On the stability and performance of the projection-3 method for the time integration of the Navier–Stokes equations

M. P. Kirkpatrick[1]   S. W. Armfield[2]

(Received 12 July 2007; revised 29 February 2008)

Abstract

The projection-2 method commonly used for the non-iterative time integration of the Navier–Stokes equations introduces a second order in time error. This error is reduced to third order when the projection-3 method is used. However, the projection-3 method can lead to solutions that are unbounded in time and, consequently, the projection-3 method has received little attention. This article compares the stability and performance of the projection-3 method with that of the projection-2 method. Both methods have been implemented in a Navier–Stokes solver that integrates the three dimensional equations on a staggered Cartesian grid. Time integration uses a second order hybrid Crank–Nicolson/Adams–Bashforth scheme. Results are presented for two test cases.

Contents

1 Introduction

Fractional step projection methods integrate the Navier–Stokes equations in time at each time step by first solving the momentum equations using an approximate pressure field to yield an intermediate velocity field that will not, in general, satisfy conservation of mass. A Poisson equation is then solved with the divergence of the intermediate velocity as a source term to provide a pressure or pressure correction, which is then used to correct the intermediate velocity field, providing a divergence free velocity field. The pressure is updated and integration then proceeds to the next time step. An extensive list of references for various fractional step methods developed for the Navier–Stokes equations was given by Armfield and Street [3].

The projection-1 (P1) method sets the pressure field to zero in the momentum equation and the Poisson equation is then solved for the new pressure, while the projection-2 (P2) method sets the pressure in the momentum equation to that obtained at the previous time step, and the Poisson equation is then solved for a pressure correction. Both the P1 method, with appropriate intermediate velocity boundary conditions, and the P2 method provide second order accuracy in time for the velocity and pressure fields, provided the

momentum equation is integrated using a second order accurate scheme [1].

The projection-3 (P3) method, first proposed by Gresho [4], uses a linear extrapolation from previous time steps to give a more accurate estimate of the pressure used in the momentum equation. This increases the order of the fractional step error—the additional error of the fractional step method with respect to an iterative scheme—from second to third order [3]. The advantage of the P3 method over an iterative method is that it is considerably faster since it only requires one iteration per time step. However, one of the main drawbacks of the P3 method concerns its stability. Shen [12] showed analytically that the P3 method can lead to solutions that are unbounded in time. Armfield and Street [3] tested the P3 method for a two dimensional natural convection flow in a square cavity and found that it remained stable for this case and reduced the size of the overall time stepping error by a factor of four. This article compares the P3 scheme with the P2 and iterative schemes for two cases. The first—two dimensional, isothermal, laminar flow in a driven cavity—is a relatively simple flow which we use primarily to validate the implementation of the schemes in the code. The second case is a typical 'real world' application—a large eddy simulation of a convective atmospheric boundary layer. Here we demonstrate that second order accuracy is maintained for all variables and assess the stability and performance of the P3 method for this more complex case. All simulations are performed using the PUFFIN code [5, 7, 8].

2   Driven cavity

The governing equations for isothermal laminar flow in a square driven cavity are the Navier–Stokes equations for incompressible flow, which are written in Cartesian tensor notation as

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{\text{Re}} \frac{\partial^2 u_i}{\partial x_j^2}, \tag{1}$$

$$\frac{\partial u_j}{\partial x_j} = 0.$$ (2)

Here x_i are the components of the position vector, t the time, u_i the components of the velocity vector, p the pressure, and Re the Reynolds number. The equations are normalized in terms of cavity width L and lid velocity U.

Numerical solution of the equations uses a finite volume method on a staggered Cartesian grid. All spatial derivatives are approximated using a second order central scheme. Time integration uses a hybrid implicit/explicit scheme in which Crank–Nicolson is used for the diffusion terms and a second order Adams–Bashforth scheme is used for the advection terms. This scheme is second order accurate in time without iteration [8].

