INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 23, 865–882 (1996)

OVERLAPPING CONTROL VOLUME APPROACH FOR CONVECTION–DIFFUSION PROBLEMS

ATUL KUMAR VERMA AND V. ESWARAN

Department of Mechanical Engineering, Indian Institute of Technology, Kanpur 208 016, India

SUMMARY

This paper introduces a finite volume method to solve 2D steady state convection–diffusion problems on structured non-orthogonal grids. Overlapping control volumes (OCV) are used to discretize the physical domain and the governing equations are solved without transformation. An isoparametric formulation is used to compute diffusion and for upwinding. Four test problems are solved using this and other schemes. The modelling of diffusion in OCV seems very effective even on distorted meshes. The convection modelling in OCV is found to be second-order-accurate, like QUICK, on regular meshes. Although its accuracy is slightly inferior to the latter on rectangular grids, its faster convergence gives it a better overall performance. On non-orthogonal grids, OCV gives better accuracy for a large and practical range of Peclet numbers than does QUICK applied to the transformed equations using the conventional five-point diffusion modelling. The results obtained also demonstrate that the scheme reduces false diffusion to a considerable extent in comparison with the power-law scheme.

KEY WORDS: overlapping control volume; finite volume method; convection–diffusion; numerical diffusion; structured non-orthogonal grid

1. INTRODUCTION

In fluid flow and heat transfer problems, convection and diffusion are usually of primary interest. Therefore a numerical scheme for fluid flow and heat transfer can be tested on the convection–diffusion equation which, while incorporating similar processes, is simpler to solve and, being linear, has some known analytic solutions.

A major difficulty in numerically modelling flow equations is in discretizing the convection term. The use of central differencing for the convection term makes the numerical scheme unstable for grid Peclet numbers greater than two. Simple upwinding, although it gives stability, introduces 'false diffusion'. For intermediate Peclet numbers the upwind scheme can be modified to blend the advantages of the central and simple upwind approximations, as for example in the power-law scheme.[1] The problem of false ('crosswind') diffusion becomes more pronounced when the flow is oblique to the grid lines.[1–7] The skew upstream differencing scheme (SUDS) of Raithby,[5] QUICK of Leonard[6] and its variants[8–10] are able to reduce the level of crosswind diffusion significantly for oblique flows.

The geometrical complexity of domains poses another challenge in applying computational methods to practical flow situations. The use of simple rectilinear finite difference grids for arbitary domains makes boundary condition implementation difficult and may induce computational error due to interpolation. The use of boundary-fitted co-ordinates allows selective refinement near the

CCC 0271–2091/96/090865–18

Received May 1995
Revised November 1995

boundary walls and in the interior of the flow where steep gradients exist and thus reduces the need for additional storage and time which would be required if a global refinement were carried out. However, the implementation of essentially finite difference schemes such as QUICK on body- fitted grids requires us to map the non-orthogonal grid onto a regular rectangular mesh on which the transformed equations (see e.g. Reference 11) are solved. (The general-purpose codes FLUENT, PHOENICS and CFDS-FLOW3D use QUICK in such a role.) The transformed equations are quite complex if the body-fitted grid is not orthogonal. Generally, for orthogonal body-fitted meshes, little control is achievable over the distribution of grid points and their spacing. On the other hand, non-orthogonal meshes can easily be generated by transfinite interpolation or other means and their spacing can be controlled.

However, owing to the complexity of the transformed equations for non- orthogonal grids, other methods have been evolved which solve the governing equations on the physical domain itself. The major work in this area started after Hsu,[12] Rhie,[13] Peric[14] and others. Mukhopadhyay *et al.*[15] and Hwang[16] have developed noteworthy algorithms for flows in arbitary geometries without the use of transformation. The control volume finite element methods[17–20] (CVFEM) for arbitrary geometry are similar in spirit.

In this study we attempt to develop a scheme, applicable directly on non- orthogonal grids in arbitrary geometries, which is simple to implement and yet is accurate. The method uses overlapping control volumes, an idea mentioned by Hirsch[21] but scarcely investigated in the literature. The convection–diffusion equation in Cartesian co-ordinates is solved in domains of varying complexity.

2. FORMULATION

Most of the extant finite volume methods avoid overlapping control volumes by defining and using intermediate points. The values of the variables on these intermediate points are often obtained by line interpolation from the neighbours. On non-orthogonal grids it may be preferable to use higher-order finite-element-type shape functions for interpolation. To do this, it is convenient to avoid intermediate points and, by accepting overlapping control volumes, to use the neighbouring points directly in the computations.

The solution domain is discretized into a structured non-orthogonal grid as shown in Figure 1(a). This can be done by any grid generation package for boundary-fitted systems. A typical control volume is shown by the shaded area in the figure and also in Figure 1(b). This choice of control volume uses the grid point co-ordinates directly, without computing any intermediate points, to form control volumes. It can be seen that each interior grid point has a control volume associated with it, of which it is the central node. Hence we can refer to these control volumes by the index of this central node, e.g. the control volume for (i, j) is shown in Figure 1(b). It can be seen that adjacent control volumes will overlap to some extent. A collocated or non-staggered arrangement is used for the dependent variables.

2.1. Governing equations for a control volume

Now we apply the conservation laws to the typical control volume to get the algebraic discrete equations. The conservation form of the two-dimensional steady state convection–diffusion equation for a scalar ϕ is

$$\nabla \cdot (\rho \tilde{U} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi, \tag{1}$$

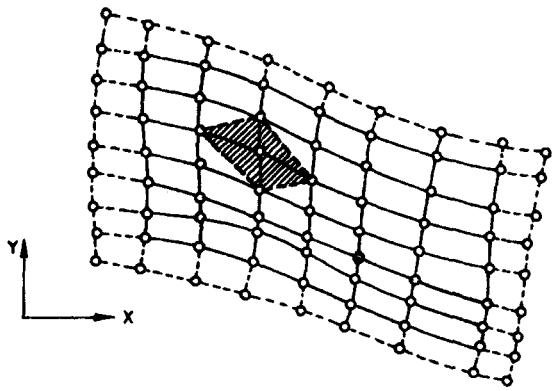

Figure 1(a)

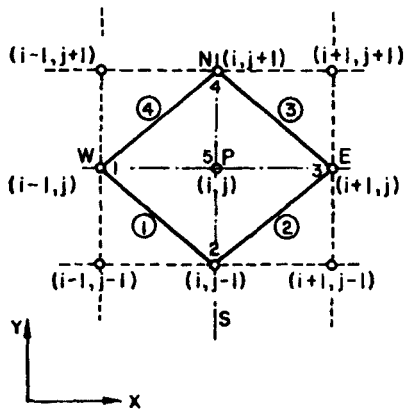

Figure 1(b)

