frac{1}{2}}|_{z=0,L_z} &= 0.
\end{aligned} \tag{7}$$

- *Pressure update:* The last substep of the algorithm consists of updating the pressure as follows:

$$p^{n+\frac{1}{2}} = p^{n-\frac{1}{2}} + \phi^{n+\frac{1}{2}} - \frac{\chi}{Re}\nabla \cdot (\frac{1}{2}(u^{n+1} + u^n)), \tag{8}$$

where the parameter χ is set to 1 in all the simulations reported thereafter.

The two-dimensional version of the algorithm is obtained by skipping the last substeps in (6) and (7) and setting $u^{n+1} = \eta^{n+1}$, $\phi^{n+\frac{1}{2}} = \varphi$, respectively. It is shown in [8] that this scheme is unconditionally stable in the Stokes regime.

2.4. MAC/parallel implementation

We have implemented the algorithms (5)–(8) with the discretization in space done on the MAC (Marker-And-Cell) stencil using central differences for the first-order and second-order derivatives. The algorithm has been implemented in parallel on a Cartesian domain decomposition (see Figure 1) using Message Passing Interface (MPI). Once approximated in space, all the one-dimensional linear problems give rise to tridiagonal linear systems that are solved in parallel using direct solves of the Schur complement induced by the domain decomposition.

The procedure goes as follows. In each direction, the domain partitioning induces a splitting of the unknowns into internal unknowns u_i and interface unknowns, u_e; as a result, a large number of

(a) Communications (b) Processors in charge of the Schur complement for x-solves (c) Processors in charge of the Schur complement for y-solves (d) Processors in charge of the Schur complement for z-solves
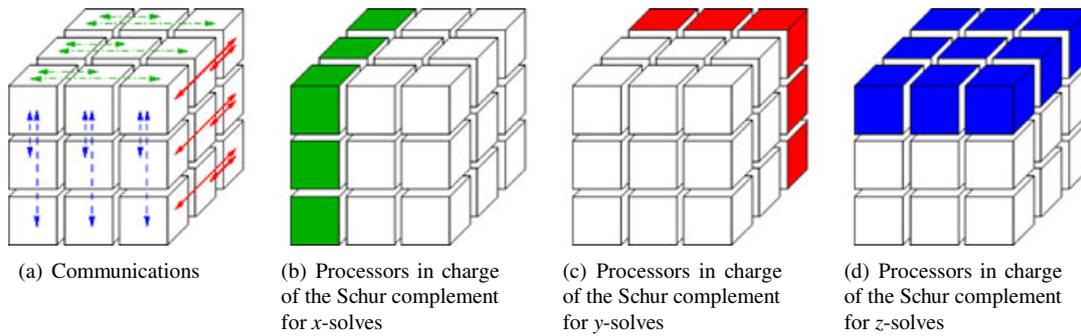
Figure 1. Communications and processors in charge of the Schur complements.

partitioned linear systems of the following type have to be solved at each time step in each space direction d, $d \in \{1, 2, 3\}$:

$$\begin{pmatrix} A_{ii} & A_{ie} \\ A_{ei} & A_{ee} \end{pmatrix} \begin{pmatrix} u_i \\ u_e \end{pmatrix} = \begin{pmatrix} f_i \\ f_e \end{pmatrix}. \tag{9}$$

The solution to the previously mentioned system is constructed by first solving $(A_{ee} - A_{ei} A_{ii}^{-1} A_{ie}) u_e = f_e - A_{ei} A_{ii}^{-1} f_i$ and then by solving $A_{ii} u_i = f_i - A_{ie} u_e$. Denoting by n_e the number of interfaces in the dth direction, the Schur complement, $S := A_{ee} - A_{ei} A_{ii}^{-1} A_{ie}$, is an $n_e \times n_e$ tridiagonal matrix. This matrix is constructed once at the pre-processing stage and stored in one of the P_d processors that are involved in the solve in the dth direction. Denoting by P the total number of processor available, $P_d = P^{\frac{1}{3}}$ if the domain decomposition is isotropic.

More specifically, the algorithm proceeds by first solving the local tridiagonal systems $A_{ii} x_i = f_i$, and then the quantity $A_{ei} x_i$ is communicated to the processor where the Schur complement is stored. On this processor, the quantity $f_e - A_{ei} A_{ii}^{-1} f_i$ is assembled, and the linear system $S u_e = f_e - A_{ei} A_{ii}^{-1} f_i$ is solved. Finally, the quantity $A_{ie} u_e$ is communicated back to each processor, and the linear system $A_{ii} u_i = f_i - A_{ie} u_e$ is solved on each processor. Note that solving the linear system $A_{ii} z_i = g_i$ does not involve any communication. Thus, the total amount of communication per processor in each direction consists of sending a vector of size equal to the number of interface unknowns (each internal domain has two interfaces in the given direction) and receiving the same amount of data. The communication process is illustrated in Figure 1.

Of course, for each direction solve, there is a large number of tridiagonal systems to solve. Actually, denoting by N the total number of grid points and assuming that the domain decomposition is isotropic, there are $n^2 = (N/P)^{2/3}$ linear systems to solve in each row of processors. The solution technique described earlier can be accelerated by storing the Schur complement in each processor in the considered row and assigning each processor to solve $n^2/P^{1/3}$ Schur complement systems instead of assigning the entire work to one processor only.

We have verified that the weak scalability of the code described earlier is quasi-perfect up to the maximum number of processors that were available to us without special request for allocation, that is, 1024 processors.

Extensive numerical tests have shown that the algorithm is stable under CFL condition in the Navier–Stokes regime. Numerical tests using analytical solutions of the Navier–Stokes equations have also shown that this algorithm is second-order accurate with respect to time in the L^2-norm for the velocity.

2.5. Parallelization: Poisson versus direction splitting

The pressure correction in traditional pressure-correction algorithms is computed by solving

$$-\Delta \phi^{n+1} = -\frac{1}{\Delta t} \nabla \cdot \boldsymbol{u}^{n+1}, \qquad \partial_n \phi |_{\partial \Omega \times (0,T]} = 0. \tag{10}$$

For large size problems and large Reynolds numbers, the cost of solving the above-mentioned Poisson equation becomes dominant. In particular, the solution of (10) may not be very easy to parallelize efficiently on very large numbers of processors. In the proposed new method, the pressure correction is evaluated by solving the following sequence of one-dimensional problems:

$$
\begin{aligned}
\psi - \partial_{xx}\psi &= -\frac{1}{\Delta t}\nabla\cdot\boldsymbol{u}^{n+1}, & \partial_x\psi|_{x=0,L_x} &= 0, \\
\varphi - \partial_{yy}\varphi &= \psi, & \partial_y\varphi|_{y=0,L_y} &= 0 \\
\phi^{n+\frac{1}{2}} - \partial_{zz}\phi^{n+\frac{1}{2}} &= \varphi, & \partial_z\phi^{n+\frac{1}{2}}|_{z=0,L_z} &= 0,
\end{aligned}
\tag{11}
$$

which we believe is easier to solve in parallel than (10).