In the P2 method, the momentum equations are first integrated using the pressure field from the previous time step, $p^{n-1/2}$, to obtain an intermediate velocity field $\tilde{u}_i^{n+1}$ at time $t = n+1$. In general, this velocity field will not be divergence-free. In order to obtain a divergence free velocity field, a Poisson equation is solved for a pressure correction p' in which the source term is the divergence of the intermediate velocity field,

$$\frac{\partial^2 p'}{\partial x_j^2} = \frac{1}{\Delta t}\frac{\partial \tilde{u}_j^{n+1}}{\partial x_j}.$$ (3)

The solution of this equation is then used to correct the pressure and velocity,

$$p^{n+1/2} = p^{n-1/2} + p',$$ (4)

$$u_i^{n+1} = \tilde{u}_i^{n+1} - \Delta t\frac{\partial p'}{\partial x_i}.$$ (5)

The velocity correction step constitutes a projection of the intermediate velocity field onto a subspace of divergence free velocity fields. For this scheme the projection error can be shown to be second order in time [1].

The P3 projection method requires only a very simple modification to the above scheme. Instead of using the pressure $p^{n-1/2}$ in the momentum

equations, an estimate of the pressure at time $t = n + 1/2$ is obtained by linear extrapolation,

$$\tilde{p}^{n+1/2} = 2p^{n-1/2} - p^{n-3/2} \,. \tag{6}$$

The pressure used in the momentum equation is now approximated to second order, resulting in a projection error that is third order in time [3].

A third scheme, which removes the projection error altogether, involves iterating a number of times through the momentum and pressure correction steps at each time step. We refer to this as the iterative scheme. For the cases presented here, the results remained essentially unchanged when more than three iterations were used and so three iterations were used for all cases.

The Adams–Bashforth scheme used for the integration of the advection terms in the momentum equations requires information from $t = n$ and $t = n - 1$ in order to perform the extrapolation to $t = n + 1/2$. Similarly the P3 method requires information from $t = n - 1/2$ and $t = n - 3/2$. For this reason, the first two time steps are performed using Crank–Nicolson for all terms in the momentum equations and the iterative scheme for pressure.

The matrix equations are solved using a preconditioned BiConjugate Gradient Stabilized method. For the momentum equations we use a Jacobi preconditioner, while for the pressure correction equation we use a preconditioner based on the Modified Strongly Implicit (MSI) method. For the momentum equations the solver iterates until the L2 norm of the residual over the domain is less than a preset maximum conservation error. For the pressure correction equation, the solver iterates until the residual is reduced by an order of magnitude. The divergence of the velocity field is then checked. If the divergence error is greater than the maximum allowed conservation error the pressure correction procedure is repeated. This continues until the conservation error criterion has been met. For this case the maximum conservation error was set to 10^{-10} for all variables. We found that it was necessary to make this value at least two orders of magnitude smaller than the time stepping errors to be measured (typically of order 10^{-8} for this case) to ensure that conservation errors did not contaminate the results.

The driven cavity simulation uses a uniform grid with 100×100 cells. The Reynolds number is set to $\text{Re} = 200$ which gives a maximum cell Reynolds number of approximately two. Dirichlet boundary conditions are used for the momentum equations with $u_b = v_b = 0$ at the side and bottom walls and $u_b = U$ and $v_b = 0$ at the lid of the cavity. Neumann boundary conditions are used for the pressure correction equation at all walls, with the gradient in the direction normal to the boundary set to zero. Initial conditions are $u(0) = v(0) = p(0) = 0$. The equations are integrated from $t = 0$ to $t = 1$. A supplementary file shows the flow field in the cavity at $t = 1$.

Errors were calculated at $t = 1$ for solutions obtained with CFL numbers of 0.4, 0.2, 0.1 and 0.05, where $\text{CFL} = \Delta t U / \Delta x$ with Δt the time step and Δx the width of a grid cell. Because the pressure solution is staggered in time with respect to the velocity solution, the pressure at $t = 1$ must be obtained by extrapolation, to avoid an apparent first order error for the pressure solution [2]. Errors were quantified as the L2 norm of the difference between the test solution and a benchmark solution obtained using the iterative scheme and a CFL number of 0.01. The time step used for the benchmark solution is more than four times smaller than the smallest time step used for the test cases. Since all the time stepping schemes used are second order, the error in the benchmark solution will be at least an order of magnitude smaller than that of the test solutions and can therefore be considered negligible.

