INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 19, 135–152 (1994)

CONTROL VOLUME FINITE ELEMENT SOLUTION OF CONFINED TURBULENT SWIRLING FLOWS

D. ELKAÏM

Natural Gas Technologies Center, 35–7 De Lauzon, Boucherville, Québec J4B 1E7, Canada

AND

F. MCKENTY, M. REGGIO AND R. CAMARERO

Department of Mechanical Engineering, École Polytechnique de Montréal, CP 6079 Succ. A, Montréal, Québec H3C 3A7, Canada

SUMMARY

A control volume finite element method that uses a triangular grid has been applied for solving confined turbulent swirling flows. To treat the velocity–pressure coupling, the vorticity–streamfunction formulation has been used. For turbulence effects the k–ε model has been adopted. Consistent with the use of wall functions in the near-wall regions, a boundary condition for the calculation of the vorticity at computational boundaries is proposed and used effectively. The discretized equations are obtained by making use of an exponential interpolation function. Its use has been beneficial in reducing numerical diffusion. Comparisons of the current predictions with available experimental and numerical data from the literature showed generally fair agreement.

KEY WORDS Turbulent flow Swirling flow k–ε model Control volume

1. INTRODUCTION

As an intermediate step towards the simulation of reacting flows in combustors and furnaces, we are interested in the computation of turbulent highly swirling flows. For realistic industrial applications and in view of the existing literature,[1] positive contributions can be made by addressing the following points: geometric complexity, reducing the amount of false diffusion that plagues standard finite differences codes, and adequate modelling of the physics of the flow. Benim[1] showed that the first two points can be dealt with efficiently using the finite element method coupled with a streamline upwind procedure (SUPG).[2,3] This leads to a significant improvement over existing numerical procedures, allowing geometrical details of swirl burners to be easily captured and reducing the amount of false diffusion. As far as the third point is concerned, it has been shown[1] that for an adequate physical modelling of turbulence effects in highly swirling flows the algebraic stress model (ASM) is to be preferred over the standard k–ε model.

In this work alternative ways of treating geometric complexity and false diffusion problems are investigated. The control volume finite element method (CVFEM)[4] is applied for the computation of turbulent swirling flows. This approach uses unstructured triangular grids, resulting in a mesh flexibility comparable with that of finite element methods for the handling of geometric complexity. The methodology incorporates an exponential interpolation function,

CCC 0271–2091/94/140135–18

Received 16 May 1991
Revised 12 January 1994

also introduced by Baliga and Patankar.[4] Its main characteristic is that it varies exponentially in the direction of the average velocity vector in each element and linearly in the normal direction. By definition it correctly simulates the one-sided nature of convection, and since it is developed along the direction of the local flow vector, it reduces the problem of false diffusion encountered when other methods are used.[5,6]

The CVFEM method has already shown promise for simulating non-swirling flows using the primitive variables formulation[4,7] or the vorticity–streamfunction formulation.[8,9] However, no studies have yet been reported for turbulent swirling flows. The scope of the present work is therefore to investigate the use of the CVFEM method together with the exponential interpolation function formulation for the numerical simulation of such flows. For this we chose the vorticity–streamfunction formulation, and because an ASM model has not yet been integrated into this formulation, the standard k–ε turbulence model[10] is used. The vorticity–streamfunction formulation was chosen because it is easier to implement algorithmically and is more cost-effective from a computational standpoint, but these are not the only reasons. We are mainly interested in 2D/axisymmetric computations so that the extension to 3D is not necessary for now. If this is required, it is obvious that the vorticity–streamfunction formulation becomes much less attractive than a velocity–pressure formulation. Near solid boundaries the wall function approach[10] is adopted and particular attention is paid to the numerical implementation of the boundary conditions near solid walls.

As a preliminary test the benchmark turbulent pipe flow test[11] is solved. Then the solution of turbulent swirling flow in a diffuser is compared with other numerical and experimental solutions from the literature.[1]

2. GOVERNING EQUATIONS

The phenomenon under consideration is represented by a steady axisymmetric turbulent swirling flow. The axisymmetric Reynolds-averaged equations representing the conservation of mass and momentum are written via the vorticity–streamfunction formulation (ω, ψ). The vorticity is expressed by

$$\omega = \partial v/\partial x - \partial u/\partial r, \tag{1}$$

where u and v are the velocity components in the x- and r-direction respectively.