Although it could be argued that FFT methods are very efficient to solve (10), we claim that solving (11) is always more efficient in terms of communications than solving (10) with FFT. To make this point clear, let us assume that we have a cubic domain containing N grid points and that we employ P processors. To the best of our knowledge, all current parallel implementations of FFT-based methods use codimension-one partitioning of the data; that is to say, in three space dimensions, the data are distributed in parallelepipeds, each containing N/P grid points, and the distribution changes at each directional solve (see e.g. [9, 10] for more details on FFT and parallelization). Codimension-one data partitioning is necessary because of the global nature of the one-dimensional FFT. This concept is illustrated in Figure 2 for $P = 4$. After each FFT solve, each processor re-uses only $(N/P)/\sqrt{P}$ data out of the total N/P data that it contains, and the remaining $(1-1/\sqrt{P})N/P$ data must be sent to $P^{1/2} - 1$ other processors so that the one-dimensional FFT can be done in the next direction. Meanwhile, each processor must receive the same amount of data. All in all, each processor sends and receives $2(1 - 1/\sqrt{P})N/P$ data per direction solves, which makes $6(1 - 1/\sqrt{P})N/P$ data exchanges per time step per processor.

A more natural way of distributing the grid points when solving (11) consists of dividing the processors into a cubic Cartesian grid composed of $P^{1/3}$ processors in each direction so that each processor contains n^3 grid points, where $n = (N/P)^{1/3}$, as shown in Figure 3. Then, when using a Schur complement technique to solve the one-dimensional problems, each processor needs to

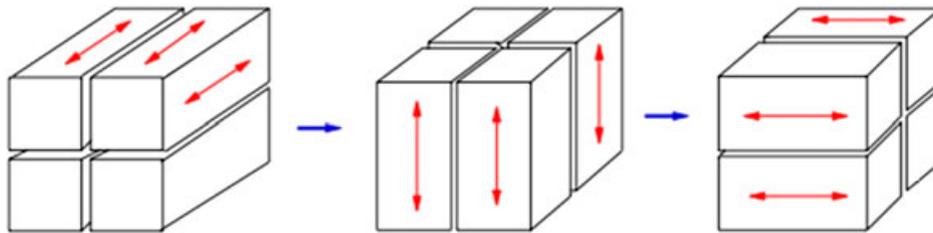Figure 2. Domain partitioning for parallel implementation of FFT with $P = 4$ processors. The arrows indicate the direction in which the transform is performed.
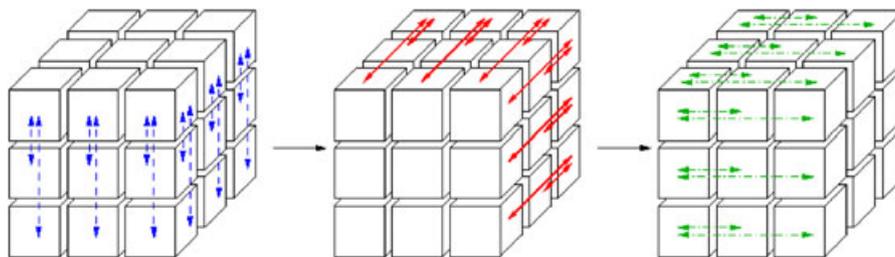

Figure 3. Domain partitioning for parallel implementation of the direction splitting algorithm using the Schur complement technique on $P = 27$ processors. The arrows indicate the communications that are involved in each direction solve.

exchange only the interface data with the processor that is in charge of solving the Schur complement system. Thus, the total amount of data exchange (sending and receiving) is not more than $3 \times 4 \times n^2 = 12(N/P)^{2/3}$ per processor per time step; see Figure 3.

It then becomes clear that the number of communications per processor per time step for FFT solves, $6(1 - 1/\sqrt{P})N/P$, is significantly larger than $12(N/P)^{2/3}$. For instance, if $N/P = 10^6$ and $P = 1000$, then $\frac{1}{2}(1 - 1/\sqrt{P})(N/P)^{\frac{1}{3}} \approx 49$; in this case, the FFT solution of (10) involves 50 times more communications than solving (11) with a Schur complement technique.

3. VALIDATION OF THE METHOD

The purpose of this section is to validate our numerical technique by comparing two-dimensional and three-dimensional results produced by the algorithm (5)–(8) with well-established driven cavity benchmark data. Geometric descriptions of the two-dimensional and three-dimensional cavities are shown in Figure 4.

3.1. Steady two-dimensional cavity flow at Re = 1000

We start with steady-state computations in the two-dimensional driven cavity $\Omega = (0, 1)^2$ at Reynolds number 1000. The problem is formulated as follows:

$$\begin{cases} \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla \mathbf{u}) - \dfrac{1}{Re}\Delta \mathbf{u} + \nabla p = 0 \quad \text{in } \Omega \times (0, T], \qquad \mathbf{u}|_{t=0} = 0 \quad \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times [0, T], \\ \mathbf{u}|_{x=0} = 0, \ \mathbf{u}|_{x=L_x} = \boldsymbol{e}_y, \ \mathbf{u}|_{y=0, L_y} = 0, \quad \text{in } (0, T] \end{cases} \tag{12}$$

where (x, y) denote the Cartesian coordinates and $\boldsymbol{e}_y$ is the unit vector in the y-direction. This is a driven cavity problem where the side $\{L_x\} \times [0, L_y]$ slides upward with unit speed. The velocity is non-dimensionalized with respect to the velocity of the lid; see Figure 4.

Beside obvious scientific and engineering applications, flows in lid-driven cavities are among the configurations that are the most frequently used to validate numerical codes. It is therefore very important to have highly accurate numerical data that can be used for such validations. Because two-dimensional flows in rectangular cavities are far less demanding in terms of computing power than their three-dimensional counterparts, there are many articles in the literature that provide benchmark
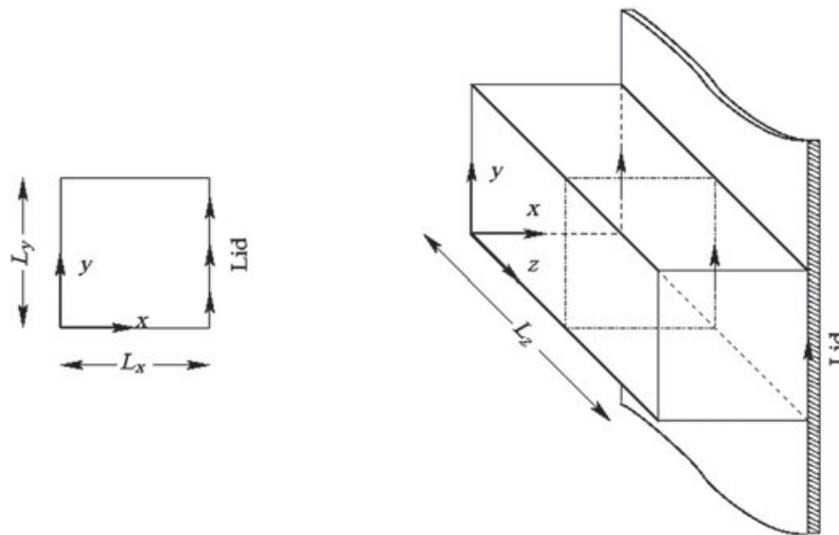

Figure 4. Notation for two-dimensional cavity (left) and three-dimensional cavity (right).

data in the two-dimensional case. Ghia *et al.* [11] provide a comprehensive analysis of steady solutions in the square cavity up to Reynolds number $Re = 10{,}000$. The computations reported in [11] used grids composed of 257×257 nodes and a multigrid solver to solve the steady two-dimensional Navier–Stokes equations formulated in terms of stream function and vorticity, that is, $\psi-\omega$.