Figure 1 shows the u and p errors for the P2, P3 and iterative schemes. The results for v were similar to those for u. All schemes display quadratic convergence with reduction in time step indicating that they are second order accurate in time. The errors for the P3 and iterative schemes are very similar. Since the projection error is negligibly small for the iterative scheme, this indicates that the P3 method reduces the projection error to a negligible value for this case. With the P2 scheme, the u error is approximately twice that obtained with the P3 and iterative schemes. The difference is less significant (approximately 30%) for p and the difference for v was similar to p. The improvement in accuracy achieved with the P3 method compared to the
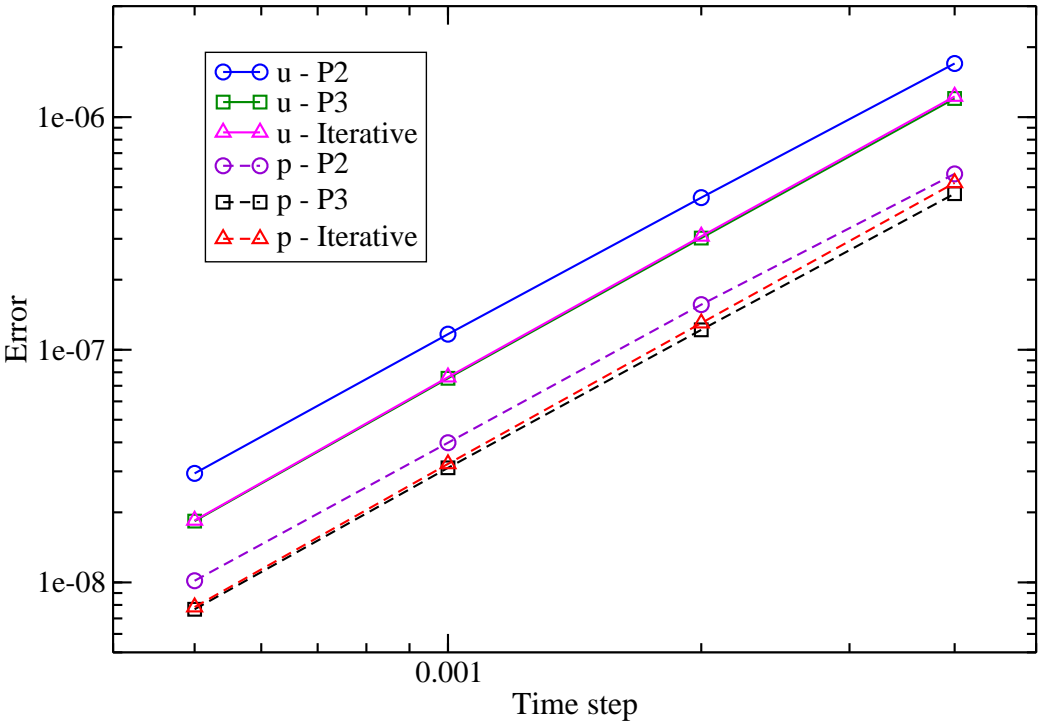
FIGURE 1: Time accuracy for the driven cavity case.

P2 method is not as marked as for the natural convection cavity tested by Armfield and Street [3] where a four-fold reduction in error was measured. Nevertheless, a two-fold reduction is significant.