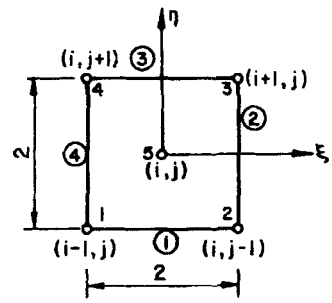

Figure 1(c)

where ρ is the density, $\tilde{U}$ is the velocity vector having components u and v in the directions x and y respectively, Γ is the diffusion coefficient and S_ϕ is a source term. On integrating equation (1) over the control volume and applying the Gauss divergence theorem, we get

$$\oint_{cs} \phi(\rho u n_x + \rho v n_y)\mathrm{d}l = \oint_{cs} \Gamma\left(\frac{\partial \phi}{\partial x} n_x + \frac{\partial \phi}{\partial y} n_y\right)\mathrm{d}l + \int\int S_\phi \mathrm{d}A, \tag{2}$$

where $\mathrm{d}l$ is an elemental length on the boundary (cs) of the control volume and n_x and n_y are the direction cosines of the outward normal $\hat{n}$ of $\mathrm{d}l$. The area of the control volume (i, j) in Figure 1(b) is calculated using the formula

$$\begin{aligned} A_s = 0{\cdot}5[&(x_{i,j+1} - x_{i+1,j})(y_{i,j-1} - y_{i+1,j}) - (x_{i,j-1} - x_{i+1,j})(y_{i,j+1} - y_{i+1,j}) \\ &+ (x_{i,j-1} - x_{i-1,j})(y_{i,j+1} - y_{i-1,j}) - (x_{i,j+1} - x_{i-1,j})(y_{i,j-1} - y_{i-1,j})], \end{aligned}$$

where $x_{i,j+1}$, $y_{i,j+1}$, etc. are the co-ordinates of the neighbouring points shown in Figure 1(b).

The contour integration is counterclockwise. The terms in equation (2) are further approximated as follows.

2.2. Convection term

Using the midpoint rule, we approximate the convection term as

$$\oint_{cs} \phi(\rho u n_x + \rho v n_y)\mathrm{d}l = \sum_{k=1}^{4} \phi^{(k)}(\rho u^{(k)}\Delta y^{(k)} - \rho v^{(k)}\Delta x^{(k)})$$
$$= \sum_{k=1}^{4} \phi^{(k)} F^{(k)}, \tag{3}$$

where the superscript (k) refers to the edges of the control volume (shown circled in Figure 1(b)). For the edge (k), $k=1, 2, 3$, the approximation used is (assuming constant density ρ)

$$u^{(k)} = 0{\cdot}5(u_k + u_{k+1}), \qquad v^{(k)} = 0{\cdot}5(v_k + v_{k+1}), \qquad \Delta y^{(k)} = y_{k+1} - y_k, \qquad \Delta x^{(k)} = x_{k+1} - x_k,$$

where the subscript k refers to the local grid point number (shown in Figure 1(b)). For $k=4$, u_{k+1}, v_{k+1}, etc. are replaced by u_1, v_1, etc. respectively in the above equations. The outward mass flux through the edge k is

$$F^{(k)} = (\rho u \Delta y - \rho v \Delta x)^{(k)}. \tag{4}$$

To incorporate upwinding, $\phi^{(k)}$ in (3) is approximated at the midpoint of control surface k by interpolation within the appropriate control volume depending on the flow direction across that surface. For example, if $F^{(1)}$ is negative (i.e. flow is entering the control volume across face 1), then $\phi^{(1)}$ is approximated by interpolation within the control volume for node $(i-1, j-1)$. That is, the values at the grid points constituting control volume $(i-1, j-1)$, through which the flow enters control volume (i, j), are used for the interpolation of $\phi^{(1)}$ at the centre of surface 1. Otherwise, if $F^{(1)}$ is positive, the values at the grid points of control volume (i, j) are used to interpolate $\phi^{(1)}$ at surface 1. This scheme for convective modelling is obviously conservative as it always uses the upwind control volume. The method used for interpolation is based on finite-element-type shape functions and is explained below.

2.3. Interpolation

Control volume (i, j) is mapped onto a square in (ξ, η)-space as shown in Figure 1(c), with node (i, j) at (0, 0) and the other nodes at the vertices $(\pm 1, \pm 1)$ respectively. The following shape functions are used for the interpolations:

$$N_1 = 0{\cdot}25(-\xi - \eta + \xi\eta) + 0{\cdot}125(\xi^2 + \eta^2), \qquad N_2 = 0{\cdot}25(\xi - \eta - \xi\eta) + 0{\cdot}125(\xi^2 + \eta^2),$$
$$N_3 = 0{\cdot}25(\xi + \eta + \xi\eta) + 0{\cdot}125(\xi^2 + \eta^2), \qquad N_4 = 0{\cdot}25(-\xi + \eta - \xi\eta) + 0{\cdot}125(\xi^2 + \eta^2), \tag{5}$$
$$N_5 = 1 - 0{\cdot}5(\xi^2 + \eta^2).$$

The isoparametric formulation is used and the dependent variable ϕ and the co-ordinates x and y in the control volume are represented as

$$\phi = \sum_{i=1}^{5} N_i \phi_i, \tag{6}$$

$$x = \sum_{i=1}^{5} N_i x_i, \tag{7}$$

$$y = \sum_{i=1}^{5} N_i y_i, \tag{8}$$

where the x_i and y_i are the x- and y-co-ordinates of the five grid points respectively. Equations (7) and (8) map the control volume in Figure 1(b) onto the transformed control volume in (ξ, η)-co-ordinates shown in Figure 1(c).

For the purposes of upwinding, $\phi^{(k)}$ is found by using (6) to interpolate the value at the midpoint of face k in the transformed control volume. For example, for face 1 we compute $N_1, N_2, \ldots, N_5$ at $\xi = 0$, $\eta = -1$ and then use (6).

The derivatives of the dependent variable ϕ are defined as

$$\frac{\partial \phi}{\partial x} = \sum_{i=1}^{5} \frac{\partial N_i}{\partial x} \phi_i, \tag{9}$$

$$\frac{\partial \phi}{\partial y} = \sum_{i=1}^{5} \frac{\partial N_i}{\partial y} \phi_i, \tag{10}$$

where the derivatives of the shape functions and the determinant of the Jacobian respectively are given by

$$\begin{Bmatrix} \partial N_i/\partial x \\ \partial N_i/\partial y \end{Bmatrix} = \frac{1}{|J|} \begin{bmatrix} \partial y/\partial \eta & -\partial y/\partial \xi \\ -\partial x/\partial \eta & \partial x/\partial \xi \end{bmatrix} \begin{Bmatrix} \partial N_i/\partial \xi \\ \partial N_i/\partial \eta \end{Bmatrix}, \tag{11}$$

$$|J| \equiv \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi}. \tag{12}$$

The derivatives with respect to ξ and η can be computed using equations (5)–(8).

