INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 26, 459–483 (1998)

COMPARISON OF SIMPLE- AND PISO-TYPE ALGORITHMS FOR TRANSIENT FLOWS

I. E. BARTON

Division A, CFD Lab., Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, U.K.

SUMMARY

Various pressure-based schemes are proposed for transient flows based on well-established SIMPLE and PISO algorithms. The schemes are applied to the solution of unsteady laminar flow around a square cylinder and steady laminar flow over a backward-facing step. The implicit treatment and the performance of the various schemes are evaluated by using benchmark solutions with a small time step. Three different second-order-accurate time derivatives based on different time levels are presented. The different time derivatives are applied to the various schemes under consideration. Overall the PISO scheme was found to predict accurate results and was robust. However, for small time step values, alternative schemes can predict accurate results for approximately half the computational cost. The choice of time derivative proved to be very significant in terms of the accuracy and robustness of a scheme. Significantly, the one-sided forward differencing scheme was the most successful used in conjunction with a strongly implicit-based algorithm. However, a greater degree of accuracy was achieved using the standard PISO algorithm with the Crank–Nicolson time derivative. Recommendations for future work are discussed. © 1998 John Wiley & Sons, Ltd.

Int. J. Numer. Meth. Fluids, **26**: 459–483 (1998).

KEY WORDS: SIMPLE algorithm; PISO algorithm; unsteady incompressible flow

1. INTRODUCTION

Incompressible flows are dominated by the effects of pressure variations where their effects on the velocity field are significant. In this case the continuity equation does not act as a conservation of mass equation as it does for compressible flows but as an important constraint on the behaviour of the velocity field. How the pressure is derived from the continuity and momentum equations is not straightforward using primitive variables. One approach is to derive a pressure equation from the continuity equation. This has been done in a number of ways. Harlow and Welch[1] derived one of the earliest primitive variable methods with the marker and cell (MAC) scheme. Other primitive variable methods include a steady state solution procedure called the semi-implicit method for pressure-linked equations (SIMPLE) scheme.[2–6] Issa and coworkers[7,8] developed an improved pressure–velocity calculation method called the pressure-implicit with splitting of operators (PISO) scheme, initially aimed at solving unsteady flows. However, a simplified version can be interpreted as an extension of the SIMPLE method and is usually superior for solving steady state problems. Wanik and Schnell[9] compared the SIMPLE and PISO schemes for turbulent flow problems following the standard

* Correspondence to: I. E. Barton, Division A, Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, U.K.

CCC 0271–2091/98/040459–25 $17.50

Received February 1997
Revised April 1997

formulations for both schemes. The focus of this work was how the turbulent source terms should be treated and overall the PISO scheme was found to be a more successful scheme than the SIMPLE scheme. Kim and Benson[10] predicted that the 'simplified' MAC scheme (SMAC) was probably more efficient than PISO for unsteady problems despite the difficulties in making the temporal derivatives second-order-accurate. Kim and Benson also concluded that the PISO scheme is a more efficient method to solve steady state problems. Cheng and Armfield[11] undertook a similar investigation comparing SIMPLE-consistent (SIMPLEC), PISO and SMAC; they showed that in terms of suppressing pressure oscillations the SMAC method is more efficient than the SIMPLEC and PISO methods. There have been several SIMPLE-like schemes proposed for steady state flows for various applications; however, the performance of the SIMPLE scheme for steady state flows is strongly dependent on the underrelaxation treatment, therefore alternative steady state schemes have received little interest.[12–14] All the schemes described are usually called a 'pressure Poisson' or 'pressure-based' method (in some cases they are known as a 'pressure correction' method), because the pressure field is a dependent variable which invariably forms a Poisson-type problem throughout the solution domain. Pressure-based schemes are reviewed in detail elsewhere.[6,15–18]

The basis of the present research is to use arguably one of the most popular pressure-based methods, the SIMPLE scheme, and its closely related counterpart the PISO scheme. The present study essentially follows the formulations of the original SIMPLE versions.[3–6] Additionally, the present study also follows the formulation of the original PISO scheme,[7] which as previously stated is closely related to the SIMPLE methodology. The SIMPLE scheme is altered to predict time-dependent flows; this has previously been presented.[19] Ströll *et al.*[20] have also considered unsteady piston–cylinder flows using a SIMPLE algorithm.

The aim of the present research is to investigate variants of the SIMPLE and PISO schemes with the objective of obtaining an efficient unsteady state solver. After discussing the governing equations and the discretization methodology, variants on the SIMPLE and PISO schemes are presented in Section 3. The current research is also concerned with applying various time derivative terms; the time derivative terms under consideration are presented in Section 4. As the main focus of the research is on deriving an efficient time-dependent solution scheme, the majority of the simulations undertaken are for laminar flow around a square cylinder.

2. GOVERNING EQUATIONS AND DISCRETIZATION

2.1. Governing equations

The governing equations under consideration for the present investigation of planar constant-density laminar flow are the equation of continuity

$$\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0 \tag{1}$$

and the conservation of momentum equations

$$\frac{\partial \rho u}{\partial t} + \frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho uv}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\mu\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y}\mu\frac{\partial u}{\partial y}, \tag{2}$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial \rho uv}{\partial x} + \frac{\partial \rho v^2}{\partial y} = -\frac{\partial p}{\partial y} + \frac{\partial}{\partial x}\mu\frac{\partial v}{\partial x} + \frac{\partial}{\partial y}\mu\frac{\partial v}{\partial y}, \tag{3}$$

where ρ, μ, u and v are the fluid density, viscosity and velocity components.

2.2. Discretization

The governing equations are presented in primitive variable form. The next stage is to apply conventional differencing to equations (1)–(3). This process is discussed in detail elsewhere[16] and requires no further elucidation here. For the sake of generality the discretized equations are presented in coefficient form, following the terminology used in Reference 5. This means that our following arguments can be applied to various finite difference or finite volume discretization schemes.

The governing equations (1)–(3) may be expressed in discretized form as

$$\Delta_x(\rho u)^{n+1} + \Delta_y(\rho v)^{n+1} = 0, \tag{4}$$

$$A^n_{\mathrm{P},u} u^{n+1}_{\mathrm{P}} = \sum A^n_{M,u} u^{n+1}_M - \Delta_x p^{n+1} + S^{n+1}_u + S^n_u, \tag{5}$$

$$A^n_{\mathrm{P},v} v^{n+1}_{\mathrm{P}} = \sum A^n_{M,v} v^{n+1}_M - \Delta_y p^{n+1} + S^{n+1}_v + S^n_v. \tag{6}$$

In equations (5) and (6) the time derivative term has been incorporated into the A^n_{P} and source terms. For the case of incompressible laminar flow the source terms S^n_u and S^n_v only contain temporal terms and the source terms S^{n+1}_u and S^{n+1}_v are zero. For more complex flows such as compressible flows or modelled turbulent flows there will be S^{n+1}_u and S^{n+1}_v source terms, which we have included above for the sake of generality. The physical source terms can be dealt with in a number of ways.[9] The A coefficient terms also contain convective and diffusive contributions. The index M is a grid identifier referring to all the nodes surrounding the pole node that are involved in the formulation of the finite difference representation of the spatial fluxes. The subscript P denotes the 'pole' node or cell. In the above discretization, similar to Patankar[5] and Issa,[7] linear implicit treatment of the terms is applied, which means that terms such as u^2 are split to form $u^n u^{n+1}$.

The above equations are general in form; for example, similar equations have been used in the presentation of various numerical algorithms.[5,7] The scheme's methodology such as the discretization procedures are described elsewhere[21] and therefore do not require further explanation here. The example calculations employ a staggered Cartesian grid system[1] and the example calculations presented later use a second-order upwind differencing scheme for the majority of the calculations.[22] Similar methodologies have been used in previous studies.[23–25]

3. NUMERICAL ALGORITHMS

The fundamental concept of the SIMPLE and PISO schemes is to derive a pressure correction equation by enforcing mass continuity over each cell. The way this is achieved is as follows. The pole velocity terms in the discretized momentum equations are substituted into the equation of continuity, which leads to the problem that the velocity and pressure fields require simultaneous solution. As this is difficult to solve, the surrounding velocity terms around the pole node are assumed to remain constant. Alternatively, we could say that the 'velocity corrections' surrounding the pole node are zero. This terminology is used by Patankar.[5] The assumption that the velocity corrections adjacent to the pole cell are negligible is a major premise of the SIMPLE algorithm. This premise has been slightly modified in closely related schemes such as SIMPLE-revised,[26] whereas the PISO scheme introduces velocity corrections in the succeeding 'correction stage'. The treatment of the velocity correction terms by the SIMPLE scheme is acceptable when solving steady state flows, as we require only a temporary linkage between the pressure and velocity fields. As the linkage is repetitively applied, the pressure and velocity fields should ultimately satisfy both the momentum and continuity equations. Thus any approximations made in the solution of the algebraic equations or the solution procedure itself are justified, provided that the approximations reduce with convergence. We can also argue that this approximation is also acceptable if small time steps are used for unsteady calculations.

The approximation of ignoring surrounding velocity corrections allows velocity corrections to be directly related to pressure correction terms, which leads to a simpler solution procedure. This is an advantage of the SIMPLE scheme in contrast with the PISO scheme. The mass continuity is enforced for each cell by applying a velocity–pressure correction relationship to obtain a pressure correction equation; after this equation is solved, the velocity–pressure corection relationship is used to 'correct' the velocity prediction, ensuring that continuity is satisfied. The SIMPLE scheme is considered to be a point-by-point method.[3] However, if the solution procedure is applied across the entire domain, as is the case for the PISO scheme, then the SIMPLE scheme can be described as a 'disguised pressure Poisson' method.[16] Such an approach is necessary when carrying out time-dependent calculations.

The SIMPLE methodology has been introduced in some detail,[3,5] while the PISO scheme has been discussed in Reference 7; flow charts of the methodology are given in Reference 27. Below, attention is focused on various alternatives to the standard SIMPLE and PISO schemes. The momentum equations are represented by a single equation where U can be replaced by either u or v. There are 11 schemes presented in total, including the SIMPLE and PISO schemes. The alternative algorithms are aimed at producing a more implicit scheme where the A coefficients are not based on the nth time level but on the $n+1$ level; this is explained later.

Scheme 1. SIMPLE (for transient flows)

The standard SIMPLE scheme[5] is not formulated to solve transient flows, although a discretized time derivative term is introduced. This term is used in order to underrelax the scheme; this approach essentially amounts to local time stepping.[4] How this relates to directly underrelaxing terms is discussed in Reference 21. However, in the present investigation, real time derivatives are applied.