3   Atmospheric boundary layer

A typical 'real world' application in which time accuracy is important is in simulations of the atmospheric boundary layer. In this section we assess the stability and performance of the P3 method for a simulation of a convective atmospheric boundary layer. The governing equations are the Boussinesq equations for a shallow atmospheric boundary layer,

$$\frac{\partial u_i}{\partial t} + \frac{\partial(u_i u_j)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \delta_{i3}g\frac{(\theta - \langle\theta\rangle)}{\Theta_{\text{ref}}} - \epsilon_{ijk}f_j u_k - \frac{\partial \tau_{ij}}{\partial x_j}, \quad (7)$$

$$\frac{\partial \theta_1}{\partial t} + \frac{\partial(\theta_1 u_j)}{\partial x_j} + w\frac{\partial \theta_0}{\partial z} = -\frac{\partial \gamma_\theta}{\partial x_j}, \quad (8)$$

$$\frac{\partial u_j}{\partial x_j} = 0. \quad (9)$$

Here δ_{ij} is the Kronecker delta, ϵ_{ijk} the permutation tensor, f_j the Coriolis parameter, g the acceleration due to gravity and angled brackets represent averaging on horizontal planes. The pressure p is the non-hydrostatic component of the pressure normalized by the reference density ϱ_{ref}. The virtual potential temperature θ is decomposed as

$$\theta(x, y, z, t) = \theta_0(z) + \theta_1(x, y, z, t), \quad (10)$$

where subscript 0 refers to the initial base state profile and subscript 1 to dynamic perturbations about the base state. Θ_{ref} is a reference virtual potential temperature and is set equal to the virtual temperature at the reference level, namely the ground. τ_{ij} and γ_θ are the turbulent fluxes of momentum and virtual potential temperature respectively.

A large eddy simulation approach is used to account for the fact that there are scales of motion that cannot be resolved on the grid. As such, the dependent variables, u_i, p and θ, in equations (7–9) are considered to represent spatially filtered quantities. The anisotropic component of the turbulent momentum flux is parameterized using the Smagorinsky–Lilly model [10, 13],

$$\tau_{ij} - \frac{1}{3}\delta_{ij}\tau_{kk} = -2K_m S_{ij}\,, \tag{11}$$

where S_{ij} is the strain rate,

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{1}{3}\delta_{ij}\frac{\partial u_k}{\partial x_k}\,, \tag{12}$$

and the eddy viscosity

$$K_m = (C_s\Delta)^2\,|S|\,\sqrt{1 - \mathrm{Ri}/\mathrm{Pr}_t}\,. \tag{13}$$

Here $|S| = \sqrt{2S_{ij}S_{ij}}$, C_s is a dimensionless model coefficient, Δ the width of the spatial filter that is applied to the equations, and Pr_t the subgrid-scale turbulent Prandtl number. The gradient Richardson number $\mathrm{Ri} = N^2/|S|^2$ with the buoyancy frequency N given by $N^2 = -(g/\varrho)\partial\varrho/\partial z$. A damping function [11] is used to account for the decreasing eddy size observed as the surface is approached. A modified length scale is calculated from

$$\frac{1}{(l^*)^2} = \frac{1}{(C_s\Delta)^2} + \frac{1}{(\kappa z)^2}\,, \tag{14}$$

where κ is the von Karman constant with a value of 0.41, and z the distance from the surface. The eddy viscosity is then computed as

$$K_m = (l^*)^2|S|\,. \tag{15}$$

The turbulent flux of virtual potential temperature is modelled as

$$\gamma_\theta = -\frac{K_m}{\mathrm{Pr}_t}\frac{\partial\theta}{\partial x_j}\,. \tag{16}$$

Kirkpatrick et al. [6] further discussed turbulence modelling in the atmospheric boundary layer.

The numerical methods used to solve these equations are the same as those described in Section 2. The treatment of the potential temperature equation is the same as that of the momentum equation, with the exception of the advection terms, where a QUICK scheme with ULTRA flux limiter [9] is used to ensure that the solution remains monotonic.