The streamfunction ψ is expressed in such a way that the continuity equation is identically satisfied. We then have

$$r\rho u = \partial\psi/\partial r, \qquad r\rho v = -\partial\psi/\partial x, \tag{2}$$

where r is the radius for axisymmetrical configurations.

Using equations (1) and (2), the governing equations for the fluid flow problem become[12]

$$\frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial r^2} = -\rho\omega r + \frac{1}{r}\frac{\partial\psi}{\partial r}, \tag{3}$$

$$\frac{\partial}{\partial x}(r\rho u\omega) + \frac{\partial}{\partial r}(r\rho v\omega) = \frac{\partial}{\partial x}\left(r\mu_e\frac{\partial\omega}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\mu_e\frac{\partial\omega}{\partial r}\right) + r\rho v\omega - \frac{\mu_e\omega}{r} + \frac{\partial}{\partial x}(\rho v_\theta^2), \tag{4}$$

$$\frac{\partial}{\partial x}(r\rho u v_\theta) + \frac{\partial}{\partial r}(r\rho v v_\theta) = \frac{\partial}{\partial x}\left(r\mu_e\frac{\partial v_\theta}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\mu_e\frac{\partial v_\theta}{\partial r}\right) - \left(\frac{\mu_e}{r} + \rho v + \frac{\partial\mu_e}{\partial r}\right)v_\theta, \tag{5}$$

where ρ is the density and v_θ is the tangential velocity component. μ_e is the effective viscosity and is given by

$$\mu_e = \mu + \mu_t, \tag{6}$$

μ being the laminar viscosity and μ_t the turbulent viscosity, evaluated according to the k–ε model of turbulence[10] as

$$\frac{\partial}{\partial x}(r\rho uk) + \frac{\partial}{\partial r}(r\rho vk) = \frac{\partial}{\partial x}\left(r\frac{\mu_e}{\sigma_k}\frac{\partial k}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\frac{\mu_e}{\sigma_k}\frac{\partial k}{\partial r}\right) + r(G_k - \rho\varepsilon), \tag{7}$$

$$\frac{\partial}{\partial x}(r\rho u\varepsilon) + \frac{\partial}{\partial r}(r\rho v\varepsilon) = \frac{\partial}{\partial x}\left(r\frac{\mu_e}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\frac{\mu_e}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial r}\right) + \frac{r\varepsilon}{k}(c_1 G_k - c_2\rho\varepsilon). \tag{8}$$

G_k and $c_1(\varepsilon/k)G_k$ are generation terms, as opposed to the destruction terms $-\rho\varepsilon$ and $-c_2\rho(\varepsilon/k)\varepsilon$. G_k is given by

$$G_k = \mu_t\left\{2\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial r}\right)^2\right] + 2\left(\frac{v}{r}\right)^2 + \left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial x}\right)^2 + \left[r\frac{\partial}{\partial r}\left(\frac{v_\theta}{r}\right)\right]^2 + \left(\frac{\partial v_\theta}{\partial x}\right)^2\right\}. \tag{9}$$

The turbulent viscosity is related to k and ε via

$$\mu_t = c_d\rho(k^2)/\varepsilon. \tag{10}$$

This turbulence model has five constants: σ_k, σ_ε, c_1, c_2 and c_d. The following values are commonly used:[10]

σ_k	σ_ε	c_1	c_2	c_d
1·0	1·30	1·44	1·92	0·09

3. BOUNDARY CONDITIONS

In this section the boundary conditions required to solve the system of differential equations are presented. Particular attention is paid to the near-wall treatment. Generally there are four types of boundaries: inlet, outlet, symmetry axis and solid walls. For each type and for each unknown let us see what boundary conditions are to be specified.

Inlet. The velocity is known, so that the stream function and vorticity distributions are calculated from their own definitions (equations (1) and (2)). Values of k or ε are not known at the inlet, and if they are not given by experimental data, some reasonable assumptions can be made. The kinetic energy of turbulence is estimated according to a certain percentage of the square of the average inlet velocity:

$$k = \lambda\bar{u}^2, \tag{11}$$

where $\bar{u}$ is the average inlet velocity and λ is a percentage.