2.4. *Diffusion term*

The control volume is in each case the control volume for (i, j). This term is also approximated using the midpoint rule. The discretized equation is represented as

$$\oint_{cs} \Gamma \left(\frac{\partial \phi}{\partial x} n_x + \frac{\partial \phi}{\partial y} n_y \right) \mathrm{d}l = \sum_{k=1}^{4} \sum_{i=1}^{5} \Gamma \left(\frac{\partial N_i}{\partial x}\bigg|_{\text{mid}} \Delta y^{(k)} - \frac{\partial N_i}{\partial y}\bigg|_{\text{mid}} \Delta x^{(k)} \right) \phi_i, \tag{13}$$

where $\Delta x^{(k)}$ and $\Delta y^{(k)}$ are the same as defined for convection discretization. The derivatives of the shape functions are evaluated at the midpoints in (ξ, η)-space of the control surfaces. The summation is carried out over all the surfaces of the control volume. The procedure to evaluate the derivative terms has been explained above. It can be noted that this discretization for the diffusion term is not conservative. However, we invariably found the final converged solution to satisfy closely the conservative property of the scalar. It can be shown that on a regular grid the diffusion modelling in OCV is second-order-accurate and has exactly the same error as the conventional five-point central difference discretization.

2.5. *Source terms and boundary conditions*

The source term S can be represented in the general form

$$S_\phi = S_c + S_p \phi, \tag{14}$$

where S_c and S_p are stored at the cell centre and are assumed to prevail over the entire control volume.

In the proposed scheme, fictitious boundaries are created along the domain boundary as shown in Figure 1(a) by broken lines. The values of the scalar ϕ at these fictitious points are specified using quadratic extrapolation. These additional grid points are needed for the upwinding and diffusion term calculations at the grid points on the boundary (for Neumann boundary conditions) or next to the boundary (for Dirichlet boundary conditions).

2.6. Solution procedure

Finally, with the above formulation the discretized equation for the convection–diffusion equation (1) can be written as

$$a_p\phi_p = \sum_{nb} a_{nb}\phi_{nb} + b, \tag{15}$$

where ϕ_p is the (unknown) scalar value at the central node and ϕ_{nb} are the (unknown) values at the neighbours (including those for neighbouring control volumes introduced by upwinding). The coefficients a_p and a_{nb} are given in the Appendix.

The coefficient matrix may lose its diagonal dominance in highly convective flows and the iterative scheme may thus become unstable. To facilitate iterative convergence, the terms with negative coefficients are removed from the summation in (15) and are included in b. The Gauss–Seidel iteration technique is used to solve the discretized equation. Except for very mild overshoots and undershoots for problems with step input and at high Peclet number, no major difficulties were encountered in solving the variety of test problems reported in this paper.

3. RESULTS

A number of steady state convection–diffusion problems are now solved by the overlapping control volume (OCV) method and the results are compared with those of other schemes. These test problems and the results obtained are discussed in this section. In all cases the governing equation is (1) with $\rho = 1$. The computed solutions are compared with the exact solution, with the RMS error

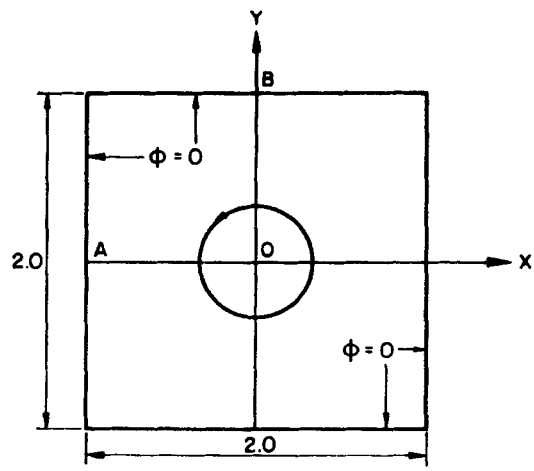

Figure 2(a). Schematic diagram of test problem 1

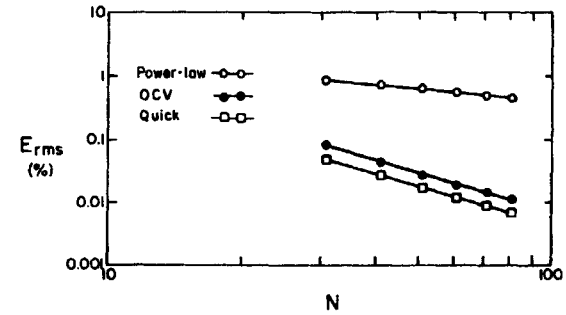

Figure 2(b). E_{rms} versus N for test problem 1

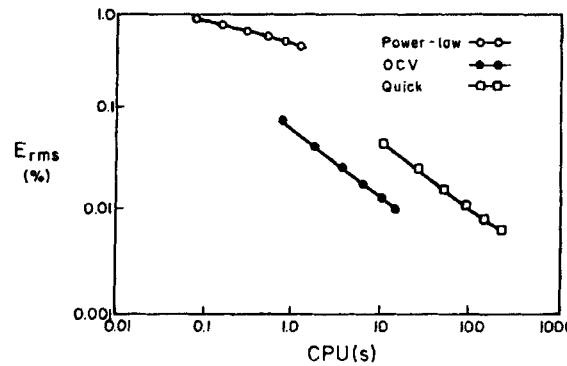

Figure 2(c). E_{rms} versus CPU time for test problem 1

defined as

$$E_{rms} = 100\left(\frac{\sum^{M} |\phi_{exact} - \phi_{computed}|^2}{M}\right)^{1/2},$$

where the summation is over all M interior points of the domain.

When other schemes are used for comparison, in all cases the method of iteration is the same: Gauss–Seidel using a 'positive-coefficient-only' matrix, with the coefficients in equation (15) varying with scheme used. All computations were done on a DEC-3000 computer.

3.1. Test Problem 1

This test problem is used to show the efficacy of the convection modelling of the OCV method. Comparisons with the power-law scheme and QUICK[6] are presented for a range of grid levels. The computational domain and boundary conditions are shown in Figure 2(a). The velocity components are defined as $u = -y$ and $v = x$. If the problem is purely convective ($\Gamma = 0$), then any scalar profile specified along OA in Figure 2(a) should be swept unchanged along the streamlines and reproduced at OB (after going through a 270° turn).