We first make comparisons with the benchmark results of [2, 3]. The data from [2] have been obtained from a highly accurate spectral method using 160 spectral modes in each direction. Special care has been given to remove the corner singularities. The results of [3] have been computed on a uniform second-order finite difference grid of 1024×1024 nodes. Our computations have been done on a uniform grid composed of 5000×5000 nodes with time step $\Delta t = 10^{-4}$. The computation at $Re = 1000$ has been run up to $T = 125$. We do not claim having reached complete steady state because doing so would require developing a radically different strategy based on Newton iterations, which is not the purpose of the present paper. By monitoring the time evolution of the solutions and repeating the computation on various nested grids, we have come to the conclusion that at least the first four (and possibly five digits) of our results should be correct.

We compare in Table I the vertical and horizontal components of the velocity along the vertical and horizontal mid-sections of the cavity at points that have been selected in [2]. Because our results are computed at points that generally differ from those used by the other studies, our results are interpolated at the required points by piecewise linear interpolation. It is clear that our results coincide with the spectrally accurate results of [2] up to the fifth digit. Because the results of [2] and ours have been obtained by totally different techniques, we conjecture that our results are correct up to $\pm 5 \times 10^{-6}$ at $Re = 1000$. Note that the results from [3] seem to be converged to the fourth digit only, that is, the accuracy of the results from [3] seems to be correct up to $\pm 1 \times 10^{-4}$.

3.2. Steady three-dimensional cavity flows at $Re = 1000$

We now turn our attention to the three-dimensional driven cavity problem formulated as follows:

$$
\begin{cases}
\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla \mathbf{u}) - \dfrac{1}{Re}\Delta \mathbf{u} + \nabla \mathrm{p} = 0 & \text{in } \Omega \times (0, T], \\
\nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \times [0, T], \\
\mathbf{u}|_{x=0} = 0, \mathbf{u}|_{x=L_x} = \boldsymbol{e}_y, \mathbf{u}|_{y=0,L_y} = 0, \mathbf{u}|_{z=0,L_z} = 0, & \text{in } (0, T] \\
\mathbf{u}|_{t=0} = 0 & \text{in } \Omega,
\end{cases}
\tag{13}
$$

Table I. Comparison of the vertical/horizontal components of the velocity along the segment $\{x \in [0, 1], y = 1/2\}/\{y \in [0, 1], x = 1/2\}$ at $Re = 1000$ with the results of [2, 3].

	$Re = 1000$, vertical component				$Re = 1000$, horizontal component		
x	[3]	[2]	Present	y	[3]	[2]	Present
1.0000	1.00000	1.0000000	1.0000000	0.0000	1.00000	1.0000000	1.0000000
0.9766	NA	0.6644227	0.6644194	0.0312	NA	−0.2279225	−0.2279177
0.9688	0.58031	0.5808359	0.5808318	0.0391	−0.29330	−0.2936869	−0.2936814
0.9609	NA	0.5169277	0.5169214	0.0469	NA	−0.3553213	−0.3553154
0.9531	0.47239	0.4723329	0.4723260	0.0547	−0.41018	−0.4103754	−0.4103691
0.8516	NA	0.3372212	0.3372128	0.0937	NA	−0.5264392	−0.5264320
0.7344	0.18861	0.1886747	0.1886680	0.1406	−0.42634	−0.4264545	−0.4264492
0.6172	NA	0.0570178	0.0570151	0.1953	NA	−0.3202137	−0.3202068
0.5000	−0.06205	−0.0620561	−0.0620535	0.5000	0.02580	0.0257995	0.2579868
0.4531	NA	−0.1081999	−0.1081955	0.7656	NA	0.3253592	0.3253529
0.2813	−0.28040	−0.2803696	−0.2803632	0.7734	0.33398	0.3339924	0.3339860
0.1719	NA	0.3885691	−0.3885624	0.8437	NA	0.3769189	0.3769119
0.1016	−0.30029	−0.3004561	−0.3004504	0.9062	0.33290	0.3330442	0.3330381
0.0703	NA	−0.2228955	−0.2228928	0.9219	NA	0.3099097	0.3099041
0.0625	−0.20227	−0.2023300	−0.2023277	0.9297	0.29622	0.2962703	0.2962650
0.0547	NA	−0.1812881	−0.1812863	0.9375	NA	0.2807056	0.2807005
0.0000	0.00000	0.0000000	0.0000000	1.0000	0.00000	0.0000000	0.0000000

NA indicates that the data at this point are not available in this reference.

where (x, y, z) denote the Cartesian coordinates and $\boldsymbol{e}_y$ is the unit vector in y-direction. The side $\{L_x\} \times [0, L_y] \times [0, L_z]$ of the cavity slides upward with unit speed. The velocity is non-dimensionalized with respect to the velocity of the lid; see Figure 4.

We validated our code in three space dimensions by computing the steady-state flow in a three-dimensional cavity $\Omega = (0, 1) \times (0, 1) \times (0, 2)$ at $Re = 1000$. This flow has been studied by many authors; detailed and comprehensive results are reported in the paper of Albensoeder and Kuhlmann [4] (see the references therein for a thorough list of articles focusing on the steady lid-driven cavity flow). The reference [4] provides extensive data at $Re = 1000$ for various cavity aspect ratios using a pseudo-spectral method on grids composed of up to 96^3 grid points.

We have computed the solution to the time-dependent problem up to $T=250$, and we expect the results to be close to steadiness within a tolerance of order $e^{-250/\sqrt{1000}} \approx 4 \times 10^{-4}$. The computations are done with time step $\Delta t = 1.5 \times 10^{-3}$ on a non-uniform grid composed of $400 \times 400 \times 800$ points defined as follows. Let N_x be the number of grid points in the x direction, then the coordinates of the grid points in the x direction, say x_i, $i = 1, \ldots, N_x$, are defined by the following formula:

$$C_x = 1 + (2/L_x)^{\frac{1}{2}} \tag{14}$$

$$X_i = \frac{(i-1)}{N_x - 1} L_x, \quad i = 1, \ldots, N_x \tag{15}$$

$$x_i = \begin{cases} C_x X_i^{\frac{3}{2}} (1 + X_i^{\frac{1}{2}})^{-1} & \text{if } X_i \leq L_x/2 \\ L_x - C_x (L_x - X_i)^{\frac{3}{2}} (1 + (L_x - X_i)^{\frac{1}{2}})^{-1} & \text{otherwise.} \end{cases} \tag{16}$$

We use similar formulae to distribute the grid points in the y and z directions. The purpose of this grid arrangement is to resolve well the boundary layers without having extremely stretched grid cells at the boundary, which cosine and sine distributions would produce when N_x is large (see e.g. [12] where cosine distributions are used with moderate N_x).