The dissipation is calculated according to the equation

$$\varepsilon = c_d(k^{3/2})/aD, \tag{12}$$

where D is the inlet diameter. The values $\lambda = 0.02$ and $a = 0.005$ are commonly used and may vary slightly in the literature depending on the author.

Outlet. We suppose that the flow extends over a sufficiently long domain so that it is fully developed at the exit solution. Thus for any variable ϕ the condition is

$$\partial\phi/\partial x = 0. \tag{13}$$

Symmetry axis. The streamfunction is assigned an arbitrary value because the symmetry axis is also a streamline, while the vorticity ω is equal to zero by definition and from the fact that $\partial u/\partial r = 0$ and $v = 0$ on the axis. For the other variables the radial derivative vanishes, so that

$$\partial\phi/\partial r = 0. \tag{14}$$

Solid walls. When the boundary conditions are specified right at the physical wall, all the equations must be integrated through the viscous sublayer up to the wall. This, however, is undesirable for two reasons. First, an excessive number of grid points are needed in the near-wall region in order to adequately describe the steep gradients prevailing there. Secondly, viscous effects are important in the near-wall region, so that the high-Reynolds-number turbulence model used in this work is not applicable. One alternative is to apply the so-called wall function method[10] which makes use of the availability of empirical laws that connect the wall conditions (i.e. wall shear stress) to the dependent variables just outside the viscous sublayer. In doing so, the computational procedure skips over the viscous sublayer. Therefore wall conditions are not imposed at the wall but rather at a certain distance from the wall. The actual computational domain is then smaller than the physical one. The numerical implementation of the wall function method follows the approach proposed by Benim and Zinser.[11] A description of this approach and its extension to the vorticity–streamfunction formulation follows.

We first suppose that the shear stress at the wall, τ_w, is constant up to an *a priori* set distance δ_1 from the wall. By further supposing that in the turbulent sublayer local equilibrium prevails so that the rate of k-production is equal to its destruction rate, the following near-wall values of k and ε are obtained:

$$k = \left| \frac{\tau_w/\rho}{\sqrt{c_d}} \right|, \tag{15}$$

$$\varepsilon = c_d^{3/4} k^{3/2}/\kappa\delta_1. \tag{16}$$

Of course, the quality of the boundary conditions will depend on how the wall shear stress is evaluated. Following the approach proposed by Benim and Zinser,[11] the wall shear stress is evaluated by using the velocity profile within the calculation domain. Consider for this the near-wall nodes 1 and 2 of Figure 1. Node 1 is the boundary node where the conditions are to be applied, while node 2 is in the outer flow region where the governing equations are solved. The two nodes lie in the triangle containing the normal to the wall that passes through boundary point 1. The assumption is then that the velocity at point 2 lies on the same variation curve as point 1. In this way conditions at point 2 can be used to calculate the wall shear stress. For this the velocity variation of Reichardt[13] close to the wall is used, which is defined as

$$U_P = u^* \left\{ 2{\cdot}5 \log\left(1 + \frac{0{\cdot}4\rho Yu^*}{\mu}\right) + 7{\cdot}8\left[1 - \exp\left(\frac{-\rho Yu^*}{11\mu}\right) - \frac{\rho Yu^*}{11\mu} \exp\left(\frac{-0{\cdot}33\rho Yu^*}{\mu}\right)\right]\right\}, \tag{17}$$

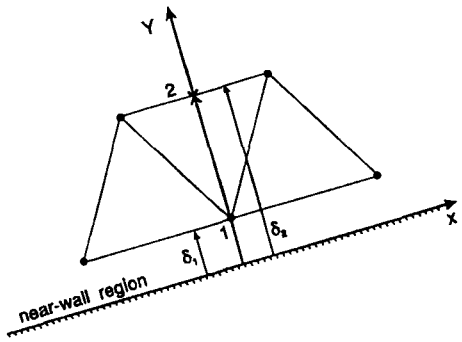

Figure 1. The near-wall region and nomenclature

where U_P is the velocity parallel to the wall and Y is the local distance from the wall to point P. For a cylindrical co-ordinate system in the presence of swirl the distance Y is the radial distance from the wall to point P, while the velocity U_P is the magnitude of the total velocity vector, which includes the contribution of the tangential velocity component v_θ. The symbol u^* denotes the so-called friction velocity and can be written in terms of the wall shear stress as