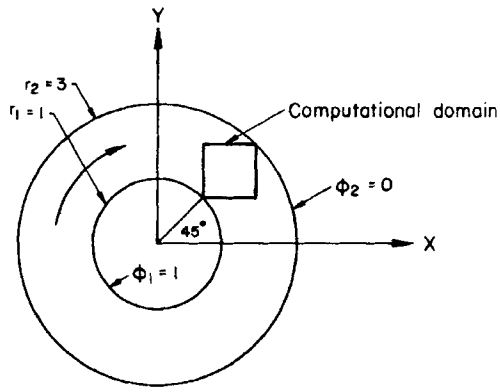

Figure 3(a). Schematic diagram of test problem 2

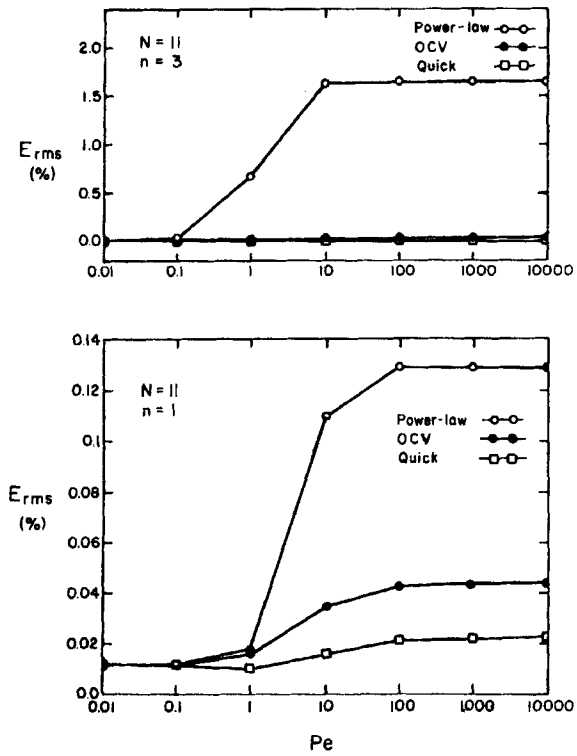

Figure 3(b). E_{rms} versus Pe for test problem 2

The computational domain is discretized using $N \times N$ uniform grids with $N = 31$, 41, 51, 61, 71 and 81. The diffusion coefficient $\Gamma = 10^{-6}$, $\rho = 1$ and $L = 2$ are used in the computations. The smooth profile specified along OA is the Gaussian distribution $\phi = e^{2|x|} \sin^2(\pi x)$. The percentage RMS error for the points on OB (*not* for the entire domain) for power-law, QUICK and OCV at various grid levels are shown in Figure 2(b) on a log–log scale. It can be seen that the percentage RMS error for the OCV scheme is much less than that for the power-law scheme. However, the performance of

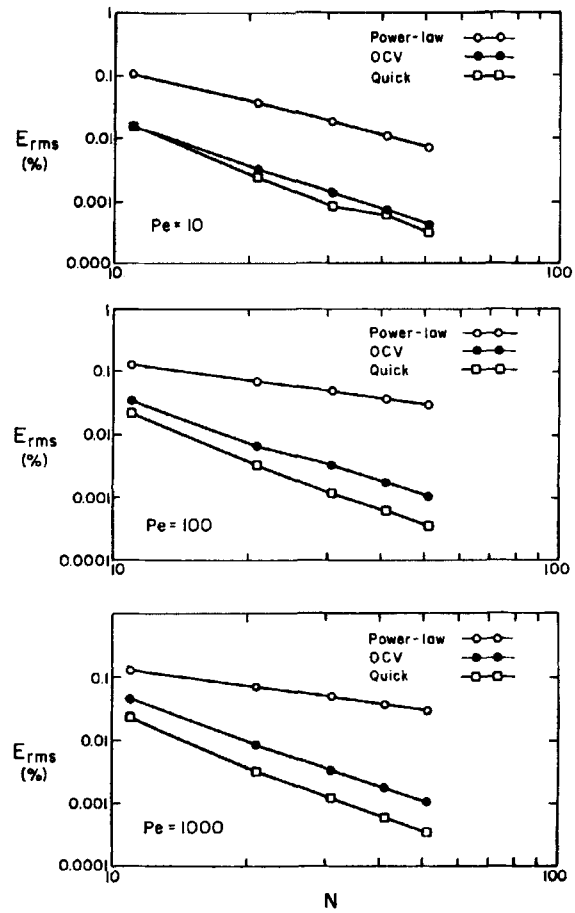

Figure 3(c). E_{rms} versus N for test problem 2 ($n = 1$)

QUICK is best in this case. To estimate the order of accuracy for a smoothly varying solution, the error level ε can be assumed to vary in proportion to Δx^m, where m is the order of accuracy. The exponents for the power-law, QUICK and OCV schemes are found to be 0·606, 2·004 and 2·054 respectively. This shows that the convection modelling used in the OCV scheme is, like QUICK, second-order-accurate. The error of OCV is slightly larger than that of QUICK.

The CPU times taken by the various schemes obviously play some part in our estimation of them. We compare E_{rms} versus CPU time in Figure 2(c) for the same cases. We find that OCV, because it converges faster than QUICK, gives better accuracy for a given CPU time. The power-law scheme performs relatively poorly in comparison with the other two schemes.

3.2. Test problem 2

This problem, used by Runchal,[3] is shown in Figure 3(a). The inner and outer surfaces of a cylindrical annulus are maintained at constant ϕ_1 and ϕ_2 (assumed as unity and zero respectively). The Cartesian velocity components are represented as $u = 2r^{n-1}y$ and $v = -2r^{n-1}x$. Two values of

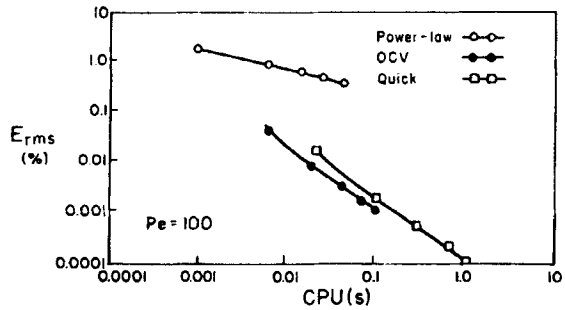

Figure 3(d). E_{rms} versus CPU time for test problem 2 ($n = 3$)

the constant n are considered: $n = 1$, representing solid body rotation with angular velocity $\omega = 2$, and $n = 3$, for which the angular velocity is $\omega = 2r^2$.

The problem is one-dimensional in cylindrical co-ordinates, hence its exact solution can be obtained easily. However, in the Cartesian co-ordinate system this problem becomes two-dimensional. The exact solution for all Peclet numbers and values of n is given by

$$\phi = 1 - \frac{\ln(r/r_1)}{\ln(r_2/r_1)},$$

where r_1 and r_2 are the inner and outer radii of the annulus respectively.

The computational domain is the square shown in the figure and is divided into $N \times N$ grid points. In this problem we take $r_1 = 1$ and $r_2 = 3$. The exact solution is used to specify the Dirichlet boundary conditions along the boundaries of the square domain.