The evaluation of the velocity profiles at the benchmark locations that are reported in [4] is done by piecewise linear interpolation from our $400 \times 400 \times 800$ grid. We show in Table II values of the vertical component of the velocity along the segment $[0, 1] \times \{1/2\} \times \{1\}$ and values of the horizontal component along the segment $\{1/2\} \times [0, 1] \times \{1\}$. The results from [4] and ours agree up to the third digit and differ very slightly in the fourth digit at some points.

Table II. Velocity profiles in median plane $\{z = 1\}$ at $Re = 1000$.

Vertical velocity, $y = 1/2$, $z = 1$, $Re = 1000$			Horizontal velocity, $x = 1/2$, $z = 1$, $Re = 1000$		
x	[4]	Present	y	[4]	Present
1.0000	1.0000000	1.000000	1.0000	0.0000000	0.00000
0.9766	0.6081626	0.608220	0.9688	0.2103534	0.21040
0.9688	0.5111673	0.511170	0.9609	0.2685775	0.26863
0.9609	0.4352632	0.435310	0.9531	0.3229981	0.32306
0.9531	0.3800975	0.380120	0.9453	0.3716805	0.37173
0.8516	0.2124395	0.212340	0.9063	0.4720530	0.47176
0.7344	0.1037698	0.103720	0.8594	0.3472198	0.34697
0.6172	0.0333083	0.033307	0.8047	0.2042160	0.20416
0.5000	-0.0129662	-0.012960	0.5000	0.0006955	0.00070
0.4531	-0.0305044	-0.030500	0.2344	-0.2378601	-0.23784
0.2813	-0.1385097	-0.138540	0.2266	-0.2461943	-0.24618
0.1719	-0.2937825	-0.293660	0.1563	-0.2924917	-0.29239
0.1016	-0.3054909	-0.305220	0.0938	-0.2773856	-0.27705
0.0703	-0.2586761	-0.258430	0.0781	-0.2627359	-0.26234
0.0625	-0.2418095	-0.241580	0.0703	-0.2528278	-0.25240
0.0547	-0.2225479	-0.222340	0.0625	-0.2405949	-0.24015
0.0000	0.0000000	0.000000	0.0000	0.0000000	0.00000

3.3. Transient in the three-dimensional cavity at $Re = 1000$

Validation of the code on transient solutions in cavities is a little more problematic than for steady solutions because of the lack of accurate data available in the literature. For instance, transient solutions in a cubic cavity have been computed in [13] by means of a least-squares finite element discretization of the velocity–vorticity–pressure formulation. This reference presents results at $Re = 1000, 2000$, and 3200 on grids composed of up to $61 \times 61 \times 30$ $\mathbb{P}_1$ hexahedral finite elements. Unfortunately, the data therein are reported in a qualitative manner in the form of graphs and pictures making accurate comparisons difficult. We refer to [14] for a review on the dynamics in three-dimensional cavities of various aspect ratios. Plots of transient velocity profiles can be found therein but again none of these results satisfy the high accuracy requirements that we are looking for.

We have compared our results with those from [12] (see Figure 5) and found reasonable agreement with maximum deviation of order 5%. The results from [12] have been qualitatively cross validated in [4, Figure 4] and have been produced on a $\mathbb{P}_2 - \mathbb{P}_1$ finite element cosine-type grid composed of 456,300 tetrahedra for the half cavity, representing $\approx 600{,}000$ $\mathbb{P}_2$ grid points for the velocity field on the entire cavity. We suspect that the 5% maximum difference observed comes from the coarse grid resolution used in [12]; our finest grid has 3000 and 2000 times more grid points than those used in [12] and [4], respectively.

3.4. Unsteady versus steady two-dimensional cavity flows at high Reynolds numbers

Steady-state results for the two-dimensional driven cavity are reported by many authors. Many of them also attempted to determine the critical Reynolds number at which the Hopf bifurcation occurs, and there is a wide variety of claims in the literature about this critical value. Baragy and Carey [15] obtained steady results up to $Re = 12{,}500$ by solving the $\psi - \omega$ equations with high-resolution p finite elements. In [16], it is claimed that steady state can be reached at $Re = 21{,}000$ on a 600×600 grid. On the other hand, the linear stability study of Ramanan and Homsy [17] indicates that the two-dimensional cavity flow is unstable to two-dimensional perturbations when the Reynolds number is of order 7000. By using a highly accurate spectral method with regularization of the corner singularities and by carefully studying the energy behavior of the flow, it is concluded in [18] that the Hopf bifurcation must occur at a Reynolds number comprised between 8017.6 and 8018.8. This result is consistent with the linear stability analysis of [3], which indicates that the critical Reynolds number is in the range of $8000 < Re < 8050$.

(a) First component of velocity along segment $\{x = \frac{1}{2}, y \in [0, 1], z = 1\}$ at $t = 4, 8, 12$.

(b) Second component of velocity along segment $\{x = [0, 1], y = \frac{1}{2}, z = 1\}$ at $t = 4, 8, 12$.
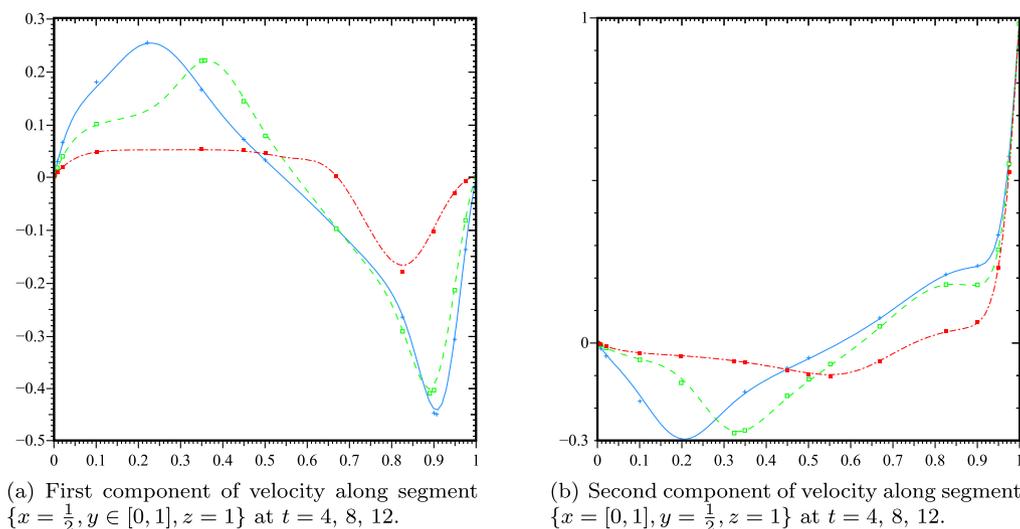
Figure 5. Comparisons between results from [12] (lines) and results from Tables III and IV (symbols) at $Re = 1000$.