$$u^* = (\tau_w/\rho)^{0.5}. \tag{18}$$

Note that the velocity in each triangle is constant because the streamfunction is linear in each element. However, only for the purpose of calculating boundary conditions, a weighted area nodal velocity can be computed according to

$$u_p = \sum_{i=1}^{nb} u_i A_i \Big/ \sum_{i=1}^{nb} A_i, \tag{19}$$

where the sum is over all elements neighbouring point P.

Let us now address the calculation of the near-wall vorticity and streamfunction values. First we suppose that the velocity at the computational boundary is parallel to the wall so that the boundary is also a streamline. Then the value of the streamfunction on that boundary can be found by integrating the velocity profile at a section where it is known (inlet). For the vorticity the following approach is adopted. In terms of the local reference frame (X, Y) of Figure 1 the vorticity is expressed as

$$|\omega| = |\partial U/\partial Y|, \tag{20}$$

where U is the local velocity component in the X-direction and because the Y-velocity component (V) is zero along X. Furthermore, if we assume a constant shear stress in the near-wall region, then by equation (18) the friction velocity is also constant in the near-wall region. One can therefore calculate the near-wall value of the vorticity by taking the derivative of equation (17) with respect to Y.

We can now summarize the computational procedure for the near-wall values.

1. Fix the distance δ_1 of the physical wall to the computational boundary.
2. Evaluate the velocity at point 2 from the solution of the governing equations.
3. Solve the non-linear equation (equation (17)) for u^*, U_P being equal to U_2 and the distance Y equal to δ_2.

4. Use u^* and equation (17) this time to calculate the velocity U_1 at point 1 with Yequal to δ_1.
5. Evaluate the vorticity from the derivative of the velocity calculated at point 1 (equations (17) and (20)).
6. Use the values of the vorticity and the velocity as Dirichlet boundary conditions for the next iteration step.
7. Use the value of U_1 found at step 4 to compute the tangential velocity component v_θ and use it as a boundary condition for the next iteration step. v_θ is the component of U_1 in the θ-direction.

4. THE NUMERICAL SOLUTION

In order to solve the set of partial differential equations previously described, the control-volume-based finite element technique presented by Baliga and Patankar[4] has been followed. Each equation in the system can be cast in the generic form

$$\frac{\partial}{\partial x}(r\rho u\phi) + \frac{\partial}{\partial r}(r\rho v\phi) = \frac{\partial}{\partial x}\left(r\Gamma \frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\Gamma \frac{\partial \phi}{\partial r}\right) + S_\phi, \tag{21}$$

where ϕ represents the scalar which undergoes convection and is diffused through the field, Γ is the exchange coefficient and S_φ is a source or sink term. A description of the discretization method by reference to this general transport equation follows. More details may be found in References 4, 7 and 14.

4.1. Domain discretization and interpolation function

The domain of interest is first divided into three-node triangular elements. Around the computational point P a control volume is created by joining the centroids of all neighbouring elements through the midpoints of the corresponding sides (Figure 2).

Following the finite volume framework, equation (21) is integrated over the control volume.