We compare the results of the OCV scheme with those of the power-law and QUICK schemes for this problem for a range of Peclet numbers from 0·01 to 10,000, where $Pe \equiv \omega r_1^2/\Gamma$. It should be noted, however, that this is only the nominal Peclet number (based on the inner radius of the annulus). The true Peclet number $V\sqrt{2}/\Gamma$ (the side of the square is $\sqrt{2}$) is a variable and its value at the midpoint of the domain is $2\sqrt{2}$ and $8\sqrt{2}$ times the nominal value for $n = 1$ and 3 respectively.

Figure 3(b) shows the percentage RMS error for $n = 1$ and 3. The OCV scheme shows better results than the power-law scheme, but QUICK is best for this case. For low Peclet numbers the schemes show nearly the same error, since the diffusion modelling in power-law and QUICK is the same central difference scheme and that of OCV is equivalent.

In Figure 3(b) for $n = 3$ the percentage RMS error for the power-law scheme increases very rapidly with the Peclet number, whereas E_{rms} for QUICK and OCV remains almost same as that for $n = 1$. Figure 3(c) can be used to estimate the order of accuracy of the schemes for a range of Peclet numbers, $Pe = 10$, 100 and 1000. The value of n used to specify the velocity fields is $n = 1$. Five different grid levels, $N = 11$, 21, 31, 41 and 51, are used. It can be seen that the slope of the E_{rms} line for OCV is nearly same as that for QUICK and the accuracy is much better than that of the power-law scheme. The exponents m, the same as defined in test problem 1, for the power-law, OCV and QUICK are found to be 1·65, 2·25 and 2·40 for $Pe = 10$, 0·88, 2·28 and 2·55 for $Pe = 100$ and 0·88, 2·27 and 2·56 for $Pe = 1000$ respectively. Once again OCV demonstrates that it is a second-order method. QUICK is somewhat better than OCV at all Peclet numbers for the same grid size.

However, we compare E_{rms} versus CPU in Figure 3(d) for $Pe = 100$, $n = 3$, which is typical, and once again find OCV slightly better than QUICK for the same CPU time.

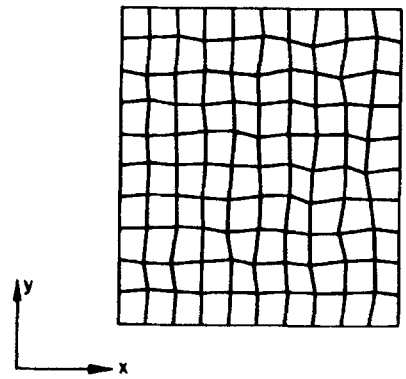

Figure 4(a). Distorted (10 per cent) grid for test problem 3

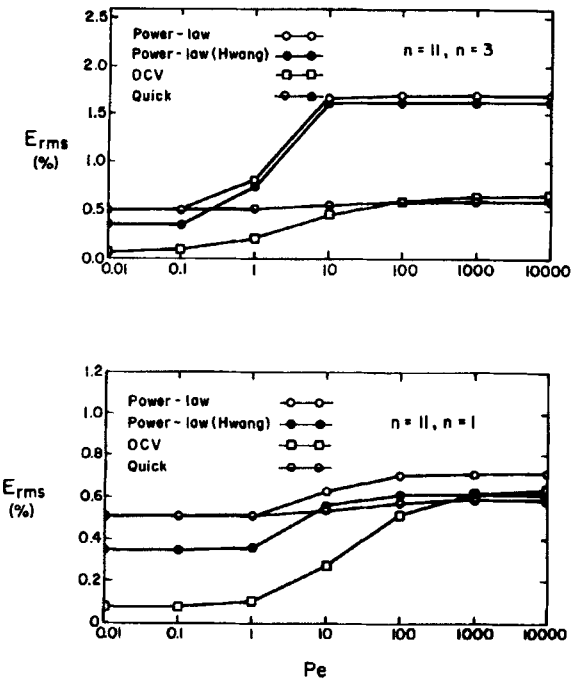

Figure 4(b). E_{rms} versus Pe for test problem 3

3.3. *Test Problem 3*

This problem is designed to test the applicability of the proposed scheme to arbitrary geometries. The details and exact solution of this problem are the same as those of test problem 2. The only difference is that the interior grid points are randomly perturbed from their original uniform grid positions to a maximum extent of 10 per cent of the average grid distance. The domain and grid for $N = 11$ are shown in Figure 4(a).

We now consider four schemes for this problem. The power-law and QUICK schemes (both with the conventional central difference diffusion modelling[11]) are used for the transformed equation generated by mapping the grid shown in Figure 4(a) onto a rectangular grid. This is simply to map the

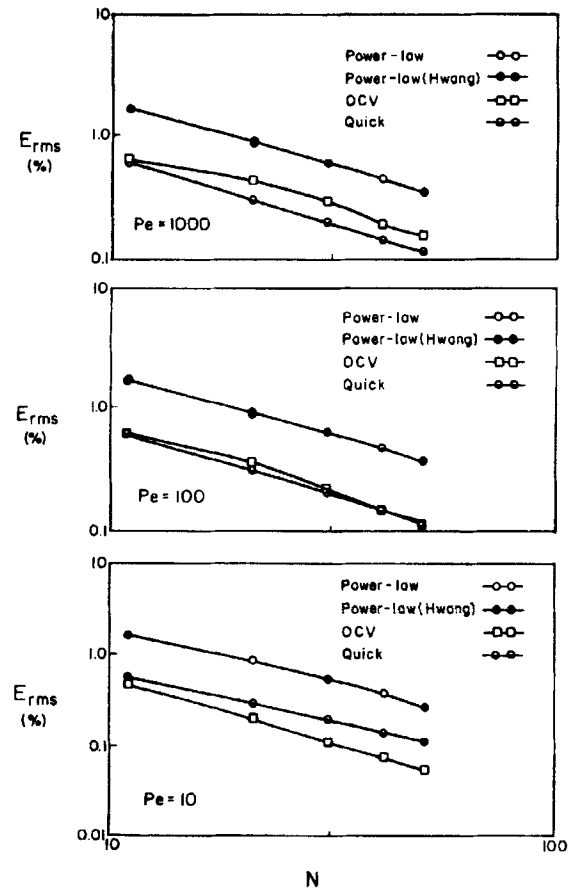

Figure 4(c). E_{rms} versus N for test problem 3 ($n=3$)

points $x(i, j)$ and $y(i, j)$ onto $i\Delta\xi$ and $j\Delta\eta$ on the (ξ, η)-plane. The other two methods we consider are those of Hwang[16] and OCV which apply directly on the grid in the physical plane. The diffusion modelling presented by Hwang[16] (taken from Reference 22) is used here along with the power-law method for convection.

Figure 4(b) shows a comparison of the errors obtained on the $N=11$ grid for $n=1$ and 3. It can be seen that at Peclet numbers at least up to 100 the OCV scheme gives the best results. This result is repeated for values of $N=21$, 31, 41 and 51 and also for grid distortions of 3 per cent and 5 per cent.