Step 1, predict velocities based on the momentum equations:

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{*} = \sum A_{M}^{n} U_{M}^{*} - \Delta p^{n} + S_{u}. \tag{7}$$

Step 2, predict pressure and velocities in order to satisfy continuity using equation (4) and the equation

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{**} = \sum A_{M}^{n} U_{M}^{*} - \Delta p^{*} + S_{u}. \tag{8}$$

As previously described, the second step is achieved by substituting the u_{P}^{**} and v_{P}^{**} terms into the discretized equation for continuity, equation (4), and thereby obtaining a Poisson pressure equation with source terms based on the velocity terms u^* and v^*. After this is solved, the solution can then be substituted back into equation (8) in order to obtain the terms u^{**} and v^{**}. It is probably most efficient to actually derive a Poisson pressure *correction* equation by taking step 2 away from step 1, which gives

$$U_{\mathrm{P}}^{**} = U_{\mathrm{P}}^{*} - \Delta(p^{*} - p^{n})/A_{\mathrm{P}}^{n}, \tag{9}$$

and so it is this expression that is substituted into the discretized continuity equation. For more details on deriving the velocity–pressure correction equation, see Reference 5 or 21.

The SIMPLE scheme has been enhanced in a number of ways by others; some recent enhancements include a non-staggered formulation using a finite difference approach.[28] Multigrid techniques have been incorporated[29] as well as a multigrid solver.[30] The original formulations of the SIMPLE scheme used Cartesian grid systems; however, body-fitted co-ordinates are now very popular[31,31] and the finite element method has been used with the SIMPLE scheme.[33] Enhancements made to the SIMPLE-type methodologies have been reviewed.[18,34,35]

Note that we have used a slightly different argument from Patankar[5] for equations (7) and (8). In Reference 5, equation (8) uses the term $A_M U_M^{**}$ instead of $A_M U_M^{*}$; however, Patankar then applies the

approximation that $A_M U_M^{**} - A_M U_M^{*} = 0$. The advantage with the current presentation is that no hidden approximations are applied.

Scheme 2. SIMPLE (doubled)

This attempts to improve the SIMPLE algorithm by updating the coefficients and reapplying the scheme.

Step 1, predict velocities based on the momentum equations:

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{*} = \sum A_M^{n} U_M^{*} - \Delta p^{n} + S_u. \tag{10}$$

Step 2, predict pressure and velocities in order to satisfy continuity using equation (4) and the equation

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{**} = \sum A_M^{n} U_M^{*} - \Delta p^{*} + S_u. \tag{11}$$

Next update the coefficient terms so that $A_{\mathrm{P}}^{n} \rightarrow A_{\mathrm{p}}^{**}$ and then repeat steps 1 and 2, accounting for steps 3 and 4.

Note that in previous studies using the SIMPLE and PISO schemes the A coefficient does not have a superscript, as it is implicitly known that the A term is represented by the nth level of the velocity terms. In the present study the 'n-level' superscript of the A coefficients is explicitly stated, referring to which velocity terms are being used for their derivation.

Scheme 3. SIMPLE + additional continuity level

This scheme is an extension of the SIMPLE algorithm with an additional correction level that satisfies continuity, so the additional step is as follows.

Step 3, correct velocities and pressure such that continuity is satisfied using equation (4) and the equation

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{***} = \sum A_M^{n} U_M^{*} - \Delta p^{**} + S_u. \tag{12}$$

This scheme leads on to the PISO scheme where the U_M^{*} term is replaced by a U_M^{**} term, thus introducing non-zero velocity corrections that surround the pole node.

Scheme 4. PISO

Scheme 4 is the standard PISO scheme.[7,8] The similarities and differences of the PISO algorithm compared with the SIMPLE algorithm are discussed in Reference 7.

Step 1, predict velocities based on the momentum equations:

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{*} = \sum A_M^{n} U_M^{*} - \Delta p^{n} + S_u. \tag{13}$$

Step 2, predict pressure and velocities in order to satisfy continuity using equation (4) and the equation

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{**} = \sum A_M^{n} U_M^{*} - p^{*} + S_u. \tag{14}$$

Step 3, correct pressure and velocities such that continuity is satisfied using equation (4) and the equation

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{***} = \sum A_M^{n} U_M^{**} - \Delta p^{**} + S_u. \tag{15}$$

From the expressions above we can see how similar the PISO and SIMPLE algorithms are, although in contrast with the SIMPLE scheme the PISO scheme has an additional corrector level that satisfies continuity. The PISO scheme is not fully implicit, as the velocity correction terms are

explicitly introduced, i.e. the pole velocity term is a correction order higher than the surrounding nodes. This means that the PISO scheme has the extra complexity that the velocity correction terms need to be calculated and stored.

The PISO scheme, like the SIMPLE scheme, has been developed mainly for combusting flows.[8,27,36] This includes combusting particle flows.[37] The multigrid method has been applied to PISO, again for a combustion chamber application.[38] Dukowicz[39] presented an example of particulate time-dependent flow with a Monte Carlo simulation for particle dispersion.

Scheme 5. PISO + additional continuity level

This scheme is the same as Scheme 4 with an additional correction level for pressure and velocities which satisfies continuity. The additional step is as follows.

Step 4, correct pressure and velocities such that continuity is satisfied using equation (4) and the equation

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{****} = \sum A_{M}^{n} U_{M}^{***} - \Delta p^{***} + S_u. \tag{16}$$

Scheme 6. PISO + additional momentum level

This scheme is the same as Scheme 4 with an additional correction level for velocities derived from the momentum equations. The coefficients terms are updated from the previous step. The current pressure field is used. Thus the additional level is as follows.

Step 4, correct velocities based on the momentum equations:

$$A_{\mathrm{P}}^{***} U_{\mathrm{P}}^{****} = \sum A_{M}^{***} U_{M}^{****} - \Delta p^{**} + S_u. \tag{17}$$

Scheme 7. PISO + SIMPLE

This scheme first applies PISO, then updates the coefficient terms so that $A_{\mathrm{P}}^{n} \rightarrow A_{\mathrm{P}}^{**}$ and then applies the SIMPLE algorithm. Obviously, the most up-to-date pressure gradient is used, so that $\Delta p^{n} \rightarrow \Delta p^{**}$ is used for step 4. The scheme therefore has two additional steps which take the following form.

Step 4, correct the velocities based on the momentum equations:

$$A_{\mathrm{P}}^{***} U_{\mathrm{P}}^{****} = \sum A_{M}^{***} U_{M}^{****} - \Delta p^{**} + S_u. \tag{18}$$

Step 5, correct pressure and velocities such that continuity is satisfied using equation (4) and the equation

$$A_{\mathrm{P}}^{***} U_{\mathrm{P}}^{*****} = \sum A_{M}^{***} U_{M}^{****} - \Delta p^{***} + S_u. \tag{19}$$

Scheme 8. PISO (doubled)

This scheme is similar to Scheme 2 in that the standard algorithm is repeated twice, but in this case for the PISO scheme. The A coefficients are updated so that $A_{\mathrm{P}}^{n} \rightarrow A_{\mathrm{P}}^{***}$ after the first loop, as well as the pressure field.

Scheme 9

This scheme has similarities to the SIMPLE scheme in that there are no velocity correction terms. The only difference is that the coefficient terms are updated after the first step and the velocities are then recalculated.

Step 1, predict velocities using the momentum equations:

$$A_{\mathrm{P}}^{n} U_{\mathrm{P}}^{*} = \sum A_{M}^{n} U_{M}^{*} - \Delta p^{n} + S_{u}. \tag{20}$$

Step 2, predict velocities using the momentum equations and updated coefficients:

$$A_{\mathrm{P}}^{*} U_{\mathrm{P}}^{**} = \sum A_{M}^{*} U_{M}^{**} - \Delta p^{n} + S_{u}. \tag{21}$$

Step 3, correct pressure and velocities in order to satisfy continuity using equation (4) and the equation

$$A_{\mathrm{P}}^{*} U_{\mathrm{P}}^{***} = \sum A_{M}^{*} U_{M}^{**} - \Delta p^{*} + S_{u}. \tag{22}$$

Scheme 10

This is an extension of Scheme 9 with an additional step that enforces mass continuity. The additional step is as follows.

Step 4, correct pressure and velocities in order to satisfy continuity using equation (4) and the equation

$$A_{\mathrm{P}}^{*} U_{\mathrm{P}}^{****} = \sum A_{M}^{*} U_{M}^{**} - \Delta p^{**} + S_{u}. \tag{23}$$

Scheme 11

In contrast with Scheme 10, an alternative extension of Scheme 9 is to use an extra step which requires velocity correction terms in its solution.

Step 4, correct pressure and velocities in order to satisfy continuity using equation (4) and the equation

$$A_{\mathrm{P}}^{*} U_{\mathrm{P}}^{****} = \sum A_{M}^{*} U_{M}^{***} - A p^{**} + S_{u}. \tag{24}$$

4. TIME DERIVATIVES

The standard PISO scheme[7] uses the Crank–Nicolson scheme for the time derivative term. This option seems to be the most obvious as it requires the minimal amount of memory storage of the velocity fields. Also, the standard PISO scheme splits the operators so that they are linearly implicit, which means that $U^{n}U^{n+1}$ terms are obtained. If this is the case, it is straightforward to show that a time derivative term based on the $n+\frac{1}{2}$ time level should give a good approximation. However, some of the previous schemes presented attempt to update the coefficient terms so that we actually obtain fully implicit terms $U^{n+1}U^{n+1}$. If this is successfully achieved, it is anticipated that one-sided forward differencing will be a better approximation as all the terms are based on the $n+1$ level.

The original SIMPLE scheme[5] does not use real time stepping but introduces the concept of 'pseudo time stepping'; in this context the standard SIMPLE scheme also uses the Crank–Nicolson scheme.

4.1. Crank–Nicolson ($n+\frac{1}{2}$ level)

The Crank–Nicolson scheme is discussed in Reference 40:

$$\left.\frac{\partial U}{\partial t}\right|_{n+1/2} = \frac{U^{n+1} - U^{n}}{\Delta t} + O(\Delta t^{2}). \tag{25}$$

Kim and Benson[10] state that this term is first-order-accurate when applied to the PISO scheme. This is true if we consider the PISO scheme to be fully implicit; however, as the coefficient terms are explicitly treated, this assumption does not appear to be wholly valid. It is therefore more enlightening to regard the Crank–Nicolson scheme as being second-order-accurate based around the $n+\frac{1}{2}$ time level. Note that it may prove to be more accurate to base the time derivative on the $n+\frac{1}{2}$ level for the standard PISO scheme.