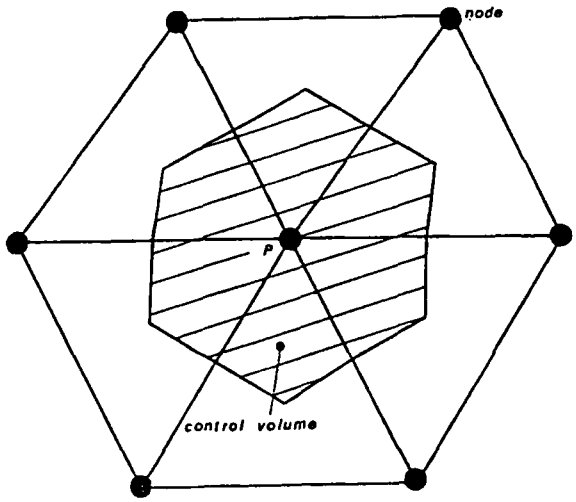

Figure 2. The polygonal control volume

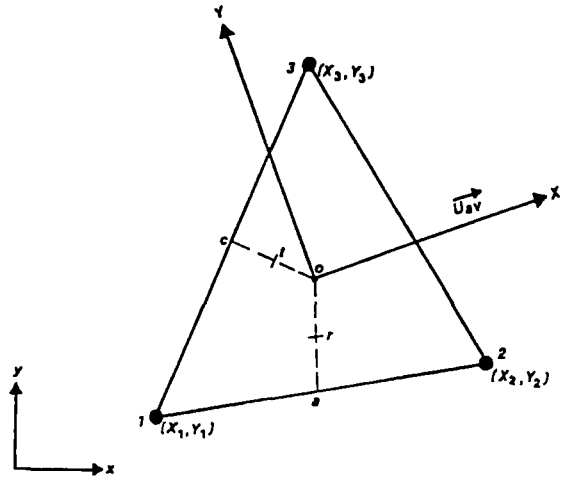

Figure 3. Triangular element with the global (x, y) system and the local (X, Y) flow-oriented co-ordinate system and related nomenclature

This procedure requires an interpolation function for ϕ. Baliga and Patankar[4] developed an approach based on the idea of using the exact solution of the one-dimensional convection–diffusion equation as the interpolation function, namely the exponential scheme proposed earlier by Patankar.[15] As an extension an interpolation function that is as close as possible to the exact solution of the two-dimensional convection–diffusion equation was introduced. With the origin located at the centroid of the element, a locally flow-oriented co-ordinate system is defined ((X, Y), Figure 3). For each triangular element the interpolation function for ϕ is given by

$$\phi = AZ + BY + C, \tag{22}$$

with

$$Z = \frac{\Gamma}{\rho U_{av}}\left[\exp\left(\frac{\rho U_{av}(X - X_{max})}{\Gamma}\right) - 1\right], \tag{23}$$

where X and Y are the local co-ordinates and $X_{max} = \max(X_1, X_2, X_3)$. U_{av} is the average local velocity expressed as

$$U_{av} = (u_{av}^2 + v_{av}^2)^{1/2}, \tag{24}$$

where

$$u_{av} = \frac{u_1 + u_2 + u_3}{3}, \qquad v_{av} = \frac{v_1 + v_2 + v_3}{3}.$$

Γ and ρ are average values of the exchange coefficient and the density respectively that prevail over the element.

The values of A, B and C are uniquely determined by the values of ϕ pertaining to the three nodes 1–2–3. For cylindrical co-ordinates there is no difference in the expression of the interpolation function (equation (22)).

The good characteristics of this shape function have been further demonstrated by Prakash and Patankar[16] and Hookey and Baliga[7,17] for fluid flow or heat transfer problems using the

primitives variable formulation or the vorticity–streamfunction formulation.[9,18] It must be pointed out that because there are no convection terms in the differential equation for the streamfunction ψ, the interpolation function for this variable is bilinear and given by $\psi = ax + br + c$. The resulting numerical scheme is then applied to solve turbulent incompressible swirling flows.

4.2. Discretized equations

The discretization of the equations is carried out by integrating equation (21) over the defined control volume. Using Simpson's rule through points *a–r–o* or *o–t–c* (Figure 3) and applying the divergence theorem, an equation of the form

$$a_p\phi_p + \sum a_{nb}\phi_{nb} = d_p \tag{25}$$

is obtained for each computational point, with a_p, and a_{nb} and d_p called the discretization coefficients;[7] the subscript *nb* refers to neighbouring points and *p* stands for the computational point *P*.

4.3. Boundary nodes and source terms

Boundary nodes. For nodes at Dirichlet boundaries $a_p = 1$ and $a_{nb} = 0$; thus d_p becomes the known value of ϕ. For Neumann boundaries an equation of the type (25) is written.

Source terms. Source terms are integrated over the control volume (CV) by considering an average value calculated at the centroid and which prevails over the element, i.e.

$$\int_{\mathrm{CV}} S \, \mathrm{d}V = \sum_{i=1}^{nb} S_i V_i/3, \tag{26}$$

where V_i is the volume (an area in 2D) of the element and the sum is taken over all neighbours of the computational point.