Figure 4(b) also shows (in the limit of $Pe \to 0$) that the OCV diffusion modelling is superior to that of Hwang,[16] which in turn is superior to the (transformed) central difference diffusion modelling. It is clear that the Laplacian term introduces an error into the schemes for the transformed equations which has a deleterious effect on their accuracy except at high Peclet numbers (remember that the actual Peclet number of the problem is higher than the nominal values used in the figures). This conjecture was confirmed by studying the case of pure diffusion on the distorted grid. It was seen that the error in the schemes relying on transformation increases very rapidly with the degree of distortion in the grid. Figure 4(c) shows E_{rms} versus N for the four schemes for $Pe=10$, 100 and 1000 for $n=3$. In all these cases the power-law schemes give approximately the same results. It can be seen that for a Peclet number of 10 the accuracy and the improvement with N are best for OCV. For $Pe=100$, OCV

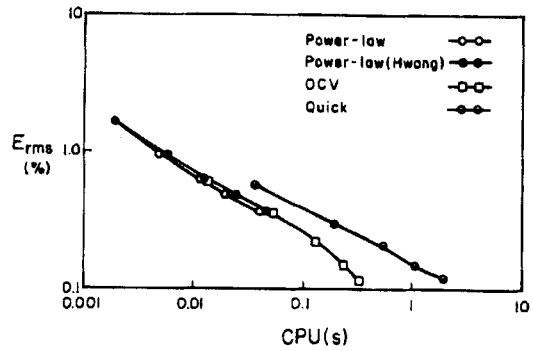

Figure 4(d). E_{rms} versus CPU time for test problem 3 ($n = 3$)

Table I. CPU time (seconds) and number of iterations for convergence (in parentheses) for $Pe = 10$, $n = 3$

Grid	Power-law (Hwang)	OCV	QUICK	Power- law
11 × 11	0·00195 (5)	0·00585 (24)	0·03025 (34)	0·00195 (5)
21 × 21	0·00976 (11)	0·02342 (22)	0·18446 (41)	0·00976 (11)
31 × 31	0·03025 (21)	0·04684 (18)	0·50849 (50)	0·03025 (21)
41 × 41	0·07515 (35)	0·09174 (19)	1·03356 (66)	0·08003 (35)
51 × 51	0·16299 (47)	0·24490 (47)	2·43122 (87)	0·18153 (46)

and QUICK are comparable. Only above that Peclet number does QUICK (in the transformed domain) perform better. Therefore on distorted grids it may be supposed that OCV has a better performance in the range of practical grid Peclet numbers.

The variation in E_{rms} versus CPU time is shown in Figure 4(d) for $Pe = 100$, $n = 3$, a typical case. The OCV scheme does better than QUICK. The power-law schemes on both transformed and physical domain also do well, presumably because their superior convergence properties outweigh their first-order accuracy.

The CPU times (in seconds) are given in Table I for the various schemes at different N. The quantities in parentheses are the iterations needed to reach convergence (residual of 10^{-7} with double- precision arithmetic). QUICK is seen to be expensive and to become increasingly so with grid refinement, not only taking more CPU time per iteration but also requiring a greater number of iterations. It must be mentioned that the relatively poor performance of QUICK in the CPU time computations can be attributed directly to the poor convergence properties of the coefficient matrix of the scheme used in equation (15). This matrix will not be used in explicit time stepping and will be different for implicit time stepping. Therefore the performance of QUICK here is uniquely for the steady state solution and extrapolating to unsteady solutions would be injudicious.

3.4. Test Problem 4

This model problem, first proposed by Raithby,[4] is widely used to test the cross-stream numerical diffusion of difference schemes.

The problem is shown in Figure 5(a). The flow is assumed uniform and passing through the square domain, making an angle such that the streamline through $(0, y_c)$ passes through the centre of the

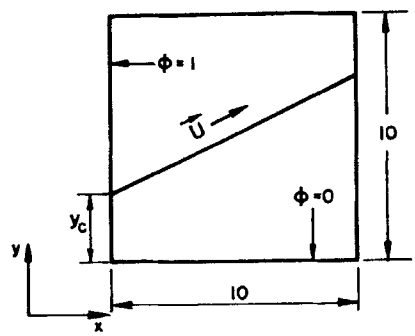

Figure 5(a). Schematic diagram of test problem 4

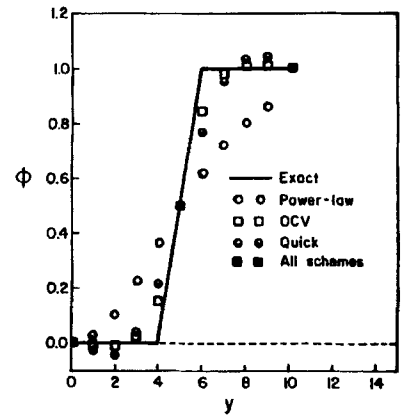

Figure 5(b). Results of test problem 4 for $y_c = 0$

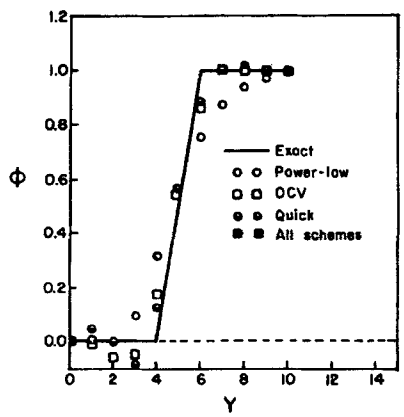

Figure 5(c). Results of test problem 4 for $y_c = 3$

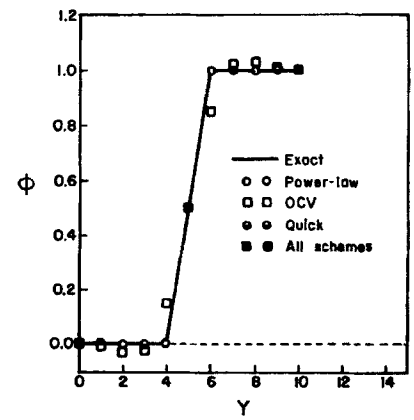

Figure 5(d). Results of test problem 4 for $y_c = 5$

square. The scalar field at the left (inflow) boundary has an abrupt step change, with $\phi = 1$ above y_c and $\phi = 0$ below. The Peclet number is taken as infinity and thus the scalar property is transported by convection only. The boundary conditions are shown on the figure. The value of ϕ is assumed to be 0·5 at the point of the step change, $y = y_c$, $x = 0$. The exact solution for this problem is the convection of the step change in the flow direction without any diffusion. Thus $\phi = 1$ above the slanted line shown and $\phi = 0$ below it.