The test cases use a $60 \times 60 \times 64$ cell grid on a domain of size of $6\,\text{km} \times 6\,\text{km} \times 2.2\,\text{km}$. The grid is stretched in the vertical direction to give $20\,\text{m}$ vertical resolution at the ground and the capping inversion. Resolution in the horizontal directions is $100\,\text{m}$. This is typical of the resolution currently used for large eddy simulations of the atmospheric boundary layer. The base state is a neutrally stratified boundary layer with $\theta = 295\,\text{K}$ up to a height of $1500\,\text{m}$. The boundary layer is capped by a $6.5\,\text{K}$ temperature inversion at $1500\,\text{m}$, above which the air is stably stratified with $\partial\theta/\partial z = 2\,\text{K/km}$. The initial velocity above the boundary layer is equal to the geostrophic winds, which are set to $U_\text{geo} = 10\,\text{m/s}$ and $V_\text{geo} = 0\,\text{m/s}$. Within the boundary layer the velocity profile follows a 1/7th power law. The initial large scale pressure gradient is set such that it balances the Coriolis acceleration associated with the geostrophic winds. The Coriolis parameter is set to $f = 10^{-4}\,\text{s}^{-1}$. Periodic boundary conditions are used at the lateral boundaries, while the top boundary is impermeable, and adiabatic with zero shear stress. At the ground the mean friction velocity is set to $u_* = \langle\tau_w\rangle^{1/2} = 0.6\,\text{m/s}$, and the mean temperature flux to $\langle\gamma_w\rangle = 0.24\,\text{K\,m/s}$. The coefficient in the Smagorinsky–Lilly model is set to $C_s = 0.18$ and the turbulent Prandtl number to $\text{Pr}_t = 0.4$. These are typical values for an unstably stratified boundary layer.

The atmospheric simulations were run on a multiprocessor Tyan VX50–B4881 computer. The computer has 8 dual core AMD Opteron 865 chips (16 CPUs) with $2\,\text{MB}$ L2 cache and $8\,\text{GB}$ of DDR400 PC3200 RAM. The code is parallelized using Message Passing Interface (MPI) and linked to the public domain MPICH2 libraries. Parallelization is achieved through domain

decomposition with the domain divided into horizontal slices. The computations for each slice are performed by a different processor. The BiConjugate Gradient solver parallelizes in a straightforward manner, as does the Jacobi preconditioner. However, the MSI preconditioner does not. Instead, for the parallel runs, we use a 'zebra' Gauss–Seidel preconditioner with successive overrelaxation and Chebyshev acceleration. By 'zebra' we mean that an odd–even alternation strategy is used in the vertical direction, which allows parallelization with minimal interprocessor communication when horizontal slices are used for the domain decomposition.

The equations were first integrated from $t = 0$ to $t = 1800$ s using the P3 scheme with a maximum CFL number of 0.4 to allow the turbulent boundary layer to develop. Accuracy tests were then performed over a period of 180 s and errors quantified in the same manner as described in Section 2. For this case, the maximum conservation errors were set to 10^{-6}, as the smallest time stepping error was of order 10^{-4}. The benchmark simulation was run with $\Delta t = 0.05$ s, which corresponds to a maximum CFL number in the domain of 0.015. Test simulations were then run for $\Delta t = 0.25, 0.5, 1.0, 2.0$ s, corresponding to $\text{CFL}_{\max} = 0.075, 0.15, 0.3, 0.6$.

Simulations were run on eight of the 16 available CPUs and required ~ 0.8 s wall clock time per time step. With $\Delta t \sim 1.3$ for CFL $= 0.4$, this corresponds to approximately 40 minutes computation time per hour of simulation time. The parallel efficiency of the code, that is the ratio of the speed-up to the number of processors, is approximately 82% for this case. A supplementary file contains an animation showing the evolution of the flow during the first 30 minutes of the simulation.