The objective of the present study is not to confirm or contradict any of the claims mentioned earlier but we would like to mention in passing that our time-dependent simulations up to $T = 250$ at $Re = 12,500$ and $Re = 21,000$ on 600×600 and 5000×5000 grids using time steps $\Delta t = 0.0015$ and 10^{-4}, respectively, did not converge to steady state. In each case, the time records show unsteady oscillatory solutions with large amplitude that do not seem to decrease as time grows. One possible explanation for the steady states reported in [15, 16] at $Re = 12,500$ is that the semi-implicit pseudo-time integration techniques used therein follow an unstable steady branch of the solution after the Hopf bifurcation has occurred. We suspect that the numerical techniques that use pseudo-time stepping and semi-implicit approximations with large time steps bypass the Hopf bifurcation happening at a smaller Reynolds number and produce instead one trivial branch of the bifurcated solution.

4. TIME-DEPENDENT BENCHMARK IN THE THREE-DIMENSIONAL CAVITY

The previous discussion shows that benchmarking codes and algorithms on steady-state solutions may be controversial. Moreover, reaching steadiness requires a very long time integration or specialized Newton-time iterative algorithms. Finally, testing steady solutions quantifies only the spatial accuracy and says nothing about the time accuracy. To avoid the steady-state ambiguity and the long-time integration waste, and to quantify both the space and time accuracy of algorithms, we provide here benchmark data for the early unsteady development of the flow in a three-dimensional cavity. Contrary to more traditional steady-state benchmarks, time-dependent benchmarks both validate the space and the time accuracy and eliminate the long-time integration waste of resource.

The purpose of this section is to report highly accurate results for the velocity field in the three-dimensional driven cavity $\Omega = (0, 1) \times (0, 1) \times (0, 2)$ in the transient regime at times $t = 4, 8,$ and 12 at $Re = 1000$ and $Re = 5000$.

4.1. Parameters of the discretization and accuracy assessment

The computations are done on a non-uniform grid composed of $1000 \times 1000 \times 2000$ MAC cells with time step $\Delta t = 5 \times 10^{-4}$ for $Re = 1000$ and 4×10^{-4} at $Re = 5000$. The parameter χ is set to 1 in all cases. The mesh is refined using the grid distribution defined in (14)–(16).

Comparisons between results at $Re = 1000$ produced on a non-uniform grid composed of $400 \times 400 \times 800$ MAC cells and with time step 1.5×10^{-3}, and a grid of $800 \times 800 \times 1600$ cells and a time step of 6.25×10^{-4} revealed that the results on the latter grid are converged up to the fourth digit at least. Further comparisons between the results on the $800 \times 800 \times 1600$ grid and a grid of $1000 \times 1000 \times 2000$ MAC cells for the horizontal and vertical velocity profiles on the lines $x = 1/2, z = 1$ and $y = 1/2, z = 1$ at $Re = 1000$ are presented in the top panels of Figure 6. The results from the finer grid are interpolated at the nodes of the coarser grid using piecewise linear interpolation. This comparison suggests that the first four digits of the $1000 \times 1000 \times 2000$ results are correct for $t = 4$, $t = 8$, and $t = 12$.

The difference in the velocity profiles at $Re = 5000$ on the $800 \times 800 \times 1600$ and $1000 \times 1000 \times 2000$ grids with time steps $\tau = 5 \times 10^{-4}$ and $\tau = 4 \times 10^{-4}$ are reported in the bottom panels of Figure 6. The two velocities differ only in the fifth digit at $t = 4$ and in the fourth digit at $t = 8$ at most locations, but at $t = 12$, there are differences in the third digit.

4.2. Profiles

We report in Tables III–X various velocity profiles in the cavity along the segments $\{x = 1/2, y \in [0, 1], z = 1\}$, $\{x \in [0, 1], y = 1/2, z = 1\}$, and $\{x = 1/2, y = 1/2, z \in [0, 2]\}$ at times $t = 4, 8,$ and 12 and at $Re = 1000$ and 5000.

The complete set of data at each grid point for the profiles presented in Tables III–X as well as the data for the pressure distribution in the corresponding planes can be found at the following web site http://www.math.ualberta.ca/~pminev/cavity.
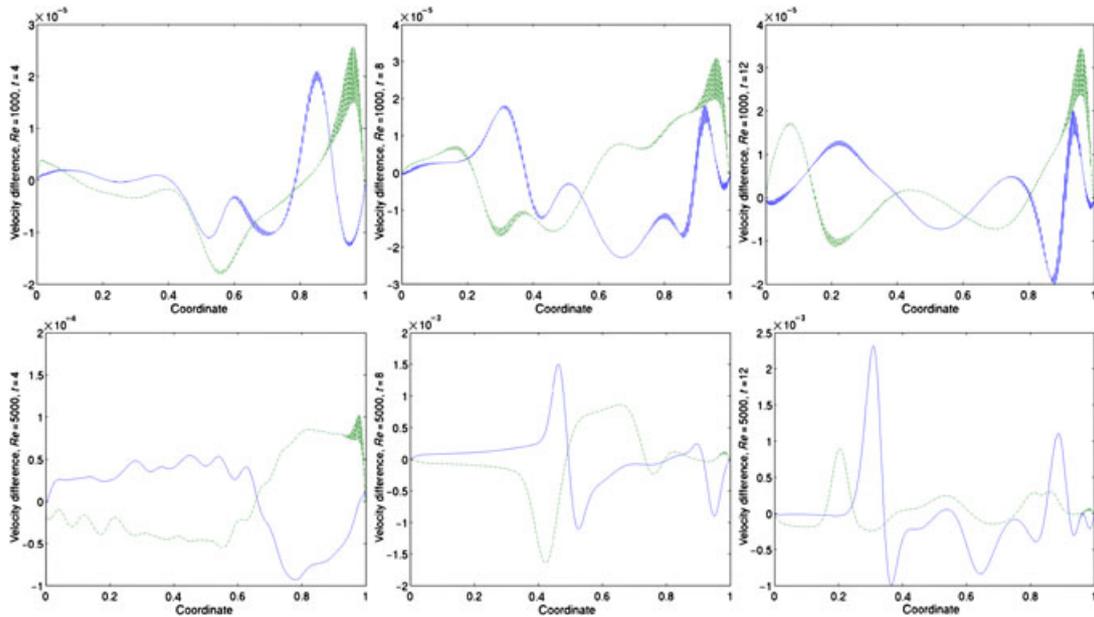
Figure 6. Difference in horizontal (continuous line) and vertical (dashed line) velocity profiles on segment $\{x = 1/2, y \in [0, 1], z = 1\}$ and $\{x \in [0, 1], y = 1/2, z = 1\}$ between grid $800 \times 800 \times 1600$, $\tau = 6.25 \times 10^{-4}$, and grid $1000 \times 1000 \times 2000$, $\tau = 5 \times 10^{-4}$, at $Re = 1000$ (top) and $\tau = 5 \times 10^{-4}$, $\tau = 4 \times 10^{-4}$ correspondingly at $Re = 5000$ (bottom); $t = 4$ (left), 8 (center), and 12 (right).

Table III. First component of the velocity along the segment $\{x = 1/2, y \in [0, 1], z = 1\}$, at $Re = 1000$ for $t = 4$, 8, and 12.