4.2. Leap-frog (nth level)

The leap-frog scheme is discussed in Reference 40:

$$\left.\frac{\partial U}{\partial t}\right|_n = \frac{U^{n+1} - U^{n-1}}{2\Delta t} + O(\Delta t^2). \tag{26}$$

This time derivative is expected to perform poorly in terms of accuracy, as the schemes under consideration are semi-implicit, and the main motivation of deriving the variants from the SIMPLE and PISO schemes is to obtain a fully implicit scheme, the advantage of using an implicit approach as opposed to an explicit approach.[5,40]

4.3. One-sided forward differencing (n + 1 level)

This scheme is discussed in Reference 41 and is used by Kim and Benson[10] in their simulations with the PISO scheme:

$$\left.\frac{\partial U}{\partial t}\right|_{n+1} = \frac{3U^{n+1} - 4U^n + U^{n-1}}{2\Delta t} + O(\Delta t^2). \tag{27}$$

In conclusion, the time derivative terms presented above are all second-order-accurate (based on different time levels). We have also assumed that the time step value is constant during the simulation.

5. RESULTS

The schemes presented in Section 3 are applied to the solution of laminar flow around a square cylinder to test the 'implicit' nature of the schemes. We have also carried out a transient flow calculation of steady state laminar flow over a backward-facing step for two grid systems, mainly in order to evaluate the discretization schemes used in the present investigations.

5.1. Flow around a square cylinder

Planar flow around a square cylinder is of great interest in fluid mechanics because the geometry is simple and fundamental; also, for certain Reynolds numbers it exhibits the phenomena of vortex shedding. The flow behaviour is similar to the extensively studied flow around a circular cylinder.[42] The flow readily becomes unsteady at low Reynolds numbers and is therefore a good test case for a transient solution procedure.

Unsteady laminar flows around a square cylinder tend to be only studied numerically, although there are some experimental data.[43] The numerical studies tend to use the flow problem as a computational exercise comparing differencing schemes[44] or outlet boundary conditions[45] or as a validation exercise.[46,47] The main issue for low-Reynolds-number flows is how the width-to-height ratio of the cylinder ($w:h$) affects the flow. Although cylinders that do not have a 1 : 1 ratio are not square, these studies closely relate to the present test case and usually include an example of flow

around a square cylinder.[48–51] The effect of varying the angle of attack has been studied by Davis and Moore.[48] Davis *et al.*[43] have also studied numerically how the flow is affected by enclosing the square cylinder in a channel. At low Reynolds numbers this suppresses the unsteady nature of the flow. A similar issue has been studied by Arnal *et al.*,[52] who investigated three cases for a square cylinder, one of which included the freestream case. The other two cases were flow past a square cylinder which is in contact with either a moving or a stationary solid boundary. The velocity of the moving solid boundary is set to the freestream; similarities between this case and the freestream case were observed.

Finally, the pioneering numerical work of Fromm and Harlow[53] attempted to predict the onset of vortex shedding. The study by Fromm and Harlow predicted the flow around a 1 : 6 cylinder for a range of Reynolds numbers from very low values ($Re = 10$) to reasonably high values ($Re = 1000$). Unfortunately, the central differencing scheme used tended to fail at the high Reynolds numbers. The study found that the flow became unstable at $Re = 50$. Kelkar and Patankar[54] also studied a series of low-Reynolds-number cases in order to predict when the flow becomes unsteady. They concluded that Reynolds numbers above 50 are unstable by using a steady state flow solution procedure with a perturbation variable to predict when the flow becomes unstable.

5.2. *Numerical analysis of square cylinder problem*

The main test case used in the present investigations is laminar flow around a square cylinder for a Reynolds number $Re = 250$. This problem has been studied numerically.[46–49] Okajima[50] has undertaken predictions for the test cases of $Re = 150$ and 500, so his study is also of interest. In addition to the numerical calculations, Okajima[49] and Davis and Moore[48] have presented limited experimental data for the test case of $Re = 250$. The Reynolds number is based on the diameter of the cylinder and the freestream velocity.

Two interpolation schemes were used in the present study: the hybrid scheme[55] and the SOUD scheme.[22] The hybrid scheme assumes either linear or 'first-order upwind' interpolation depending on the grid Peclet number. The SOUD scheme assumes second-order upwind interpolation between grid points and is generally considered a superior scheme.[56,57]

The geometry of the problem is illustrated in Figure 1. Three grid systems 1–3 have been used in the present investigation which have grid resolutions of 50×50, 110×90 and 200×150 respectively. The grid systems are non-uniformly distributed with refinements of ($\Delta x = 0{\cdot}098$, $\Delta y = 0{\cdot}127$), ($\Delta x = 0{\cdot}098$, $\Delta y = 0{\cdot}065$) and ($\Delta x = 0{\cdot}063$, $\Delta y = 0{\cdot}037$) away from the square cylinder. Grid systems 2 and 3 have the same problem domain. All problem domains place the upper and lower boundaries $8D$ away from the cylinder, where D is the diameter of the cylinder. Length scales are non-dimensionalized by D. The problem domain for grid 1 places the inlet boundary $3D$ upstream of the cylinder and the outlet boundary $9D$ away from the cylinder. The problem domain for grids 2 and 3 places the inlet boundary $4D$ upstream of the cylinder and the outlet boundary $15D$ away from the cylinder. A uniform velocity profile is set at the inlet boundary. The upper/lower boundary condition sets $\mathrm{d}u/\mathrm{d}y = 0$ and $v = 0$ across it. The outlet boundary condition is calculated via extrapolation.[21] (The SIMPLE and PISO methodologies do not require outlet pressure terms.) The present outlet condition was developed for steady state flows; nevertheless, the current results appear to be acceptable in the outlet region. A possible adaptation of the outlet condition is discussed in Reference 58. This is an area of future research.

The matrix is solved using the tridiagonal matrix algorithm[40] (TDMA); the algorithm sweeps through the solution domain 125 times. This means that the matrix array is solved to an accuracy of at least 0·1% of the maximum change.

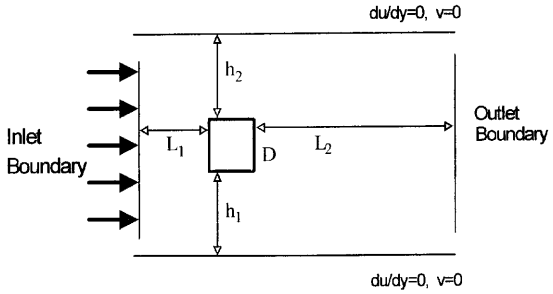

Figure 1. Illustration of problem of flow around a square cylinder

5.3. *Flow predictions around a square cylinder*

If the flow around the cylinder is impulsively started, then initially the velocity field is characterized by two main features, namely (i) the stagnation point on the axis of symmetry behind the cylinder and (ii) the centre of recirculation. Initially the flow separates from the downstream corner, forming recirculation regions behind the cylinder. The upstream flow that is forced away from the cylinder recovers slowly downstream, returning to its original uniform profile. The recirculation regions grow in size with respect to time. However, the growth in the recirculation regions is not maintained and the vortex regions start to oscillate, building up perturbation effects from one another until the system finally becomes unstable and vortices are shed. The vortices are periodically shed from opposing upper and lower surfaces, where the frequency increases with time until it finally maintains a constant value. This behaviour is observed in Figure 2, which shows the streamlines at non-dimensionalized times of 10–90 at intervals of 10. There are strong pressure gradients at the leading edge of the cylinder. The pressure recovers behind the cylinder, initially forming symmetric peaks which start to oscillate up and down as the flow becomes unsteady. When the vortex shedding starts, the pressure peaks are also 'shed', appearing in the wake where the flow changes direction. The pressure field is shown in Figure 3 at non-dimensionalized times of 10–90 at intervals of 10. The previous results presented are taken from predictions using the standard PISO scheme with the SOUD scheme and using non-dimensionalized time steps $\Delta T = 0{\cdot}05$ and grid 2. The figures do not show the complete problem domain, only the region $3D$ upstream, $10D$ downstream and $5D$ either side of the cylinder.

5.4. *Comparison of numerical results*

A start-up file at $T = 100$ was created in order to make comparisons. This is done for the following reasons: (i) we want to make sure that the transient nature of the flow has been established, which can take up to 100 non-dimensionalized time units for certain simulations; (ii) initially large C_D values are predicted because the flow is impulsively started; (iii) we can ensure to a fair degree of accuracy that the various calculations will be in phase. Tests comparing the various schemes were undertaken only for grid 1, using a relatively large time step $\Delta T = 0{\cdot}05$. In fact, this time step is probably approaching the maximum size of time step we could use in order to capture the physics of the flow (the drag fluctuations have a sinusoidal non-dimensionalized period of three). Since we are only interested in the implicit treatment of the results, we have taken as a datum set of results the predictions using the PISO scheme with a small time step $\Delta T = 0{\cdot}005$. These datum results will of course be subject to the same spatial discretization errors as the other sets of results. The effect of the spatial discretization error is initially demonstrated in Table I, which includes predictions from the

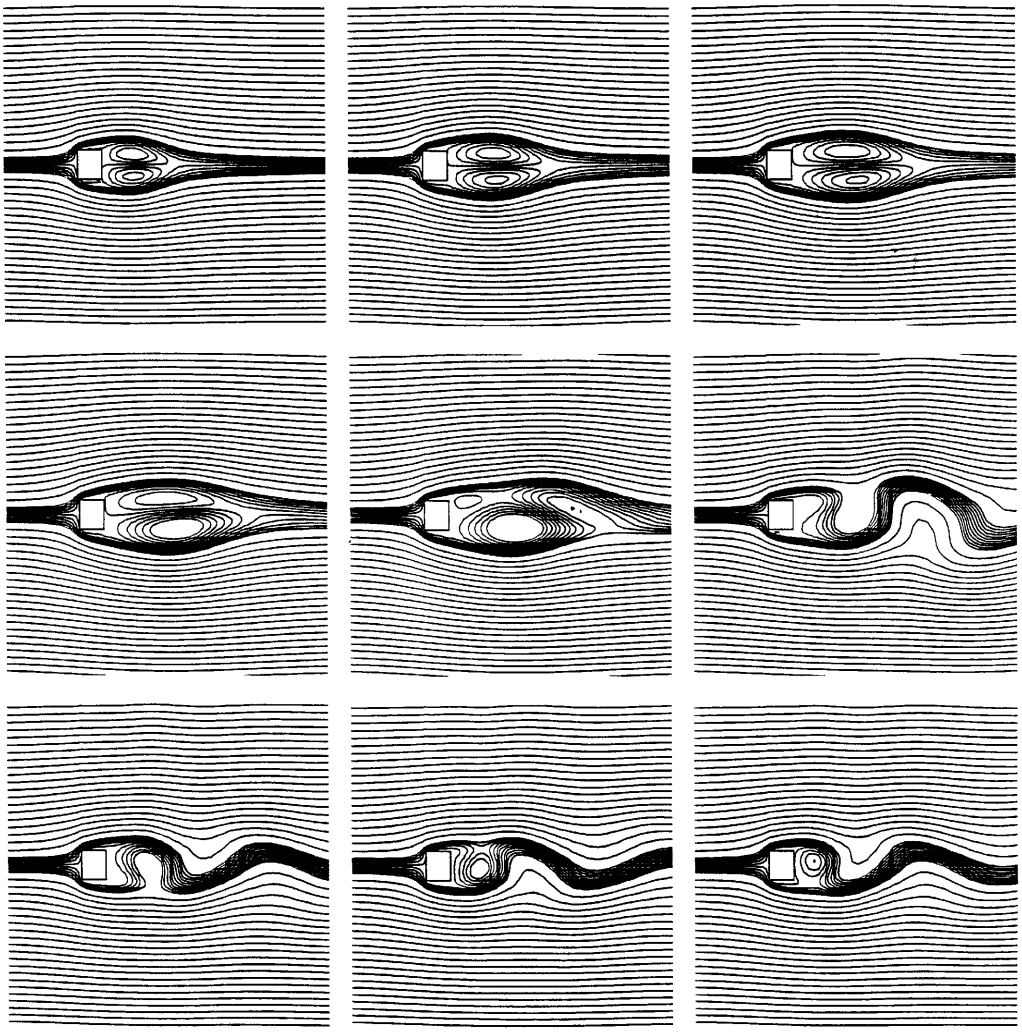

Figure 2. Streamlines of flow around a square cylinder at intervals of 10 non-dimensionalized time units from an impulsive start

PISO scheme with a time step $\Delta T = 0{\cdot}05$ using the hybrid differencing scheme.[55] The other sets of predictions use the SOUD scheme,[22] as is the case for the majority of the predictions.