Figure 2 shows the u, v, w, θ and p errors for the P3 scheme. The errors for all dependent variables display quadratic convergence with reduction in time step indicating that second order accuracy was achieved for all variables for this complicated test case. The errors for the P2 and iterative schemes were very similar to those of the P3 scheme and are not shown. This indicates that the projection error is negligible for this more complex case. Nevertheless,
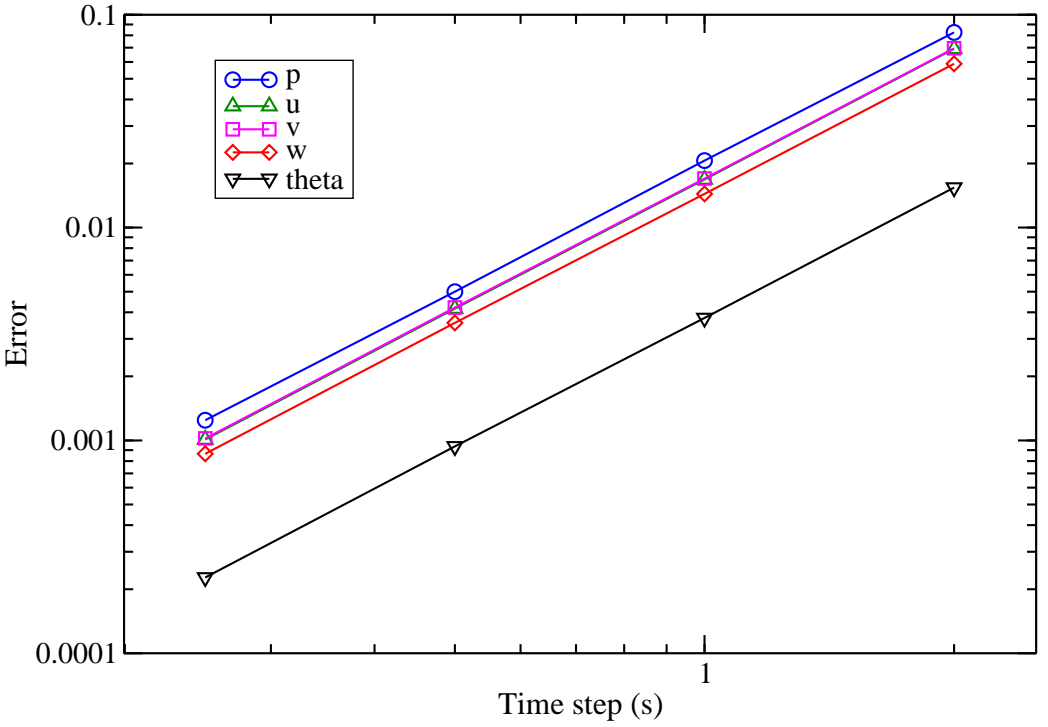
FIGURE 2: Time accuracy of the P3 method for the atmospheric boundary layer case.

the P3 method remained stable for the 30 min 'spinup' of this complex case, and remained stable when the simulation was run to $t = 2$ h.

4   Conclusions

The results for the driven cavity test case show that the P3 method significantly reduces the size of the projection error, as expected since the order of the projection error is increased from second to third order with this scheme. This led to a two-fold reduction in the overall time stepping error for u and a lesser reduction for v and p for this case. The reduction in error obtained using the P3 method depends on the size of the projection error relative to the other time integration errors and this will vary from case to case. For the natural convection cavity flow tested by Armfield and Street [3] the error was reduced by four-fold. For the atmospheric boundary layer case, the projection error was negligible compared to the other errors, and the P2, P3 and iterative methods all gave very similar results. However, even in this case the P3 method still has the advantage of lower run time. By using a more accurate approximation of the pressure in the momentum equations, the number of sweeps of the solver needed to reduce the conservation errors to the required level is reduced. For the atmospheric boundary layer case, the number of sweeps of the matrix equation solver per time step was reduced by 35–40% on average, which reduced run time by approximately 30%.

The time accuracy of the code has been verified, with the accuracy of all variables, including pressure, shown to be second order for both the simple driven cavity case and the complex atmospheric boundary layer case. This is noteworthy considering the complexity of the equations, turbulence models and boundary conditions used for the atmospheric boundary layer case.