	$x = 1/2, z = 1, Re = 1000$		
y	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
1.4034714e−04	1.4508133e−04	3.0865118e−04	5.2210372e−04
2.0387121e−03	2.0737393e−03	4.4125478e−03	7.4674594e−03
8.0061410e−03	7.7452512e−03	1.6486075e−02	2.7938534e−02
2.0521140e−02	1.7880155e−02	3.8051667e−02	6.4647070e−02
1.0064008e−01	4.6696601e−02	9.9178370e−02	1.7848742e−01
2.2124428e−01	5.1016412e−02	1.3002488e−01	**2.5322428e−01**
3.4925492e−01	**5.1498407e−02**	2.1882494e−01	1.6413073e−01
3.5771066e−01	5.1491324e−02	**2.1964293e−01**	1.5526669e−01
4.4944092e−01	4.9177857e−02	1.4235761e−01	7.0067168e−02
5.0064629e−01	4.4011313e−02	7.6719600e−02	3.1056902e−02
6.6872504e−01	7.5696050e−04	−9.9177224e−02	−9.8378888e−02
8.2575599e−01	**−1.8011280e−01**	−2.9331655e−01	−2.6639041e−01
8.9010354e−01	−1.2061613e−01	**−4.1036179e−01**	−4.3364671e−01
9.0027484e−01	−1.0405456e−01	−4.0451907e−01	−4.4774158e−01
9.0662275e−01	−9.3584182e−02	−3.9450664e−01	**−4.5032835e−01**
9.4947219e−01	−3.1599145e−02	−2.1532683e−01	−3.0853427e−01
9.7509665e−01	−8.8443544e−03	−8.3150307e−02	−1.3854079e−01
9.9908073e−01	−7.1766561e−05	−1.9895024e−03	−4.0488981e−03
1.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00

The maximum/minimum values of each profile are presented in boldface.

5. CONCLUSIONS

We have validated the new direction splitting algorithm introduced in [1, 8] for solving the time-dependent incompressible Navier–Stokes equation. This algorithm only requires solving one-dimensional problems. It is currently limited to simple Cartesian geometries, but generalizations

Table IV. Second component of the velocity along the segment $x \in [0, 1]$, $y = 1/2$, $z = 1$, at $Re = 1000$ for $t = 4$, 8, and 12.

	$y = 1/2$, $z = 1$, $Re = 1000$		
x	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
1.4034714e−04	−8.2444153e−05	−1.1612837e−04	−3.0434609e−04
2.0387121e−03	−1.1814422e−03	−1.6679658e−03	−4.3678162e−03
8.0061410e−03	−4.4477533e−03	−6.3267108e−03	−1.6557052e−02
2.0521140e−02	−1.0441183e−02	−1.5109641e−02	−3.9864610e−02
1.0064008e−01	−3.0647620e−02	−5.1586542e−02	−1.7888451e−01
1.9850609e−01	−3.9971558e−02	−1.2238869e−01	**−2.9845315e−01**
3.2413757e−01	−5.6048571e−02	**−2.7690680e−01**	−1.7516767e−01
3.4925492e−01	−6.0720895e−02	−2.6864547e−01	−1.5005601e−01
4.4944092e−01	−8.3945004e−02	−1.6237848e−01	−7.7253657e−02
5.0064629e−01	−9.5903784e−02	−1.1142934e−01	−4.5571855e−02
5.5182543e−01	**−1.0169440e−01**	−6.4728958e−02	−1.2689309e−02
6.6872504e−01	−5.6109400e−02	5.1185578e−02	7.7361160e−02
8.2575599e−01	3.6110262e−02	1.7998007e−01	2.1094945e−01
9.0027484e−01	6.3777573e−02	1.7891472e−01	2.3724457e−01
9.4947219e−01	2.3043437e−01	2.8716755e−01	3.3224791e−01
9.7509665e−01	5.2690636e−01	5.5152071e−01	5.7434675e−01
9.9908073e−01	9.8114695e−01	9.8221866e−01	9.8308774e−01
1.0000000e+00	**1.0000000e+00**	**1.0000000e+00**	**1.0000000e+00**

The maximum/minimum values of each profile are presented in boldface.

Table V. First component of the velocity along the segment $\{x = 1/2, y = 1/2, z \in [0, 2]\}$, at $Re = 1000$ for $t = 4$, 8, and 12.

	$x = 1/2$, $y = 1/2$, $Re = 1000$		
z	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	**0.0000000e+00**	**0.0000000e+00**	**0.0000000e+00**
1.1626738e−04	2.3916656e−04	6.9695004e−04	6.3805342e−04
5.1359941e−03	9.8463370e−03	2.9291910e−02	2.6946040e−02
5.0385478e−02	5.1896922e−02	1.7849731e−01	1.6711660e−01
7.8102419e−02	5.6389753e−02	2.0708011e−01	**1.8618114e−01**
8.1858741e−02	**5.6440430e−02**	2.0857947e−01	1.8581070e−01
9.1041204e−02	5.6210738e−02	**2.1011046e−01**	1.8236238e−01
2.0069752e−01	4.3894856e−02	9.5074118e−02	5.5718368e−02
4.0086680e−01	4.4759855e−02	1.1413365e−01	4.5423388e−02
6.0027520e−01	4.5147427e−02	1.0773139e−01	3.6912602e−02
8.0039503e−01	4.4221491e−02	8.6504399e−02	2.7131040e−02
1.0006249e+00	4.4011313e−02	7.6719600e−02	3.1056902e−02
1.2008100e+00	4.4224604e−02	8.6624634e−02	2.7111414e−02
1.4008714e+00	4.5153831e−02	1.0782841e−01	3.7043443e−02
1.6001973e+00	4.4742780e−02	1.1406359e−01	4.5365408e−02
1.8002273e+00	4.3963703e−02	9.5732433e−02	5.6241196e−02
1.9089588e+00	5.6210738e−02	**2.1011046e−01**	1.8236238e−01
1.9181413e+00	**5.6440430e−02**	2.0857947e−01	1.8581070e−01
1.9218976e+00	5.6389753e−02	2.0708011e−01	**1.8618114e−01**
1.9502844e+00	5.1664283e−02	1.7736842e−01	1.6612522e−01
1.9952203e+00	9.2090395e−03	2.7357442e−02	2.5157873e−02
1.9998837e+00	2.3916656e−04	6.9695004e−04	6.3805342e−04
2.0000000e+00	**0.0000000e+00**	**0.0000000e+00**	**0.0000000e+00**

The maximum/minimum values of each profile are presented in boldface.

Table VI. Second component of the velocity along the segment $\{x = 1/2, y = 1.2, z \in [0, 2]\}$, at $Re = 1000$ for $t = 4$, 8, and 12.