Whenever possible, source terms are linearized so that the coefficient a_p may also include a part due to the linearization and discretization of the source term. However, for k and ε source terms special care was taken to avoid overshoots in the solution (negative values of k or ε). The procedure consisted of first linearizing the negative part of the source term and then including it in the discretization coefficient a_p on the left-hand side of equation (25). For a reason that will become apparent later, even if the remaining positive part of the source term can also be linearized and moved over to the left-hand side, it is left on the right-hand side. The linearization of the negative part is performed in the following way.

Let S_ϕ^- denote the negative part of the source term, ϕ being k or ε. S_ϕ^- may be written as

$$S_\phi^- = -\phi f(\phi), \tag{27}$$

where $f(\varphi)$ is a function of φ. The linearization is then taken to be

$$(S_\phi^-)^{n+1} = -\phi^{n+1} f(\phi)^n, \tag{28}$$

where n stands for the value at the previous iteration.

This practice, of taking the negative part only of the source term and including it on the left-hand side of equation (25), makes the resulting matrix more diagonally dominant and the

iterative procedure converges. Also, the apparent behaviour of ϕ from one iteration to another (according to the remaining part of the source term) is that of a growing function. During the first iteration steps the values of k and ε still remain negative and after that they become admissible.

4.4. Solution of the discretized equations

For each computational point an equation of the (25) type is written. These equations are then assembled to solve for each variable the entire field implicitly. However, owing to the coupling between the vorticity and streamfunction variables, a coupled approach is preferred, thus treating ψ and ω as unknowns in the same algebraic linear system. The solution for the rest of the variables (v_θ, k, ε) is achieved in a sequential (segregated) manner. Whether a coupled or a segregated approach is used, the resulting matrix is sparse and without any particular structure and there are a number of suitable methods for solving the discretized equations. We used a sparse matrix solver from IBM's ESSL library.[19] The overall solution procedure can be outlined as follows.

(a) Guess all the necessary variables.
(b) Solve the vorticity and streamfunction transport equations.
(c) Solve the swirl velocity component.
(d) Solve the transport equations for k–ε.
(e) Calculate the near-wall values of k, ε and the velocity.
(f) Update the turbulent viscosity in the entire field.
(g) Treat the updated values of all variables as improved guesses and return to step 2 and repeat the process until convergence

5. NUMERICAL RESULTS

The procedure and method described above were applied to solve the following test cases:

(a) turbulent channel flow
(b) confined turbulent swirling flow.

The test cases are accompanied by experimental or numerical data from the literature which will be used to assess the validity and accuracy of the method.

5.1. Turbulent flow in a channel

In this problem the fluid enters a circular channel with a uniform velocity profile and a Reynolds number (based on the diameter and the average inlet velocity) equal to $3{\cdot}0 \times 10^5$ (Figure 4). The results are compared with the experimental data of Richman and Azad[20] and with Nikuradse's quasi-analytical velocity profile.[13] A non-uniform mesh containing 550 nodes (Figure 5) is used for the computations. Figure 6 shows the variation in the axial velocity (divided by the bulk velocity) at various heights in the channel. The correlation with the experimental data can be seen to be acceptable. In general our predictions tend to underestimate the developing length, while the velocity close to the wall is overestimated. This shows the sensitivity of the solution to adequate inlet conditions and a proper choice of the *a priori* set distance from the physical wall to the computational boundary (δ_1). For this simulation (experimental or