A regular 11 × 11 grid is used to discretize the solution domain. The results at the midplane ($x = 0{\cdot}5$) are shown in Figures 5(b)–5(d) for $\Gamma = 10^{-6}$ and $y_c = 0$, 3 and 5 respectively. The results of the OCV scheme are compared with those of the power-law and QUICK schemes. Figure 5(b) shows that OCV gives minimum smearing errors compared with all other schemes at $y_c = 0$. This is the worst case for all other schemes. For $y_c = 3$ it can be seen from Figure 5(c) that the levels of overshoots and undershoots are approximately of the same order for OCV and QUICK. The power-law scheme again gives maximum crosswind diffusion, but there is no overshoot or undershoot. At

$y_c = 5$, when the flow is aligned with the grid lines, it can be observed from Figure 5(d) that all other schemes except OCV give the exact solution. The OCV scheme introduces small overshoots and undershoots in this case.

Thus OCV introduces cross-stream numerical diffusion, except for the case where the flow is aligned with the grid lines, at levels below that of the power-law scheme and comparable with QUICK. However, like QUICK and other higher-order schemes, OCV allows overshoots and undershoots.

4. CONCLUSIONS

The following observations can be made on the basis of the results described above. The proposed OCV scheme performs consistently well in all the test cases.

The OCV treatment of diffusion seems to be quite effective even on distorted meshes. On uniform meshes it is second-order-accurate. For convection- dominated flows the OCV treatment of the convection term, like that of the QUICK scheme, is second-order-accurate on a uniform mesh. Although the OCV scheme does not perform as well as QUICK for the same grid size, it does bettter for the same CPU time.

When both convection and diffusion processes are significant, the results obtained using OCV are quite encouraging on non-orthogonal meshes. It does much better than the power-law scheme. Its accuracy for the non-orthogonal test problem attempted is comparable with the QUICK scheme used with central difference modelling, while it is computationally less expensive. An interesting sidelight to these results is offered by Leonard and Drummond[23] demonstrating the superiority of QUICK to 'hybrid', 'power-law' and other exponential schemes.

Two other issues perhaps need to be mentioned: the extension of the scheme to three dimensions and the applicability of flux limiter procedures. It is possible that the latter may be applied to this scheme without major difficulty as they have been to other finite volume schemes.[24] However, extension of the scheme to three dimensions, it seems, cannot be a direct generalization and needs to be done by analogy, i.e. by developing a 3D scheme that uses similar principles. This is decidedly a disadvantage of this scheme.

ACKNOWLEDGEMENT

We are indebted to the anonymous reviewers for suggestions which improved this paper.

APPENDIX

Diffusion term

For face 1, at the midpoint (i.e. $\xi = 0$, $\eta = -1$) of the control surface as shown in Figures 1(b) and 1(c), the diffusion term is

$$\mathrm{DIFF\_1}(k) = \Gamma\left(\left.\frac{\partial N_k}{\partial x}\right|_{(0,-1)}\Delta y^{(1)} - \left.\frac{\partial N_k}{\partial y}\right|_{(0,-1)}\Delta x^{(1)}\right),$$

where $k = 1$–5 are the local node numbers in the counterclockwise sense as shown in Figure 1(c). The other terms on the right-hand side of the above expression have already been defined in Section 2. Similarly, for control volume faces 2, 3 and 4 the diffusion coefficients are respectively

$$\text{DIFF\_2}(k) = \Gamma\left(\left.\frac{\partial N_k}{\partial x}\right|_{(1,0)}\Delta y^{(2)} - \left.\frac{\partial N_k}{\partial y}\right|_{(1,0)}\Delta x^{(2)}\right),$$

$$\text{DIFF\_3}(k) = \Gamma\left(\left.\frac{\partial N_k}{\partial x}\right|_{(0,1)}\Delta y^{(3)} - \left.\frac{\partial N_k}{\partial y}\right|_{(0,1)}\Delta x^{(3)}\right),$$

$$\text{DIFF\_4}(k) = \Gamma\left(\left.\frac{\partial N_k}{\partial x}\right|_{(-1,0)}\Delta y^{(4)} - \left.\frac{\partial N_k}{\partial y}\right|_{(-1,0)}\Delta x^{(4)}\right),$$

where $k = 1$–5. If we define local nodes 1, 2, 3 and 4 as the west, south, east and north neighbours of node P respectively as shown in Figure 1(b), the final expressions for the diffusion coefficients for a control volume can be written as

$$\begin{aligned}
D_{\mathrm{N}} &= \text{DIFF_1}(4) + \text{DIFF_2}(4) + \text{DIFF_3}(4) + \text{DIFF_4}(4),\\
D_{\mathrm{S}} &= \text{DIFF_1}(2) + \text{DIFF_2}(2) + \text{DIFF_3}(2) + \text{DIFF_4}(2),\\
D_{\mathrm{E}} &= \text{DIFF_1}(3) + \text{DIFF_2}(3) + \text{DIFF_3}(3) + \text{DIFF_4}(3),\\
D_{\mathrm{W}} &= \text{DIFF_1}(1) + \text{DIFF_2}(1) + \text{DIFF_3}(1) + \text{DIFF_4}(1),\\
D_{\mathrm{P}} &= \text{DIFF_1}(5) + \text{DIFF_2}(5) + \text{DIFF_3}(5) + \text{DIFF_4}(5).
\end{aligned}$$

Convection term

For face 1, again at the midpoint, the convection term is approximated as

$$\begin{aligned}
\text{CONV_1} &= (\phi^{(1)}F^{(1)})|_{\mathrm{mid}}\\
&= (N_k\phi_k)^{(1)}F^{(1)}|_{\mathrm{mid}}, \quad k = 1\text{-}5.
\end{aligned}$$

Here the midpoint is ($\xi = 0$, $\eta = -1$) for positive $F^{(1)}$ and ($\xi = 0$, $\eta = 1$) for negative $F^{(1)}$. Combining both possibilities in a single expression, we get

$$\begin{aligned}
\text{CONV_1} = {} & \max(F^{(1)}, 0)[C_1(1)\phi_{i-1,j} + C_1(2)\phi_{i,j-1} + C_1(3)\phi_{i+1,j} + C_1(4)\phi_{i,j+1}\\
& + C_1(5)\phi_{i,j}] - \max(-F^{(1)}, 0)[C_11(1)\phi_{i-2,j-1} + C_11(2)\phi_{i-1,j-2}\\
& + C_11(3)\phi_{i,j-1} + C_11(4)\phi_{i-1,j} + C_11(5)\phi_{i-1,j-1}].
\end{aligned}$$