The results shown in Table I use the Crank–Nicolson time derivative scheme. Table I shows predictions of the maximum and minimum lift coefficients $C_{\mathrm{L}}^{\mathrm{MAX}}$ and $C_{\mathrm{L}}^{\mathrm{MIN}}$ as well as the maximum, minimum and average drag coefficients $C_{\mathrm{D}}^{\mathrm{MAX}}$, $C_{\mathrm{D}}^{\mathrm{MIN}}$ and $C_{\mathrm{D}}^{\mathrm{AVE}}$. The coefficient terms are defined as in previous studies.[48] The table also shows the average period of lift oscillations, T_{L}, which is the inverse of the Strouhal number. The T_{D} is the total period of drag oscillations. Physically one might expect that T_{D} should be the same as T_{L}; however, as Vickery[59] argues, this is not necessarily the case.

Schemes 1–3 failed to predict stable oscillations and the results eventually diverged. The hybrid differencing scheme predicts a significantly different set of results, demonstrating how the scheme essentially behaves as a first-order upwind differencing scheme. The schemes that managed to predict

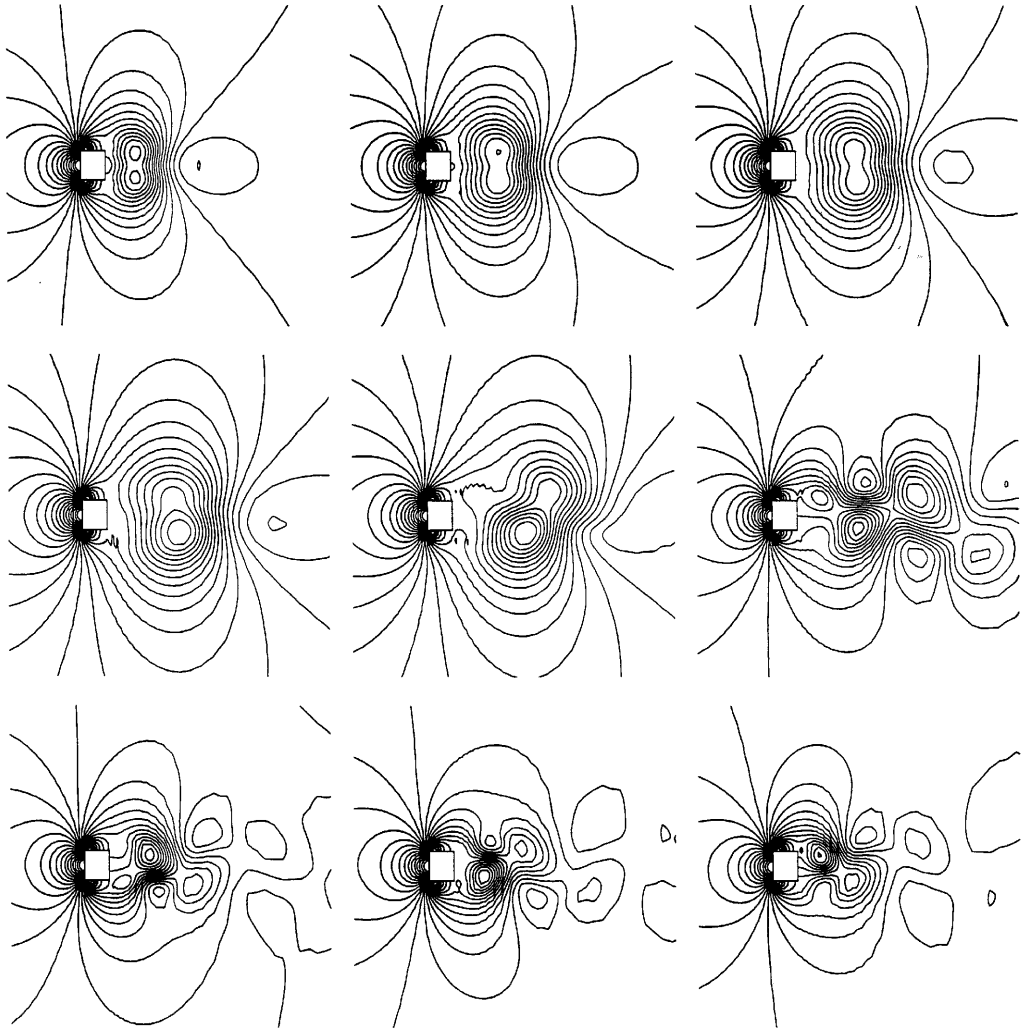

Figure 3. Pressure contours of flow around a square cylinder at intervals of 10 non-dimensionalized time units from an impulsive start

stable oscillations are compared with the datum set of results. The largest discrepancies occur in the peak-to-peak results, with errors as large as 20%. For this set of comparisons, Scheme 4 (the PISO scheme) gives the best agreement with the datum values. Scheme 5 predicts results almost identical to Scheme 4, albeit with a very slight improvement. The next largest discrepancy occurs for the mean drag coefficients, where again Scheme 4 predicts closest agreement with the datum results. However, the oscillation periods T_L and T_D are, relatively speaking, poorly predicted by Scheme 4 and better comparisons are made with Schemes 6, 7 and 9–11. Despite this fact, Scheme 4 only has discrepancies of 1% and 3% for the T_L and T_D terms, which are therefore still in very good agreement with the datum results, whereas the other schemes tend to fail with greater magnitude for other flow characteristics. Schemes 6, 7 and 9–11 predict overall reasonable agreement with the datum results. It is interesting to note that the Scheme 8 results have greater discrepancies with the datum results than Scheme 4 and yet Scheme 8 is simply Scheme 4 applied twice; far from being discouraging, this

Table I. Flow characteristics for problem of flow around a square cylinder using various algorithms and Crank–Nicolson scheme

	C_L^{MAX}	C_L^{MIN}	T_L	C_D^{MAX}	C_D^{MIN}	C_D^{AVE}	T_D
Datum	1·006	−0·998	6·112	1·876	1·616	1·744	6·019
Hybrid	0·348	−0·341	7·551	1·686	1·648	1·668	7·548
1	—	—	—	—	—	—	—
2	—	—	—	—	—	—	—
3	—	—	—	—	—	—	—
4	0·894	−0·886	6·177	1·832	1·622	1·726	6·177
5	0·895	−0·888	6·177	1·833	1·621	1·726	6·174
6	0·855	−0·796	6·075	1·815	1·609	1·679	6·076
7	0·872	−0·824	6·057	1·809	1·612	1·685	6·056
8	0·872	−0·824	6·044	1·812	1·612	1·685	6·044
9	0·856	−0·789	6·085	1·816	1·609	1·679	6·082
10	0·876	−0·827	6·060	1·813	1·614	1·688	6·062
11	0·849	−0·781	6·066	1·815	1·607	1·676	6·070

confirms our supposition made in Section 4 that the standard PISO scheme may predict better results with the Crank–Nicolson time derivative as the flow solution uses $U^n U^{n+1}$ terms.

Further results were run for only the time period $T = 100$–125, which is estimated to give a reasonable time period for comparisons. Table II shows the same flow characteristics as presented in Table I for Schemes 1–11 but in this case for the leap-frog time derivative. The datum results have different values compared with Table I because the flow characteristics are only compared for the time period $T = 100$–125, otherwise the datum results are the same as those used in Table I. Compared with the previous set of results, more schemes failed to predict stable oscillations. Thus, in addition to Schemes 1–3 failing, Schemes 6, 9 and 10 diverged. In contrast with the results using the Crank–Nicolson scheme, the discrepancies have increased as expected. Curiously, however, the lift predictions for Scheme 11 have improved. As we have previously observed, Schemes 4 and 5 predicted virtually identical results (this is in fact in agreement with Issa's[7] analysis of the PISO scheme). Overall, the deterioration in the results is most significant for the peak-to-peak values as well as the T_L values. Schemes 4, 5 and 11 give the best agreement for the majority of the datum values.

Table II. Flow characteristics for problem of flow around a square cylinder using various algorithms and leap-frog scheme

	C_L^{MAX}	C_L^{MIN}	T_L	C_D^{MAX}	C_D^{MIN}	C_D^{AVE}	T_D
Datum	1·006	−0·997	6·207	1·876	1·616	1·742	6·006
1	—	—	—	—	—	—	—
2	—	—	—	—	—	—	—
3	—	—	—	—	—	—	—
4	0·854	−0·793	6·234	1·818	1·613	1·689	6·258
5	0·853	−0·794	6·234	1·817	1·613	1·688	6·258
6	—	—	—	—	—	—	—
7	0·844	−0·747	6·117	1·793	1·596	1·660	6·158
8	0·840	−0·747	6·101	1·797	1·599	1·664	6·130
9	—	—	—	—	—	—	—
10	—	—	—	—	—	—	—
11	0·879	−0·779	6·300	1·800	1·603	1·668	6·128

Table III shows the same flow characteristics predicted by Schemes 1–11 using the one-sided forward differencing time derivative term, again for the time period $T = 100$–125. For this case, all schemes predict stable oscillatory flow. Generally speaking, despite a slight disagreement with the T_L value, Schemes 2, 7, 8 and 10 predict superior results for this case. Very poor predictions are made using Scheme 6, whereas Schemes 4 and 5 again predict virtually identical results which are reasonable and Scheme 1 predicts results to the same order of accuracy.