The P3 method has been shown to be stable for a complex, 'real world' simulation. Tests of other similar cases have also shown it to be stable. The only exceptions we have found to date involved the use of a Neumann-

type boundary condition such as an 'outlet' for the momentum equation. With periodic, 'specified flux', or Dirichlet type boundary conditions, global conservation of mass is automatically ensured as long as the boundary values have been set appropriately. This is not true with a Neumann boundary condition and consequently the mass flux across the outlet boundary must be corrected as part of the pressure correction step in order to enforce global mass conservation. We suggest that this correction may generate time step scale oscillations in the pressure field close to the outlet, which are then amplified by the extrapolation procedure used in the P3 method, causing the simulation to become unstable.

The question as to why the P3 method appears to be stable for most cases, when it has been shown analytically to be unstable, is the subject of ongoing research by the authors. The stability analysis of Shen [12] does not take into account the effect of boundary conditions. It may be an appropriate choice of boundary conditions stabilizes the method.

References

[1] S. W. Armfield and R. Street. An analysis and comparison of the time accuracy of fractional-step methods for the Navier–Stokes equations on staggered grids. *Int. J. Numer. Meth. Fluids*, 38:255–282, 2002. doi:10.1002/fld.217 C561, C562

[2] S. W. Armfield and R. Street. The pressure accuracy of fractional-step methods for the Navier–Stokes equations on staggered grids. *ANZIAM J.*, 44:C20–C39, 2003. http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/670 C564

[3] S. W. Armfield and R. Street. Modified fractional-step methods for the Navier–Stokes equations. *ANZIAM J.*, 45:C364–C377, 2004.

http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/894 C560, C561, C563, C566, C571

[4] P. Gresho. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via finite element method that also introduces a nearly consistent mass matrix. Part 1: Theory. *Int. J. Numer. Meth. Fluids*, pages 587–620, 1990. doi:10.1002/fld.1650110509 C561

[5] M. P. Kirkpatrick. *A large eddy simulation code for industrial and environmental flows.* PhD thesis, The University of Sydney, 2002. C561

[6] M. P. Kirkpatrick, A. S. Ackerman, D. E. Stevens, and N. N. Mansour. On the application of the dynamic smagorinsky model to large eddy simulations of the cloud-topped atmospheric boundary layer. *J. Atmos. Sci.*, 63(2):526–546, 2006. doi:10.1175/JAS3651.1 C568

[7] M. P. Kirkpatrick, S. W. Armfield, and J. H. Kent. A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid. *J. Comp. Phys.*, 184:1–36, 2003. doi:10.1016/S0021-9991(02)00013-X C561

[8] M. P. Kirkpatrick, S. W. Armfield, J. H. Kent, and T. Dixon. Simulation of vortex-shedding flows using high-order fractional step methods. *J. ANZIAM*, 42(E):856–876, 2000. http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/625 C561, C562

[9] B. P. Leonard and S. Mokhtari. Beyond first-order upwinding: The ULTRA-SHARP alternative for non-oscillatory steady-simulation of convection. *Int. J. Num. Meth. Eng.*, 30:729–766, 1990. doi:10.1002/nme.1620300412 C568

[10] D. K. Lilly. On the numerical simulation of buoyant convection. *Tellus*, 14(2):148 – 172, 1962. C567

[11] P. J. Mason and D. J. Thomson. Stochastic backscatter in large-eddy simulations of boundary-layers. *J. Fluid Mech.*, 242:51–78, 1992. doi:10.1017/S0022112092002271 C567

[12] J. Shen. A remark on the projection-3 method. *Int. J. Numer. Meth. Fluids*, 16:249–253, 1993. doi:10.1002/fld.1650160308 C561, C572

[13] J. Smagorinsky. General circulation experiments with the primitive equations: I. The basic experiment. *Mon. Weath. Rev.*, 91:99–164, 1963. C567

Author addresses

1. **M. P. Kirkpatrick**, School of Aerospace, Mechanical & Mechatronic Engineering, University of Sydney, Sydney, Australia.
   mailto:michael.kirkpatrick@aeromech.usyd.edu.au

2. **S. W. Armfield**, School of Aerospace, Mechanical & Mechatronic Engineering, University of Sydney, Sydney, Australia.