Similar expressions for the other surfaces (CONV_2, CONV_3 and CONV_4) of a control volume can be obtained. Now, rearranging the terms and writing expressions for each node of a control volume,

we get

$$C_W = [C\_1(1)\max(F^{(1)}, 0) + C\_2(1)\max(F^{(2)}, 0) + C\_3(1)\max(F^{(3)}, 0) + C\_4(1)\max(F^{(4)}, 0)] - [C\_11(4)\max(-F^{(1)}, 0) + C\_44(2)\max(-F^{(4)}, 0)],$$

$$C_E = [C\_1(3)\max(F^{(1)}, 0) + C\_2(3)\max(F^{(2)}, 0) + C\_3(3)\max(F^{(3)}, 0) + C\_4(3)\max(F^{(4)}, 0)] - [C\_22(4)\max(-F^{(2)}, 0) + C\_33(2)\max(-F^{(3)}, 0)],$$

$$C_N = [C\_1(4)\max(F^{(1)}, 0) + C\_2(4)\max(F^{(2)}, 0) + C\_3(4)\max(F^{(3)}, 0) + C\_4(4)\max(F^{(4)}, 0)] - [C\_33(1)\max(-F^{(3)}, 0) + C\_44(3)\max(-F^{(4)}, 0)],$$

$$C_S = [C\_1(2)\max(F^{(1)}, 0) + C\_2(2)\max(F^{(2)}, 0) + C\_3(2)\max(F^{(3)}, 0) + C\_4(2)\max(F^{(4)}, 0)] - [C\_11(3)\max(-F^{(1)}, 0) + C\_22(1)\max(-F^{(2)}, 0)],$$

$$C_P = [C\_1(5)\max(F^{(1)}, 0) + C\_2(5)\max(F^{(2)}, 0) + C\_3(5)\max(F^{(3)}, 0) + C\_4(5)\max(F^{(4)}, 0)],$$

where the subscripts N, S, E and W denotes the neighbouring nodes as defined earlier and the remaining terms of CONV_1, CONV_2, CONV_3 and CONV_4 can be included in the term b of equation (15). Finally, the coefficients can be represented as

$$a_P = C_P - D_P + \max(C_N, 0) + \max(C_S, 0) + \max(C_E, 0) + \max(C_W, 0),$$
$$a_N = D_N + \max(-C_N, 0), \qquad a_S = D_S + \max(-C_S, 0),$$
$$a_E = D_E + \max(-C_E, 0), \qquad a_W = D_W + \max(-C_W, 0),$$
$$b = SS - \max(C_N, 0)(\phi_{i,j+1} - \phi_{i,j}) - \max(C_S, 0)(\phi_{i,j-1} - \phi_{i,j}) - \max(C_E, 0)(\phi_{i+1,j} - \phi_{i,j}) - \max(C_W, 0)(\phi_{i-1,j} - \phi_{i,j}),$$

where SS consists of source terms as well as any other terms which cannot be included in the other coefficients defined above.

REFERENCES

1. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New York, 1980.
2. D. B. Spalding, 'A novel finite-difference formulation for differential expressions involving both first and second derivatives', *Int. j. numer. methods eng.*, **4**, 551–559 (1972).
3. A. K. Runchal, 'Convergence and accuracy of three finite differences schemes for a two-dimensional conduction and convection problem', *Int. j. numer. methods eng.*, **4**, 541–550 (1972).
4. G. D. Raithby, 'A critical evaluation of upstream differencing applied to problems involving fluid flow', *Comput. Methods Appl. Mech. Eng.*, **9**, 75–103 (1976).
5. G. D. Raithby, 'Skew upstream differencing schemes for problems involving fluid flow', *Comput. Methods Appl. Mech. Eng.*, **9**, 153–164 (1976).
6. B. P. Leonard, 'A stable and accurate convective modelling procedure based on quadratic upstream interpolation', *Comput. Methods Appl. Mech. Eng.*, **19**, 59–98 (1979).
7. G. de Vahl Davis and G. D. Mallinson, 'An evaluation of upwind and central difference approximations by study of recirculating flows', *Comput. Fluids*, **4**, 29–43 (1976).
8. T. Han, J. A. C. Humphrey and B. E. Launder, 'A comparison of hybrid and quadratic upstream differencing in high Reynolds number elliptic flows', *Comput. Methods Appl. Mech. Eng.*, **29**, 81–95 (1981).
9. A. Pollard and L. W. A. Siu, 'The calculations of some laminar flows using various discretization schemes', *Comput. Methods Appl. Mech. Eng.*, **35**, 293–313 (1982).
10. W. Shyy, 'Determination of relaxation factors for high cell Peclet number flow situation', *Comput. Methods Appl. Mech. Eng.*, **43**, 221–230 (1984).

11. C. M. Rhie and W. L. Chow, 'Numerical study of the turbulent flow past an airfoil with trailing edge separation', *AIAA J.*, **21**, 1525–1532 (1983).
12. C. F. Hsu, 'A curvilinear coordinate method for momentum heat and mass transfer in domains of irregular geometry', *Ph.D. Thesis*, University of Minnesota, 1981.
13. C. M. Rhie, 'A numerical study of flow past an isolated airfoil with separation', *Ph.D. Thesis*, University of Illinois at Urbana-Champaign, 1981.
14. M. Peric, 'A finite volume method for prediction of three dimensional fluid flow in complex ducts', *Ph.D. Thesis*, Univ of London, 1985.
15. A. Mukhopadhyay, T. Sundararajan and G. Biswas, 'An explicit transient algorithm for predicting incompressible flows in arbitrary geometry', *Int. j. numer. methods fluids*, **17**, 975–993 (1993).
16. Y.-H. Hwang, 'Arbitrary domain velocity analyses for the incompressible Navier–Stokes equations', *J. Comput. Phys.*, **110**, 134–139 (1994).
17. B. R. Baliga and S. V. Patankar, 'A new finite element formulation for convection–diffusion problems', *Numer. Heat Transfer*, **3**, 393–410 (1980).
18. B. LeDain-Muir and B. R. Baliga, 'Solution of three-dimensional convection–diffusion problems using tetrahedral elements and flow-oriented upwind interpolation functions', *Numer. Heat Transfer*, **9**, 253–276 (1986).
19. S. Ramadhyani and S. V. Patankar, 'Solution of the convection diffusion equation by a finite-element method using quadrilateral elements', *Numer. Heat Transfer*, **8**, 595–612 (1985).
20. G. E. Schneider and M. J. Raw, 'A skewed positive influence coefficient upwinding procedure for control volume base finite element convection–diffusion computation', *Numer. Heat Transfer*, **9**, 1–26 (1986).
21. C. Hirsch, *Numerical Computation of Internal and External Flows*, Vol. I, Wiley, New York, 1988.
22. A. A. Amsden, J. D. Ramshaw, P. J. O'Rouke and J. K. Dukowicz, 'KIVA: a computer program for two- and three-dimensional fluid flows with chemical reactions and fuel sprays', *LA-10245-MS*, 1985 (unpublished).
23. B. P. Leonard and J. E. Drummond, 'Why you should not use hybrid, power-law or related exponential schemes for convection-modelling—there are much better alternatives', *Int. j. numer. methods fluids*, **20**, 421–442 (1995).
24. S. Majumdar, W. Rodi and J. Zhu, 'Three-dimensional finite-volume method for incompressible flows with complex boundaries', *Trans. ASME*, **114**, 496–503 (1992).