The overall accuracy of the various schemes with the various time derivatives is examined in Table IV. In order to make a direct comparison with the datum results, the standard deviations of the lift and drag coefficients are calculated for the time period $T = 100$–125. The results are presented in Table IV.

The PISO results using the hybrid scheme with the Crank–Nicolson scheme are initially very good in comparison with some of the other schemes: $\sigma_L = 4{\cdot}09 \times 10^{-1}$ and $\sigma_D = 8{\cdot}49 \times 10^{-2}$. However, they deteriorate rapidly if the time period for the comparison is increased, unlike other schemes. The reason for this is that the hybrid results are more prone to become out of phase with the datum results

Table III. Flow characteristics for problem of flow around a square cylinder using various algorithms and one-sided forward differencing scheme

	C_L^{MAX}	C_L^{MIN}	T_L	C_D^{MAX}	C_D^{MIN}	C_D^{AVE}	T_D
Datum	1·006	−0·997	6·207	1·876	1·616	1·742	6·006
1	1·113	−1·098	6·167	1·988	1·613	1·782	6·114
2	0·974	−0·964	6·084	1·867	1·609	1·734	6·014
3	1·173	−1·159	6·151	2·011	1·615	1·792	6·086
4	1·107	−1·092	6·134	1·984	1·613	1·781	6·114
5	1·108	−1·093	6·134	1·985	1·612	1·780	6·100
6	0·895	−0·889	6·051	1·858	1·610	1·727	6·030
7	0·949	−0·940	6·034	1·863	1·612	1·733	6·000
8	0·943	−0·936	6·017	1·864	1·612	1·733	5·986
9	0·890	−0·882	6·101	1·855	1·610	1·726	6·014
10	0·956	−0·946	6·051	1·868	1·612	1·735	6·000
11	0·887	−0·869	6·034	1·853	1·609	1·724	6·014

Table IV. Estimated accuracy of lift and drag coefficients using various algorithms and various time derivativeschemes

	Crank–Nicolson		Leap-frog		Forward difference	
	σ_L	σ_D	σ_L	σ_D	σ_L	σ_D
1	—	—	—	—	$7{\cdot}00 \times 10^{-2}$	$5{\cdot}75 \times 10^{-2}$
2	—	—	—	—	$2{\cdot}36 \times 10^{-2}$	$6{\cdot}47 \times 10^{-3}$
3	—	—	—	—	$1{\cdot}00 \times 10^{-1}$	$5{\cdot}01 \times 10^{-2}$
4	$3{\cdot}48 \times 10^{-2}$	$2{\cdot}51 \times 10^{-2}$	$1{\cdot}22 \times 10^{-1}$	$1{\cdot}71 \times 10^{-2}$	$6{\cdot}97 \times 10^{-2}$	$5{\cdot}95 \times 10^{-2}$
5	$3{\cdot}33 \times 10^{-2}$	$2{\cdot}54 \times 10^{-2}$	$1{\cdot}20 \times 10^{-1}$	$1{\cdot}73 \times 10^{-2}$	$7{\cdot}12 \times 10^{-2}$	$5{\cdot}99 \times 10^{-2}$
6	$1{\cdot}37 \times 10^{-1}$	$7{\cdot}81 \times 10^{-3}$	—	—	$6{\cdot}12 \times 10^{-2}$	$1{\cdot}72 \times 10^{-2}$
7	$1{\cdot}12 \times 10^{-1}$	$1{\cdot}46 \times 10^{-2}$	$2{\cdot}05 \times 10^{-1}$	$3{\cdot}81 \times 10^{-2}$	$3{\cdot}32 \times 10^{-2}$	$7{\cdot}99 \times 10^{-3}$
8	$1{\cdot}22 \times 10^{-1}$	$1{\cdot}23 \times 10^{-2}$	$2{\cdot}03 \times 10^{-1}$	$3{\cdot}36 \times 10^{-2}$	$3{\cdot}55 \times 10^{-2}$	$1{\cdot}06 \times 10^{-2}$
9	$1{\cdot}46 \times 10^{-1}$	$1{\cdot}02 \times 10^{-2}$	—	—	$6{\cdot}82 \times 10^{-2}$	$1{\cdot}34 \times 10^{-2}$
10	$1{\cdot}20 \times 10^{-1}$	$1{\cdot}51 \times 10^{-2}$	—	—	$3{\cdot}49 \times 10^{-2}$	$6{\cdot}56 \times 10^{-3}$
11	$1{\cdot}53 \times 10^{-1}$	$9{\cdot}16 \times 10^{-3}$	$1{\cdot}61 \times 10^{-1}$	$3{\cdot}34 \times 10^{-2}$	$7{\cdot}35 \times 10^{-2}$	$1{\cdot}52 \times 10^{-2}$

as the artificial viscosity is not immediately established in the flow. The results in Table IV confirm our previous analyses, namely that Schemes 4 and 5 predict very similar results. Scheme 4 is the superior scheme for both the Crank–Nicolson and leap-frog methods. However, superior results can be achieved using either Schemes 2, 7, 8 or 10 provided that it is used in conjunction with the forward differencing scheme for the time derivative. While most schemes improve in accuracy upon changing the basis of the time derivative from n to $n+\frac{1}{2}$ to $n+1$, the results for Scheme 4 deteriorates from $n+\frac{1}{2}$ to $n+1$.

Next the efficiency of the various schemes is considered by presenting the CPU run time per solution cycle, *Rc*. This value is estimated to have an error of the order of $\pm 0{\cdot}08$ s. The results are presented in Table V. In contrast with the results in Table V, the PISO results using the hybrid scheme with the Crank–Nicolson scheme recorded an *Rc* of 2·24 s. It is worth noting that this rate is comparable with the SOUD scheme results. This result is interesting because a steady state problem using a higher-order differencing scheme invariably requires a longer run time per cycle. For example, Patel and Markatos[56] compared the QUICK scheme[60] with the hybrid differencing scheme and found that the computational cost approximately doubled for steady state problems. Note that the QUICK scheme and the SOUD scheme are closely related to one another.[21] The fact that the computational cost does not increase with a more accurate differencing scheme demonstrates the importance of the transient nature of the flow (artificial viscosity is less likely to be permanently established in the flow); also, the temporal terms in the discretization procedure dominate the numerical solution of the flow.

The results from Table V in conjunction with the results from Table IV allow us to deduce which are the most efficient schemes for the current problem. Since Schemes 7 and 8 are significantly more expensive than Scheme 4, the attraction of using these schemes diminishes as they are not particularly more accurate than Scheme 4. We are left with Schemes 2 and 10 using the forward differencing scheme as well as Scheme 4 using the Crank–Nicolson scheme to recommend for further investigation.

Next the stability of 'robustness' of the schemes is crudely considered by attempting to run a simulation for 300 iterations and monitoring the velocity and pressure changes. If there are spurious oscillations in the residual history or if the changes tend to infinity, the simulation is defined to be diverging. The (non-dimensional) time step values tested are 0·3, 0·1, 0·06, 0·03 and 0·01. The successful non-dimensionalized time steps are recorded in Table VI. The results in Table VI only give an estimate of the stability of the various schemes tested. It is only an estimate as diverging

Table V. CPU run time per cycle for various algorithms and various time derivative schemes

	Crank–Nicolson *Rc* (s)	Leap-frog *Rc* (s)	Forward difference *Rc* (s)
1	—	—	1·20
2	—	—	2·44
3	—	—	2·20
4	2·12	2·36	2·16
5	3·20	3·41	3·12
6	2·42	—	2·37
7	3·64	3·37	3·53
8	4·22	4·26	4·28
9	1·39	—	1·35
10	2·41	—	2·33
11	2·39	2·51	2·49

Table VI. Maximum successful time step applied to various algorithms using various time derivative schemes

	Crank–Nicolson ΔT_{max}	Leap-frog ΔT_{max}	Forward difference ΔT_{max}
1	0·03	0·01	0·06
2	0·03	0·01	0·06
3	0·03	0·01	0·06
4	0·10	0·06	0·10
5	0·10	0·10	0·10
6	0·06	0·03	0·06
7	0·10	0·06	0·10
8	0·10	0·06	0·10
9	0·03	0·01	0·06
10	0·06	0·03	0·06
11	0·10	0·10	0·10

behaviour (as defined above) is dependent not only the scheme's stability but also on the physics of the flow. It is unreasonable to expect physically accurate solutions for time steps of 0·3 as the sinusoidal period of drag is three non-dimensional time units. The results from Table VI reflect the previous calculations undertaken. The trend is for the schemes to become more robust if the basis of the time derivative is increased from n to $n+1$. The scheme least affected by altering the time derivative is Scheme 5, which suggests that if we solve our matrix solution procedure to greater accuracy, then this would also be true for Scheme 4. The stability of Scheme 4 is greater than that of Schemes 2 and 10 which are recommended alternatives.

Finally the issue of grid resolution is dealt wiith, although we do not claim that the previous results presented are grid-independent, since our motivation has been to establish the implicit nature of the various schemes and this can be done despite spatial errors. However, it is useful to assess the accuracy of our current predictions and to make comparisons wiith other numerical predictions. Unfortunately the only experimental data are on the Strouhal number, which is found by Okajima[49] to be 0·145; the Strouhal number is the inverse of T_L, whose experimental value is therefore $T_L = 6{\cdot}896$. Results from the three grid systems are presented in Table VII as well as available data from other numerical predictions.