	$x = 1/2$, $y = 1/2$, $Re = 1000$		
z	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	**0.0000000e+00**	0.0000000e+00	0.0000000e+00
1.1626738e−04	−5.2585217e−05	2.4839746e−04	1.4967061e−04
5.1359941e−03	−2.4969248e−03	9.7988873e−03	5.4423061e−03
1.5504534e−02	−8.4019616e−03	2.2590113e−02	**9.6174650e−03**
2.7266599e−02	−1.5918923e−02	**2.7253961e−02**	5.2606189e−03
5.0385478e−02	−3.1026970e−02	1.4165686e−02	−2.1245248e−02
1.6267736e−01	−8.2909541e−02	−1.9805112e−01	**−1.5654960e−01**
1.9057506e−01	−9.0721179e−02	**−2.1308555e−01**	−1.4871038e−01
2.0069752e−01	−9.2931448e−02	−2.1131140e−01	−1.4323771e−01
2.8534467e−01	**−9.9953893e−02**	−1.5328221e−01	−9.3412059e−02
4.0086680e−01	−9.7571062e−02	−1.2093993e−01	−5.0125610e−02
6.0027520e−01	−9.6713492e−02	−1.2135150e−01	−4.6338516e−02
8.0039503e−01	−9.6353229e−02	−1.1261272e−01	−5.0640390e−02
1.0006249e+00	−9.5903784e−02	−1.1142934e−01	−4.5571855e−02
1.2008100e+00	−9.6357471e−02	−1.1263860e−01	−5.0655132e−02
1.4008714e+00	−9.6712714e−02	−1.2139948e−01	−4.6284343e−02
1.6001973e+00	−9.7590342e−02	−1.2098099e−01	−5.0365455e−02
1.7146553e+00	**−9.9953893e−02**	−1.5328221e−01	−9.3412059e−02
1.8002273e+00	−9.2743819e−02	−2.1159869e−01	−1.4377137e−01
1.8094249e+00	−9.0721179e−02	**−2.1308555e−01**	−1.4871038e−01
1.8373226e+00	−8.2909541e−02	−1.9805112e−01	**−1.5654960e−01**
1.9502844e+00	−3.0603230e−02	1.4864842e−02	−2.0259999e−02
1.9727334e+00	−1.5918923e−02	**2.7253961e−02**	5.2606189e−03
1.9844955e+00	−8.4019616e−03	2.2590113e−02	**9.6174650e−03**
1.9952203e+00	−2.3129445e−03	9.1958475e−03	5.1407613e−03
1.9998837e+00	−5.2585217e−05	2.4839746e−04	1.4967061e−04
2.0000000e+00	**0.0000000e+00**	0.0000000e+00	0.0000000e+00

The maximum/minimum values of each profile are presented in boldface.

Table VII. First component of the velocity along the segment $\{x = 1/2, y \in [0, 1], z = 1\}$, at $Re = 5000$ for $t = 4$, 8, and 12.

	$x = 1/2$, $z = 1$, $Re = 5000$		
y	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
1.4034714e−04	1.0953564e−04	2.2201547e−04	3.8426242e−04
2.0387121e−03	1.5439502e−03	3.1286679e−03	5.4333906e−03
8.0061410e−03	5.5197936e−03	1.1184627e−02	1.9601340e−02
2.0521140e−02	1.1638065e−02	2.3624778e−02	4.1938019e−02
1.0709938e−01	**2.0250011e−02**	4.3247461e−02	7.7916647e−02
3.4084031e−01	1.9095604e−02	5.7284717e−02	**2.4310688e−01**
4.4944092e−01	1.7241733e−02	1.1500871e−01	1.4033986e−01
4.9806146e−01	1.5217571e−02	**2.1556927e−01**	1.0601027e−01
6.6872504e−01	−1.1174660e−02	−1.0739620e−02	−9.2954442e−02
8.2679475e−01	**−4.8657653e−02**	−2.0069005e−01	−2.1806916e−01
9.0027484e−01	−4.2086790e−02	−3.0391051e−01	−2.3011339e−01
9.2332607e−01	−3.8148118e−02	**−3.7289989e−01**	−2.9769941e−01
9.5253947e−01	−2.9691275e−02	−2.1178171e−01	**−4.3410527e−01**
9.7509665e−01	−1.7925170e−02	−4.7367244e−02	−2.7481071e−01
9.9908073e−01	−6.3542999e−04	2.6645131e−04	−5.5758658e−03
1.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00

The maximum/minimum values of each profile are presented in boldface.

Table VIII. Second component of the velocity along the segment $x \in [0, 1]$, $y = 1/2$, $z = 1$, at $Re = 5000$ for $t = 4$, 8, and 12.

	$y = 1/2, z = 1, Re = 5000$		
x	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
1.4034714e−04	−7.8951934e−05	−1.4888235e−04	−2.1538831e−04
2.0387121e−03	−1.1086342e−03	−2.1052791e−03	−3.0507066e−03
8.0061410e−03	−3.9225668e−03	−7.6120878e−03	−1.1095946e−02
2.0521140e−02	−8.1672575e−03	−1.6501923e−02	−2.4390783e−02
1.0064008e−01	−1.4637202e−02	−3.4410468e−02	−5.3750248e−02
2.4114697e−01	−1.8106839e−02	−4.7676083e−02	**−2.7704685e−01**
3.4925492e−01	−2.3160505e−02	−7.2872041e−02	−1.6557463e−01
4.6341611e−01	−3.0352982e−02	**−1.7524624e−01**	−7.8283783e−02
5.0064629e−01	−3.2620349e−02	−1.5461735e−01	−4.2887045e−02
5.8075701e−01	**−3.5269859e−02**	−6.1226153e−02	3.0510197e−02
6.6872504e−01	−3.0056446e−02	4.7583579e−02	8.9026616e−02
8.2575599e−01	−1.3680025e−04	7.4403155e−02	1.6804707e−01
9.0027484e−01	9.9327726e−03	7.5520824e−02	1.0416261e−01
9.4947219e−01	2.9825496e−02	8.2504271e−02	1.0430217e−01
9.7509665e−01	1.9907800e−01	2.2057524e−01	2.3231298e−01
9.9908073e−01	9.5838610e−01	9.5911207e−01	9.5941283e−01
1.0000000e+00	**1.0000000e+00**	**1.0000000e+00**	**1.0000000e+00**

The maximum/minimum values of each profile are presented in boldface.

Table IX. First component of the velocity along the segment $\{x = 1/2, y = 1/2, z \in [0, 2]\}$, at $Re = 5000$ for $t = 4$, 8, and 12.

	$x = 1/2, y = 1/2, Re = 5000$		
z	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	**0.0000000e+00**	**0.0000000e+00**	**0.0000000e+00**
1.1626738e−04	1.3072231e−04	1.1906843e−03	1.9477803e−03
5.1359941e−03	5.0786622e−03	4.4102217e−02	7.7953402e−02
3.1949132e−02	1.6624588e−02	1.2204385e−01	**2.3283080e−01**
6.1395893e−02	**1.8188951e−02**	1.5881491e−01	2.0744271e−01
9.6501915e−02	1.7777070e−02	2.1175088e−01	1.6913843e−01
2.0069752e−01	1.6422606e−02	2.0158864e−01	1.7391391e−01
4.0086680e−01	1.5316139e−02	2.0454002e−01	1.5042217e−01
6.0027520e−01	1.5182436e−02	2.1360982e−01	1.0931967e−01
8.9406943e−01	1.5088018e−02	**2.1565678e−01**	9.3570671e−02
1.0006249e+00	1.5072556e−02	2.1535840e−01	1.0407517e−01
1.1059306e+00	1.5088018e−02	**2.1565678e−01**	9.3570671e−02
1.4008714e+00	1.5182628e−02	2.1365667e−01	1.0956619e−01
1.6001973e+00	1.5318571e−02	2.0383110e−01	1.5026902e−01
1.8002273e+00	1.6431193e−02	2.0152264e−01	1.7408229e−01
1.9034981e+00	1.7777070e−02	2.1175088e−01	1.6913843e−01
1.9386041e+00	**1.8188951e−02**	1.5881491e−01	2.0744271e−01
1.9680509e+00	1.6624588e−02	1.2204385e−01	**2.3283080e−01**
1.9952203e+00	4.7692721e−03	4.1563665e−02	7.3077751e−02
1.9998837e+00	1.3072231e−04	1.1906843e−03	1.9477803e−03
2.0000000e+00	**0.0000000e+00**	**0.0000000e+00**	**0.0000000e+00**

The maximum/minimum values of each profile are presented in boldface.

to more complex geometries using one-dimensional grid fitting and fictitious domain techniques are currently being developed. This algorithm has enormous parallelization capabilities and should scale very well on massively parallel computers composed of more than 10^4 processors.