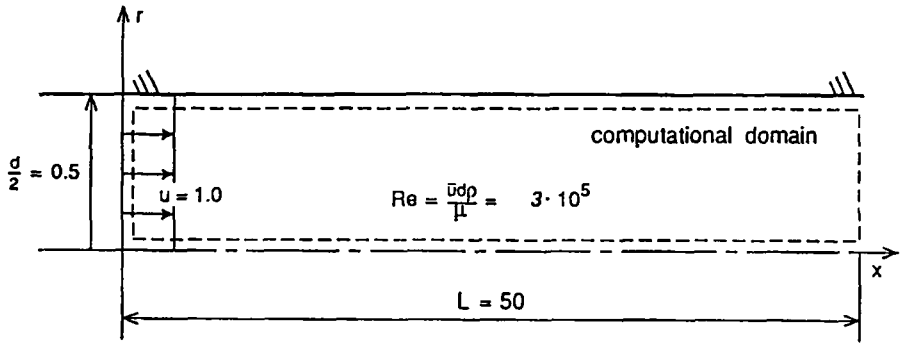

Figure 4. Global features of the turbulent channel flow

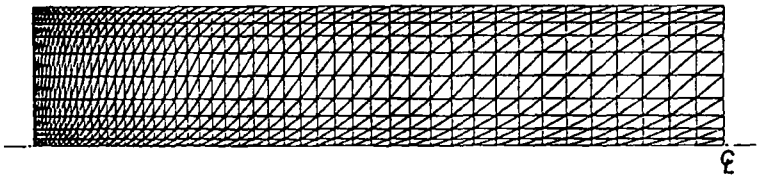

Figure 5. Mesh for the channel flow

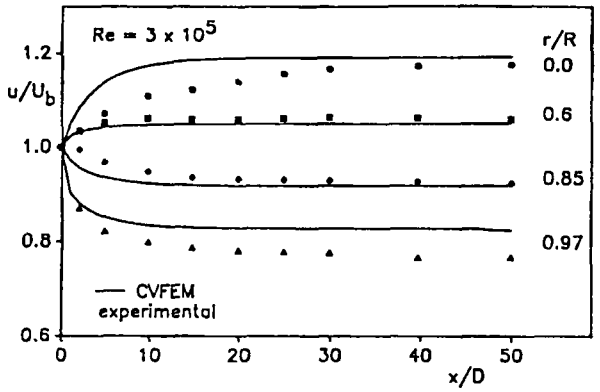

Figure 6. Axial variation in the velocity at various heights

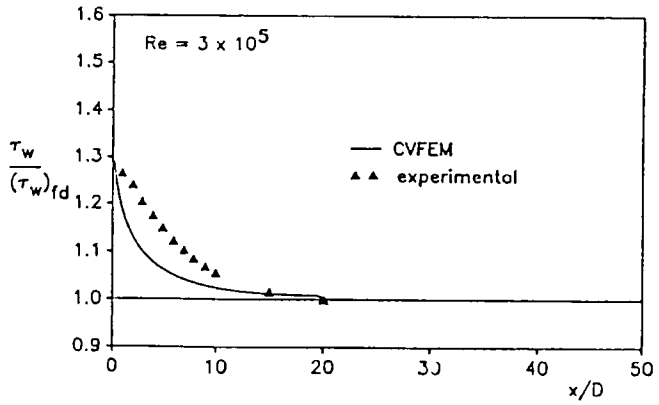

Figure 7. Axial variation in the wall shear stress

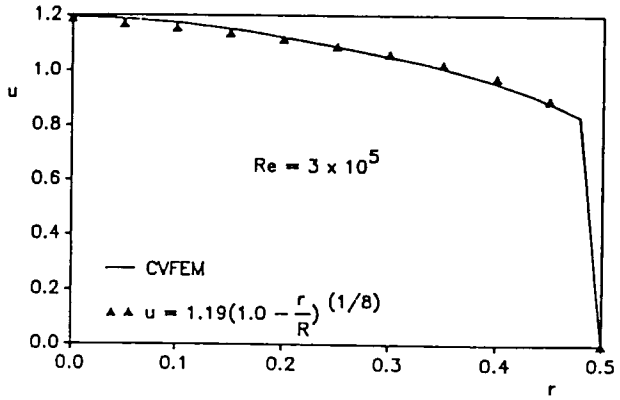

Figure 8. Radial variation in the exit velocity

numerical) it is common for authors to report an overshoot in the velocity at the centreline at around 25 diameters,[11,21] but that was not the case here. In Figure 7 the variation in the wall shear stress (over the fully developed value $(\tau_w)_{fd}$) is plotted. According to the experimental data, the wall shear stress reaches its fully developed value after approximately 20 diameters. The same observation can be made from our numerical predictions. Finally, the good agreement of the predicted fully developed velocity profile with the quasi-analytical solution is depicted in Figure 8.