The variation in the predictions in Table VII is a matter of concern. It is difficult to resolve the possible reasons why there is a large discrepancy between the current set of results and other numerical results. The predictions are dependent on the grid resolution, the time step value as well as the algorithm methodology. Also, in the present investigation we discovered that the oscillating flow pattern takes a very long time to stabilize in terms of shedding period and drag wavelength. Larger T_L and T_D values are obtained at $T = 50$ compared with $T = 200$. After a time interval of $T = 150$ from the impulsive start the flow does not appear to change in behaviour and results are therefore taken from the time period $T = 150$–200. Grid refinement around the cylinder in the region immediately downstream of the cylinder is essential for accurate prediction of the lift coefficient; the importance of grid refinement in order to accurately predict flow separation was also found by Ramaswamy.[47] Since we are interested in successfully predicting separation along the upper and lower boundaries of the cylinder, the reason why such refinement is important is obvious; however, it is surprising that this phenomenon is not greatly influenced by the far-downstream vortex shedding. It has been observed in the present investigation that a larger time step value reduces the accuracy of the predictions, although this is dependent on the type of solution algorithm applied. In the above

Table VII. Present predictions of flow characteristics using PISO scheme. (b) Numerical predictions of flow characteristics using various methodologies

	C_L^{MAX}	C_L^{MIN}	T_L	C_D^{MAX}	C_D^{MIN}	C_D^{AVE}	T_D
(a)							
Grid 1, SOUD	1·006	−0·998	6·112	1·876	1·616	1·744	6·018
Grid 2, SOUD	1·060	−1·055	6·712	1·869	1·662	1·769	6·692
Grid 3, SOUD	1·238	−1·230	6·498	1·884	1·632	1·759	7·026
Grid 1, hybrid	0·348	−0·341	7·551	1·686	1·648	1·668	7·548
Grid 2, hybrid	0·407	−0·394	7·001	1·588	1·541	1·564	5·628
Grid 3, hybrid	0·435	−0·421	6·158	1·644	1·579	1·612	6·266
(b)							
Okajima[50 b]	∼0·7	∼−0·7	7·692	N/S[a]	N/S	∼1·8	∼7·6
Okajima[50 c]	∼1·2	∼−1·2	7·143	N/S	N/S	∼2·2	∼6·9
Gresho and Chan[46 d]	1·40	−1·40	7·8	2·12	1·85	∼1·98	7·7
Gresho and Chan[46 d]	1·32	−1·32	7·7	2·05	1·79	∼1·92	7·7
Gresho and Chan[46 d]	1·75	−1·75	7·9	2·39	1·95	∼2·17	7·5
FIDAP[e]	1·3	−1·3	∼7	N/S	N/S	N/S	∼7 ± 1
Ramaswamy[47]	0·51	−0·50	5·73	1·73	1·59	∼1·66	5·7
Davis and Moore[48]	0·50	−0·50	6·1	1·83	1·73	∼1·78	5·9

[a] N/S, not stated.
[b] Test case for $Re = 150$.
[c] Test case for $Re = 500$.
[d] Different predictions calculated using various schemes.
[e] Calculations undertaken by Gresho and Chan[46] using FIDAP code.

predictions the following time step values were used: Gresho and Chan,[46] $\Delta T = 0{\cdot}05$; Ramaswamy,[47] $\Delta T = 0{\cdot}015$; Davis and Moore,[48] $\Delta T = 0{\cdot}05$; predictions in Table VII, $\Delta T = 0{\cdot}005$.

The predictions of Okajima[50] have been included in Table VII to give a comparison of results for Reynolds number cases above and below the test case of $Re = 250$. If the results are dominated by artificial viscosity, it is anticipated that the results for $Re = 250$ prediction will tend towards the $Re = 150$ set of predictions made by Okajima.[50] Okajima's results suggest that C_D^{AVE} increases with Reynolds number as do the C_D and C_L peak-to-peak values, whereas the periods of lift and drag oscillations decrease with Reynolds number. These features are broadly reflected in our current results, with the main exceptions in some of the drag results. Generally speaking, the SOUD results are in fair agreement with the data of Okajima[50] and Gresho and Chan.[46]

5.5. *Flow over a backward-facing step*

Flow over a backward-facing step is fundamental in design and geometry and is consequently found in a variety of engineering applications. The flow separation process caused by the sudden change in geometry has been used extensively, usually in order to create a recirculation region or a sudden change in pressure. Laminar flow over the backward-facing step configuration investigated by Armaly *et al.*[61] has become a classical numerical problem. The problem has been further endorsed by Gresho,[62] where for a Reynolds number of $Re = 800$ the problem has become a standard test case. Gresho *et al.*[63] have established that the flow is steady state and this test case is therefore used in the present investigation as a steady state test case of the hybrid and SOUD differencing schemes using the standard PISO scheme.

There have been many numerical studies[25–25,63–74] investigating laminar flow over a backward-facing step similar to the configuration of Armaly *et al.*[61] Durst and Pereira[68] and Gresho *et al.*[63]

considered the growth of the recirculation regions with time; Gresho *et al.*[63] also studied the stability of flow for a high-Reynolds-number case. Thangam and Knight[71] studied the effect of the step height on the downstream flow. Barton considered the effect of the viscous drag from the upper boundary[23] and also investigated the entrance effect of the flow configuration.[25]

5.6. Numerical analysis of backward-facing step problem

The present investigation is a continuation from the previous studies[23–25] as well as the current numerical investigation. In this case we are interested in a time-dependent simulation of the flow that has been impulsively started. The flow configuration is illustrated in Figure 4, where the geometry has a very small inlet channel and the inlet flow profile is a parabola. The channel expansion number is fixed at two, where the expansion number is defined as the ratio of the main channel height to the inlet channel height. The channel is 40 step heights in length (this is considered sufficiently long for the outlet boundary condition not to seriously affect the upstream results). The inlet Reynolds number is set at $Re = 800$, where the Reynolds number is defined using twice the inlet channel height, the average inlet velocity and the dynamic viscosity. This definition is the same as that of Armaly *et al.*[61] The numerical analysis used for the flow around a square cylinder is also applied to the problem of flow over a backward-facing step. The standard PISO scheme is used as the solution methodology with a time step value of $\Delta T = 0{\cdot}025$. No-slip boundary conditions are applied for the wall boundaries. Two grid systems were used: 70×50 (grid 1) and 90×70 (grid 2). Both grid systems have refinement close to the solid boundaries and incorporate a very small lip which prevents flapping in the solution.[75]

5.7. Flow predictions for backward-facing step problem

The illustration of the flow configuration shows that there are three main reattachment and separation positions, namely x_1, x_2 and x_3. Position x_1 is the main reattachment point. The strong adverse pressure gradient causes the first separation point x_2; as the pressure recovers, the flow reattaches at x_3. In Figure 5 the predicted streamlines are shown using grid 1 and the SOUD scheme; the figure shows results at non-dimensionalized time values $T = 10$, 20, 30, 40 and 200. Initially the flow has four major recirculation regions; this agrees with the predictions made by Gresho *et al.*[63] The two recirculation regions furthest downstream decrease in vorticity and eventually disappear and finally the flow only has two major recirculation regions, as is shown for $T = 200$. The growth of reattachment and separation positions with time is shown in Figure 6 along the lower boundary and in Figure 7 along the upper boundary. (There is a very small counter-recirculation region at the corner of the step which has been ignored in these illustrations.) Figure 6 shows how the main reattachment position x_1 grows with time and how the second downstream recirculation region is created and

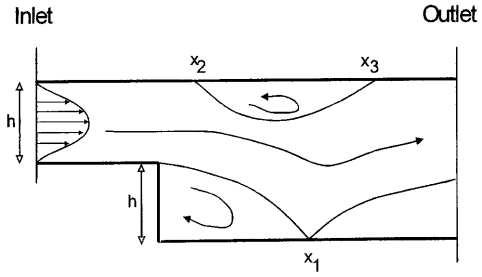

Figure 4. Illustration of problem of flow over a backward-facing step

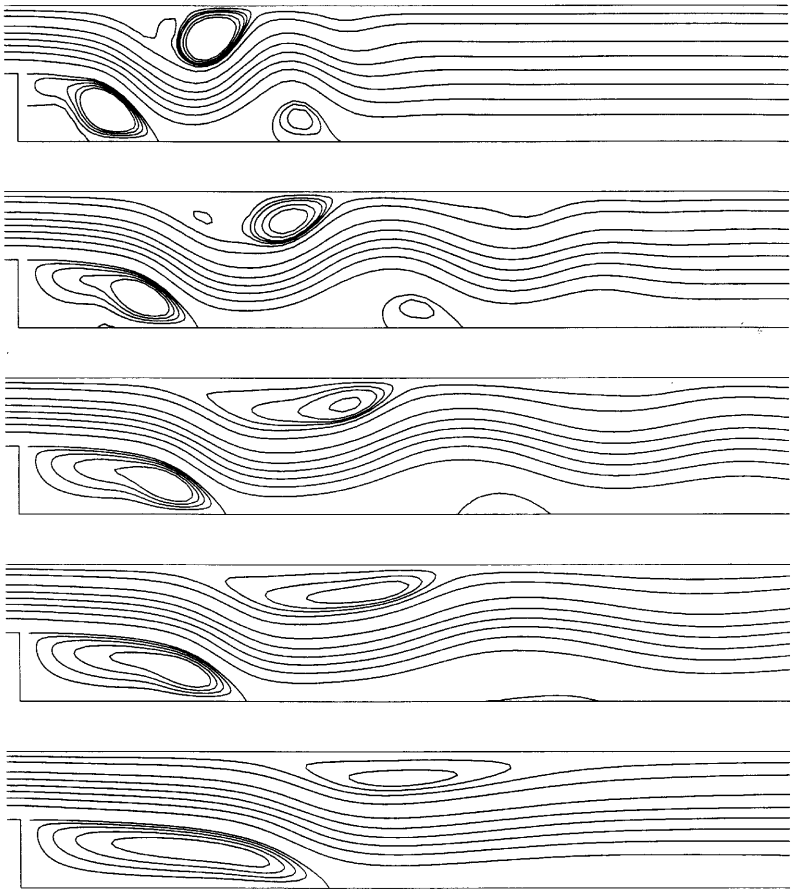

Figure 5. Streamlines of flow over a backward-facing step at non-dimensionalized time values $T = 10$, 20, 30, 40 and 200 (from top to bottom)

destroyed with time. Similar phenomena are observed in Figure 7, where the upstream recirculation grows in size and moves slowly downstream with the growth of the lower recirculation region.

5.8. Comparison of numerical results with benchmark solutions

Benchmark results from References 76 and 77 are shown in Table VIII for reattachment and separation positions (non-dimensionalized with step height) and compared with the present results. The numerical results shown in the table are taken from the results that use the finest grid and best outlet treatment in their particular study. There is quite a variety of disagreement, although the present benchmark solutions are in good agreement with recent predictions made by Gartling,[76] Srinivasan and Rubin (see Reference 77) and Barton.[24] For this problem the superiority of the SOUD scheme in contrast with the hybrid scheme is evident. This is partly observed by the predictions using grid 1 with the hybrid differencing scheme, where the predictions fail to successfully converge, and partly by the predictions using grid 2, where the results completely fail to agree with recent benchmark solutions.