The algorithm has been validated on the start-up flow in a three-dimensional lid-driven cavity of size $1 \times 1 \times 2$. Using up to 1000 processors on the distributed Linux Hurr HPC cluster of IAMCS at

Table X. Second component of the velocity along the segment $\{x = 1/2, y = 1/2, z \in [0, 2]\}$, at $Re = 5000$ for $t = 4$, 8, and 12.

	$x = 1/2, y = 1/2, Re = 5000$		
z	$t = 4$	$t = 8$	$t = 12$
0.0000000e+00	**0.0000000e+00**	0.0000000e+00	0.0000000e+00
1.1626738e−04	−8.7124330e−05	8.1334795e−04	1.1464529e−03
5.1359941e−03	−3.6874993e−03	2.8761929e−02	3.4090661e−02
9.0700068e−03	−6.2864708e−03	4.1986175e−02	**4.0536339e−02**
2.0630611e−02	−1.2761813e−02	**5.3685435e−02**	1.0839406e−02
5.0385478e−02	−2.2238806e−02	3.2720764e−02	−5.7712142e−02
9.6501915e−02	−2.6190153e−02	−1.9274035e−01	**−1.3029716e−01**
1.2485572e−01	−2.7082594e−02	**−2.3021265e−01**	−1.1338662e−01
2.0069752e−01	−2.8964729e−02	−2.1273060e−01	−7.9028308e−02
4.0086680e−01	−3.1982541e−02	−1.3203556e−01	−7.1957800e−02
6.6275187e−01	**−3.2682026e−02**	−1.4716440e−01	−2.6920571e−02
1.0006249e+00	−3.2620349e−02	−1.5461735e−01	-4.2887045e−02
1.3372481e+00	**−3.2682026e−02**	−1.4716440e−01	−2.6920571e−02
1.6001973e+00	−3.1973178e−02	−1.3117666e−01	−7.1897476e−02
1.8751443e+00	−2.7082594e−02	**−2.3021265e−01**	−1.1338662e−01
1.9034981e+00	−2.6190153e−02	−1.9274035e−01	**−1.3029716e−01**
1.9502844e+00	−2.2115611e−02	3.3780729e−02	−5.6126086e−02
1.9793694e+00	−1.2761813e−02	**5.3685435e−02**	1.0839406e−02
1.9909300e+00	−6.2864708e−03	4.1986175e−02	**4.0536339e−02**
1.9952203e+00	−3.4423357e−03	2.7217172e−02	3.2750426e−02
1.9998837e+00	−8.7124330e−05	8.1334795e−04	1.1464529e−03
2.0000000e+00	**0.0000000e+00**	0.0000000e+00	0.0000000e+00

The maximum/minimum values of each profile are presented in boldface.

Texas A&M University, we were able to compute the flow on non-uniform grids of very high resolution, consisting of up to 2×10^9 MAC nodes. We provide in this paper data for the flow at $Re = 1000$ and 5000 with guaranteed accuracy of 5×10^{-5} at $Re = 1000$ and 2.5×10^{-3} at $Re = 5000$.

REFERENCES

1. Guermond J-L, Minev PD. A new class of fractional step techniques for the incompressible Navier–Stokes equations using direction splitting. *Comptes Rendus Mathematique* 2010; **348**:581–585.
2. Botella O, Peyret R. Benchmark spectral results on the lid-driven cavity flow. *Computers & Fluids* 1998; **27**(4):421–433.
3. Bruneau C-H, Saad M. The 2D lid-driven cavity problem revisited. *Computers & Fluids* 2006; **35**:326–348.
4. Albensoeder S, Kuhlmann HC. Accurate three-dimensional lid-driven cavity flow. *Journal of Computational Physics* 2005; **206**:536–558.
5. Guermond JL, Shen J. Velocity-correction projection methods for incompressible flows. *SIAM Journal on Numerical Analysis* 2003; **41**(1):112–134.
6. Guermond JL, Shen J. On the error estimates for the rotational pressure-correction projection methods. *Mathethamatics of Computation* 2004; **73**(248):1719–1737 (electronic).
7. Douglas J, Jr. Alternating direction methods for three space variables. *Numerische Mathematik* 1962; **4**:41–63.
8. Guermond J-L, Minev PD. A new class of splitting methods for the incompressible Navier–Stokes equations using direction splitting. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**:2083–2093.

9. Chan A, Balaji P, Gropp W, Thakur R. Communication analysis of parallel 3D FFT for flat Cartesian meshes on large Blue Gene systems. In *Lecture Notes in Computer Science*, Vol. 5374, Badrinath R, Sadayappan P, Parashar M, Prasanna V (eds). Springer-Verlag: Berlin, 2008; 350–364.
10. Schulz R. 3D FFT with 2D decomposition. *CS Project Report*, 2008. (Available from: http://cmb.ornl.gov/Members/z8g/csproject-report.pdf) (accessed in October 2010).
11. Ghia U, Ghia KN, Shin CT. High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. *Journal of Computational Physics* 1982; **48**(3):387–411.
12. Guermond J-L, Migeon C, Pineau G, Quartapelle L. Start-up flows in a three-dimensional rectangular driven cavity of aspect ratio 1:1:2 at $Re = 1000$. *Journal of Fluid Mechanics* 2002; **450**:169–199.
13. Tang L, Cheng T, Tsang T. Transient solution for three-dimensional lid-driven cavity flows by a least-squares finite element method. *International Journal Numerical Methods in Fluids* 1995; **21**:413–432.
14. Shankar PN, Deshpande MD. Fluid mechanics in the driven cavity. *Annual Review of Fluid Mechanics* 2000; **32**:93–136.
15. Baragy E, Carey GF. Stream function-vorticity driven cavity solutions using p finite elements. *Computers & Fluids* 1997; **26**:453–468.
16. Erturk E, Corke TC, Gokcol C. Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. *International Journal for Numerical Methods in Fluids* 2005; **48**:747–774.
17. Ramanan N, Homsy GM. Linear stability of lid-driven cavity flow. *Physics of Fluids* 1994; **6**:2690–2701.
18. Auteri F, Parolini N, Quartapelle L. Numerical investigation on the stability of singular driven cavity flow. *Journal of Computational Physics* 2002; **183**:1–25.