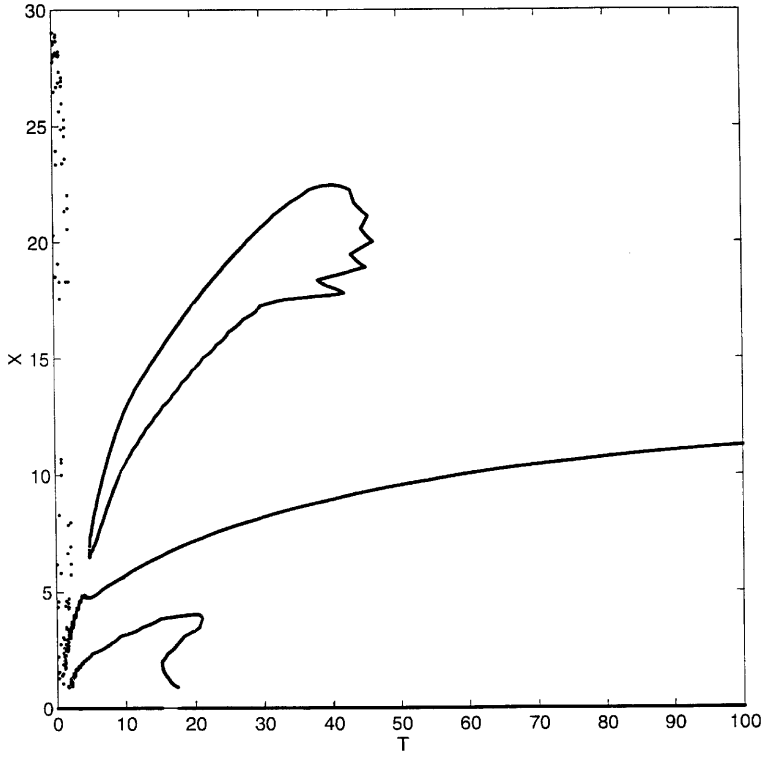

Figure 6. Growth of lower reattachment and separation positions with time

5.9. *Further discussion of results*

Square cylinder results

The reduced stability of the leap-frog method in comparison with other schemes can be argued via stability analysis;[40,41] however, there is a closely related argument which relates to our present method of discretization. The pole coefficient term is dominated by the value of $\rho\Delta x\Delta y/\Delta t$ for the Crank–Nicolson scheme; however, this term becomes $\rho\Delta x\Delta y/2\Delta t$ for the leap-frog scheme. The corresponding source term therefore also reduces in magnitude. This is important when the matrix is inverted, as instabilities caused by the discretization will be suppressed by the large source coefficient term. This argument therefore suggests that a third-order one-sided forward differencing scheme will further improve the stability of most schemes as the pole coefficient increases in magnitude from $3\rho\Delta x\Delta y/2\Delta t$ for second-order accuracy to $11\rho\Delta x\Delta y/6\Delta t$ for third-order accuracy. The third-order one-sided forward differencing scheme is given as

$$\left.\frac{\partial U}{\partial t}\right|_{n+1} = \frac{11U^{n+1} - 18U^n + 9U^{n-1} - 2U^{n-2}}{6\Delta t} + O(\Delta t^3). \tag{28}$$

The PISO scheme using the forward differencing time derivative predicts greater discrepancies than using the Crank–Nicolson scheme; this further demonstrates how the PISO scheme does not behave as a fully implicit scheme and therefore Kim and Benson's[10] assumption that it would improve the predictions is incorrect.

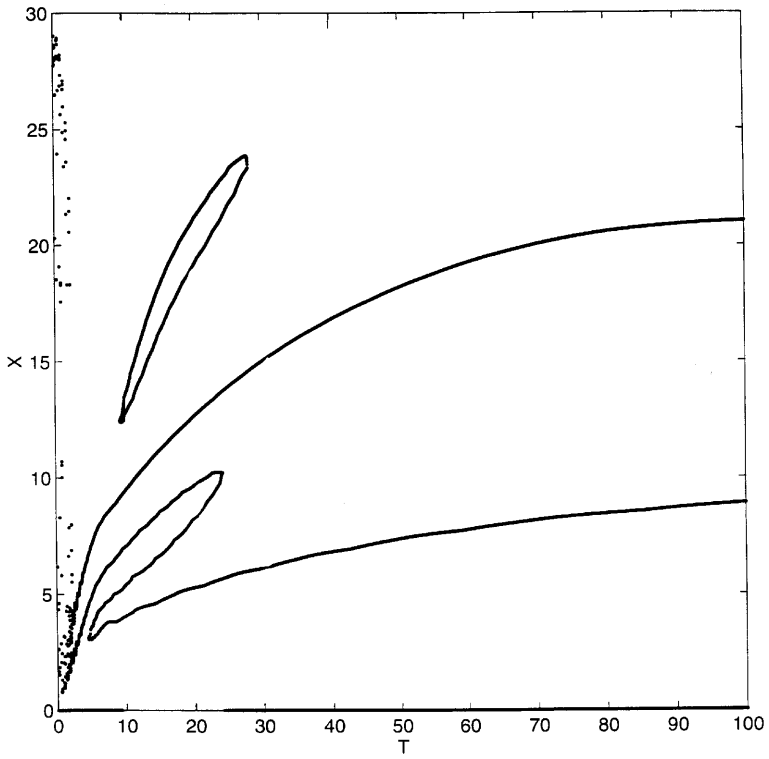

Figure 7. Growth of upper reattachment and separation positions with time

Backward-facing step results

The backward-facing step flow results evaluate the behaviour of the differencing schemes, where the SOUD scheme predicts superior results in contrast with the hybrid scheme. It is perhaps surprising that the predictions using the coarse grid with the hybrid differencing scheme do not

Table VIII. (a) Present predictions of reattachment and separation positions at $T = 200$. (b) Steady state numerical predictions of reattachment and separation positions

	x_1	x_2	x_3
(a)			
Grid 1, SOUD	11·80	9·47	20·62
Grid 2, SOUD	12·19	9·75	20·89
Grid 1, hybrid	—	—	—
Grid 2, hybrid	8·32	6·07	13·55
(b)			
Barton[24]	12·03	9·64	20·96
Srinivasan and Rubin[a]	12·44	10·18	20·50
Gartling[76]	12·20	9·70	20·95
Sohn[70]	11·50	9·40	18·80

[a] Refer to Reference 77.

successfully converge, because the scheme introduces a significant amount of artificial viscosity. Therefore we would expect the 'effective' Reynolds number to reduce, causing shorter reattachment and separation positions as well as rapid convergence. The lack of rapid convergence reflects, perhaps, a phenomenon previously observed in other numerical predictions, namely that an incorrectly defined problem or inadequately resolved problem tends not to converge.

6. CONCLUDING REMARKS

The standard PISO scheme does not behave as a fully implicit scheme and therefore using the Crank–Nicolson scheme as opposed to the one-sided forward differencing scheme is more beneficial for accurate predictions.

The addition of another correction level to the PISO scheme has a negligible effect (Schemes 4 and 5 predict virtually identical results).

The PISO scheme compares very favourably with all other schemes tested in terms of robustness, accuracy and required CPU processing time.

Generally speaking, basing the temporal derivative on the $n+1$ time level as opposed to the nth time level increases the stability of any particular scheme. A third-order one-sided forward differencing scheme may prove to be a successful replacement for the second-order one-sided forward differencing scheme, not least because it should further increase the scheme's stability.

Schemes 2 and 10 using the one-sided forward differencing scheme are recommended alternatives to the PISO scheme. They are arguably more implicit, more accurate and easier to code than the PISO scheme. Both schemes are variants on the SIMPLE methodology; in fact, Scheme 2 is simply a repetition of the SIMPLE algorith. For this reason the scheme in particular is recommended for future applications, especially when studying complex flows, provided that a small time step is applied. The reason why this scheme is recommended for complex flows is that if, for example, turbulent flow is being examined, we then need to solve additional equations such as turbulent transport terms which are dependent on the mean velocity field. Therefore a more implicit treatment of the source terms is achieved by updating all the source terms and coefficient terms before the second solution cycle compared with the PISO scheme or Scheme 10.

Finally, Schemes 2 and 10 are SIMPLE variants and have the advantage that velocity correction terms do not need to be calculated or stored.

ACKNOWLEDGEMENTS

I am grateful for Professor J. E. Ffowcs Williams' help and encouragement with this work. I would also like to thank Dr. S. Cant for helpful discussions and contributions regarding the PISO scheme, and Professor A. Pollard for helpful discussions regarding the simulations.

REFERENCES

1. F. H. Harlow and J. E. Welch, 'Numerical calculations of time-dependent viscous incompressible flow of fluid with free surface', *Phys. Fluids*, **8**, 2182–2189 (1965).
2. S. V. Patankar and D. B. Spalding, 'A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows', *Int. J. Heat Mass Transfer*, **15**, 1787–1806 (1972).
3. L. S. Caretto, A. D. Gosman, S. V. Patankar and D. B. Spalding, 'Two calculation procedures for steady, three-dimensional flows with recirculation', *Proc. 3rd Int. Conf. on Numerical Methods in Fluid Dynamics*, Paris, 1972.
4. G. D. Raithby and G. E. Schneider, 'Numerical solution of problems in incompressible fluid flow: treatment of the velocity–pressure coupling', *Numer. Heat Transfer*, **2**, 417–440 (1979).
5. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, Washington, DC, 1980.

6. J. P. Van Doormaal and G. D. Raithby, 'Enhancements of the SIMPLE method for predicting incompressible fluid flows', *Numer. Heat Transfer*, **7**, 147–163 (1984).
7. R. I. Issa, 'Solution of the implicitly discretized fluid flow equations by operator-splitting', *J. Comput. Phys.*, **62**, 40–65 (1985).
8. R. I. Issa, B. Ahmadi-Befrui, K. R. Beshay and A. D. Gosman, 'Solution of the implicitly discretized reacting flow equations by operator-splitting', *J. Comput. Phys.*, **93**, 388–410 (1991).
9. A Wanik and U. Schnell, 'Some remarks on the PISO and SIMPLE algorithms for steady turbulent flow problems', *Comput. Fluids*, **17**, 555–570 (1989).
10. S. W. Kim and T. J. Benson, 'Comparison of the SMAC, PISO and iterative time-advancing schemes for unsteady flows', *Comput. Fluids*, **21**, 435–454 (1992).
11. L. Cheng and S. Armfield, 'A simplified marker and cell method for unsteady flows on non-staggered grids', *Int. j. numer. meth. fluids*, **21**, 15–34 (1995).
12. P. F. Galpin, J. P. Van Doormaal and G. D. Raithby, 'Solution of the incompressible mass and momentum equations by application of a coupled equation line solver', *Int. j. numer. meth. fluids*, **5**, 615–625 (1985).
13. L. Guoyan, 'A numerical method of solving two-phase fluid flow', in Cheung, Lee and Leung (eds), *Computational Mechanics*, Balkema, Rotterdam, 1991, pp. 1589–1594.
14. S. L. Lee and R. Y. Tzong, 'Artificial pressure for pressure-linked equation', *Int. J. Heat Mass Transfer*, **35**, 2705–2716 (1992).
15. J. P. Van Doormaal and G. D. Raithby, 'An evaluation of the segregated approach for predicting incompressible fluid flows', *ASME Paper 85-HT-9*, 1985.
16. C. A. J. Fletcher, *Computational Techniques for Fluid Dynamics*, Vol. II, Springer, New York, 1988.
17. B. Lakshminarayana, 'An assessment of computational fluid dynamic techniques in the analysis and design of turbomachinery—the 1990 Freeman scholar lecture', *Trans. ASME, J. Fluids Engng.*, **113**, 315–352 (1991).
18. J. Simoneau and A. Pollard, 'Finite volume methods for laminar and turbulent flows using a penalty function approach', *Int. j. numer. meth. fluids*, **18**, 733–746 (1994).
19. J. J. McGuirk and J. M. L. M. Palma, 'The efficiency of alternative pressure-correction formulations for incompressible turbulent flow problems', *Comput. Fluids*, **22**, 77–87 (1993).
20. H. Ströll, F. Durst, M. Peric, J. C. F. Pereira and G. Scheuerer, 'Study of laminar, unsteady piston–cylinder flows', *Trans. ASME, J. Fluids Engng.*, **115**, 687–693 (1993).
21. I. E. Barton, 'A numerical investigation of incompressible dilute particulate flows', *Ph.D. Thesis*, Aerospace Division, School of Engineering, University of Manchester, 1995.
22. W. Shyy, 'A study of finite difference approximations to steady-state, convection-dominated flow problems', *J. Comput. Phys.*, **57**, 415–438 (1985).
23. I. E. Barton, 'Laminar flow past an enclosed and open backward-facing step', *Phys. Fluids A*, **6**, 4054–4056 (1994).
24. I. E. Barton, 'A numerical study of flow over a confined backward-facing step', *Int. j. numer. meth. fluids*, **21**, 653–665 (1995).
25. I. E. Barton, 'The computation of the entrance effect for laminar flow over a backward-facing step', *Int. j. numer. meth. fluids*, in press.
26. S. V. Patankar, 'A calculation procedure for two-dimensional elliptic situations', *Numer. Heat Transfer*, **4**, 409–425 (1981).
27. D. M. Wang, A. P. Watkins and R. S. Cant, 'Three-dimensional diesel engine combustion simulation with a modified EPISO procedure', *Numer. Heat Transfer A*, **24**, 249–272 (1993).
28. G. D. Thiart, 'Finite difference scheme for the numerical solution of fluid flow and heat transfer problems on non-staggered grids', *Numer. Heat Transfer B*, **17**, 43–62 (1990).
29. C. Y. Perng and R. L. Street, 'Three-dimensional unsteady flow simulations: alternative strategies for a volume averaged calculation', *Int. j. numer. meth. fluids*, **9**, 341–362 (1989).
30. W. Shyy and C. S. Sun, 'Development of a pressure-correction/staggered-grid based multigrid solver for incompressible recirculating flows', *Comput. Fluids*, **22**, 51–76 (1993).
31. C. R. Maliska and G. D. Raithby, 'A method for computing three-dimensional flows using non-orthogonal boundary-fitted co-ordinates', *Int. j. numer. meth. fluids*, **4**, 519–537 (1984).
32. P. Tamamidis and D. N. Assanis, 'Prediction of three-dimensional steady incompressible flows using body-fitted co-ordinates', *Trans. ASME, J. Fluids Engng.*, **115**, 457–462 (1993).
33. M. M. Grigor'ev, 'A boundary element formulation using the SIMPLE method', *Int. j. numer. meth. fluids*, **16**, 549–579 (1993).
34. B. R. Latimer and A. Pollard, 'Comparison of pressure–velocity coupling solution algorithms', *Numer. Heat Transfer*, **8**, 635–652 (1985).
35. S. V. Patankar, 'Recent developments in computational heat transfer', *Trans. ASME, J. Heat Transfer*, **110**, 1037–1045 (1988).
36. D. M. Wang and A. P. Watkins, 'Numerical modelling of diesel spray wall impaction phenomena', *Int. J. Heat Fluid Flow*, **14**, 301–312 (1993).
37. W. K. Chow and N. K. Fong, 'Application of field modelling technique to simulate interaction of sprinkler and fire-induced smoke layer', *Combust. Sci. Technol.*, **89**, 101–151 (1993).
38. P. Luchini and A. D'Alascio, 'Multigrid pressure correction techniques for the computation of quasi-incompressible internal flows', *Int. j. numer. meth. fluids*, **18**, 489–507 (1994).
39. J. K. Dukowicz, 'A particle–fluid numerical model for liquid sprays', *J. Comput. Phys.*, **35**, 229–253 (1980).

40. D. A. Anderson, J. C. Tannehill and R. H. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, Hemisphere, Washington, DC, 1984.
41. C. Hirsch, *Numerical Computation of Internal and External Flows*, Vol. 1, *Fundamentals of Numerical Discretization*, Wiley, Chichester, 1988.
42. H. C. Lowe, *Fluid Mechanics*, Macmillan, London, 1979.
43. R. W. Davis, E. F. Moore and L. P. Purtell, 'A numerical–experimental study of confined flow around rectangular cylinders', *Phys. Fluids*, **27**, 46–59 (1984).
44. T. Yoshida, I. Nakamura and T. Watanabe, 'Effect of boundary conditions and finite difference schemes on the numerical analysis of a flow past a square cylinder', *Trans. JSME*, **59B**, 374–381 (1993).
45. I. Nakamura, T. Watanabe and T. Yoshida, 'On the numerical outflow boundary condition of the viscous flow around a square cylinder', *Mem. School Engng., Nagoya Univ.*, **45**, 105–117 (1993).
46. P. M. Gresho and S. T. Chan, 'On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 2: Implementation', *Int. j. numer. meth. fluids*, **11**, 621–659 (1990).
47. B. Ramaswamy, 'Theory and implementation of a semi-implicit finite element method for viscous incompressible flow', *Comput. Fluids*, **22**, 725–747 (1993).
48. R. W. Davis and E. F. Moore, 'A numerical study of vortex shedding from rectangles', *J. Fluid Mech.*, **116**, 475–506 (1982).
49. A. Okajima, 'Strouhal numbers of rectangular cylinders', *J. Fluid Mech.*, **123**, 379–398 (1982).
50. A. Okajima, 'Numerical simulation of flow around rectangular cylinders', *J. Wind Engng. Ind. Aerodyn.*, **33**, 171–180 (1990).
51. G. C. Buscaglia and E. A. Dari, 'Implementation of the Lagrange–Galerkin method for the incompressible Navier–Stokes equations', *Int. j. numer. meth. fluids*, **15**, 23–36 (1992).
52. M. P. Arnal, D. J. Goering and J. A. C. Humphrey, 'Vortex shedding from a bluff body adjacent to a plane sliding wall', *Trans. ASME, J. Fluids Engng.*, **113**, 384–398 (1991).
53. J. E. Fromm and F. H. Harlow, 'Numerical solution of the problem of vortex street development', *Phys. Fluids*, **6**, 975–982 (1963).
54. K. M. Kelkar and S. V. Patankar, 'Numerical prediction of vortex shedding behind a square cylinder', *Int. j. numer. meth. fluids*, **14**, 327–341 (1992).
55. D. B. Spalding, 'A novel finite difference formulation for differential expressions involving both first and second derivatives', *Int. j. numer. meth. engng.*, **4**, 551–559 (1972).
56. M. K. Patel and N. C. Markatos, 'An evaluation of eight discretization schemes for two-dimensional convection–diffusion equations', *Int. j. numer. meth. fluids*, **6**, 129–154 (1986).
57. M. K. Patel, N. C. Markatos and M. Cross, 'A critical evaluation of seven discretization schemes for convection–diffusion equations', *Int. j. numer. meth. fluids*, **5**, 225–244 (1985).
58. H. Laschefski, D. Braes, H. Hank and N. K. Mitre, 'Numerical investigations of radial jet reattachment flows', *Int. j. numer. meth. fluids*, **18**, 629–646 (1994).
59. B. J. Vickery, 'Fluctuating lift and drag on a long cylinder of square cross-section in a smooth and in a turbulent stream', *J. Fluid Mech.*, **25**, 481–494 (1966).
60. B. P. Leonard, 'A stable and accurate convective modelling procedure based on quadratic upstream interpolation', *Comput. Meth. Appl. Mech. Engng.*, **19**, 59–98 (1979).
61. 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**, 473–496 (1983).
62. P. M. Gresho, 'Letter to the editor', *Numer. Heat Transfer A*, **20**, 123 (1991).
63. P. M. Gresho, D. K. Gartling, J. R. Torczynski, K. A. Cliffe, K. H. Winters, T. J. Garrat, A. Spence and J. W. Goodrich, 'Is the steady viscous incompressible two-dimensional flow over a backward-facing step at $Re = 800$ stable?', *Int. j. numer. meth. fluids*, **17**, 501–541 (1993).
64. J. Kim and P. Moin, 'Application of a fractional step method to incompressible Navier–Stokes equations', *J. Comput. Phys.*, **59**, 308–323 (1985).
65. S. C. Caruso, J. H. Ferziger and J. Oliger, 'Adaptive grid techniques for elliptic fluid flow problems', *AIAA Paper 86-0498*, 1986.
66. P. Orlandi, 'Vorticity–velocity formulation for high *Re* flows', *Comput. Fluids*, **15**, 137–149 (1987).
67. A. K. Runchal, 'CONDIF: a modified central-differencing scheme for convective flows', *Int. j. numer. meth. engng.*, **24**, 1593–1608 (1987).
68. F. Durst and J. C. F. Periera, 'Time-dependent laminar backward-facing step flow in a two-dimensional duct', *Trans. ASME, J. Fluids Engng.*, **110**, 289–296 (1988).
69. G. Guj and F. Stella, 'Numerical solutions of high-*Re* recirculating flows in vorticity–velocity form', *Int. j. numer. meth. fluids*, **8**, 405–416 (1988).
70. J. L. Sohn, 'Evaluation of FIDAP on some classical laminar and turbulent benchmarks', *Int. j. numer. meth. fluids*, **8**, 1469–1490 (1988).
71. S. Thangam and D. D. Knight, 'Effect of stepheight on the separated flow past a backward-facing step', *Phys. Fluids A*, **1**, 604–606 (1989).
72. S. Thangam and D. D. Knight, 'A computational scheme in generalized co-ordinates for viscous incompressible flows', *Comput. Fluids*, **18**, 317–327 (1990).

73. J. T. Lin, B. F. Armaly and T. S. Chen, 'Mixed convection heat transfer in inclined backward-facing step flows', *Int. J. Heat Mass Transfer*, **34**, 1568–1571 (1991).
74. T. Kondoh, Y. Nagano and T. Tsuji, 'Computational study of laminar heat transfer downstream of a backward-facing step', *Int. J. Heat Mass Transfer*, **36**, 577–591 (1993).
75. A. Pollard, personal communication, 1986.
76. D. K. Gartling, 'A test problem for outflow boundary conditions—flow over a backward-facing step', *Int. j. numer. meth. fluids*, **11**, 953–967 (1990).
77. R. L. Sani and P. M. Gresho, 'Résumé and remarks on the open boundary condition minisymposium', *Int. j. numer. meth. fluids*, **18**, 983–1008 (1994).