5.2. Confined turbulent swirling flow

This test case (Figure 9) has been studied experimentally and numerically.[1] At the inlet a uniform velocity profile ($u = 4.7$ m s^{-1}) and a swirl number equal to 0.7 are prescribed. For k and ε a 1.5% turbulence intensity is used to compute the inlet values. The mesh contains 905 nodes distributed as shown in Figures 10 and 11. The effect on the velocity field of varying the swirl number from 0.0 to 0.7 is shown in Figures 12 and 13. Quite clearly, all the recirculation zone has changed, shifting from the corner after the diffusing part downwards. This result is

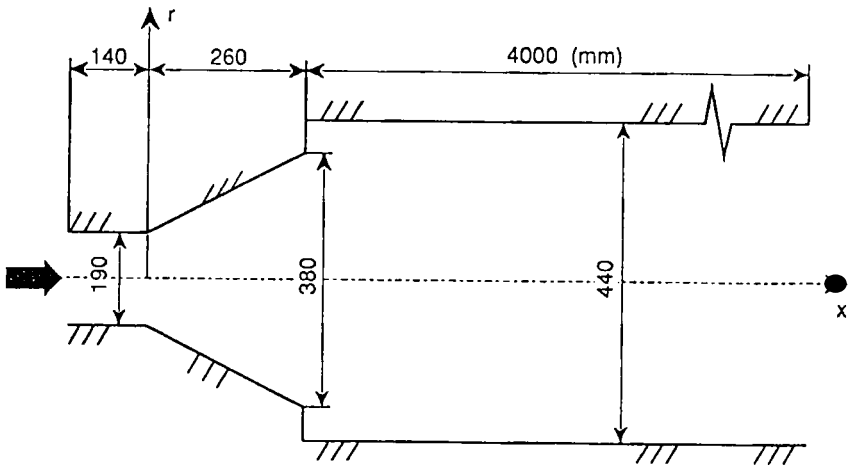

Figure 9. Global features of the confined turbulent swirling flow

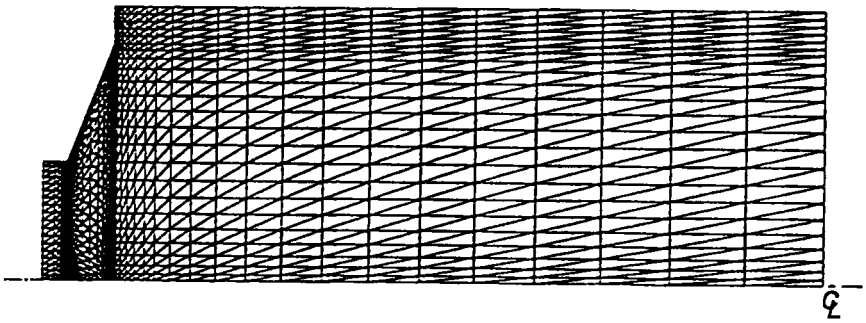

Figure 10. The computational mesh for the diffuser: 905 nodes

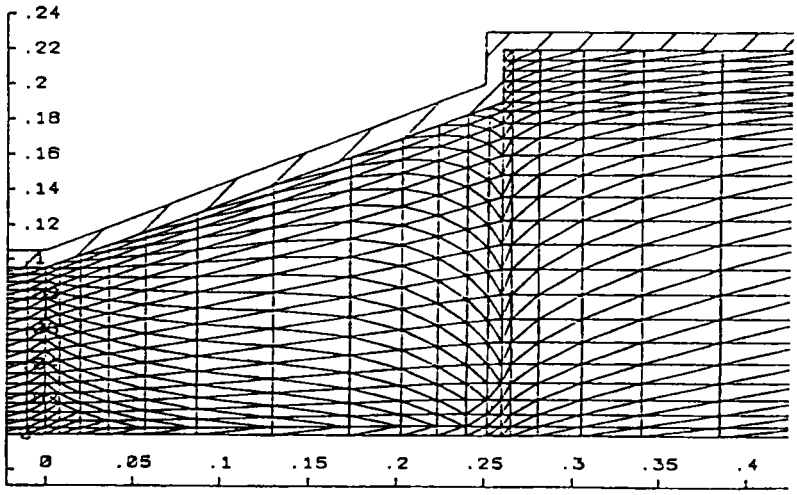

Figure 11. Detail of the computational mesh (scaled drawing)

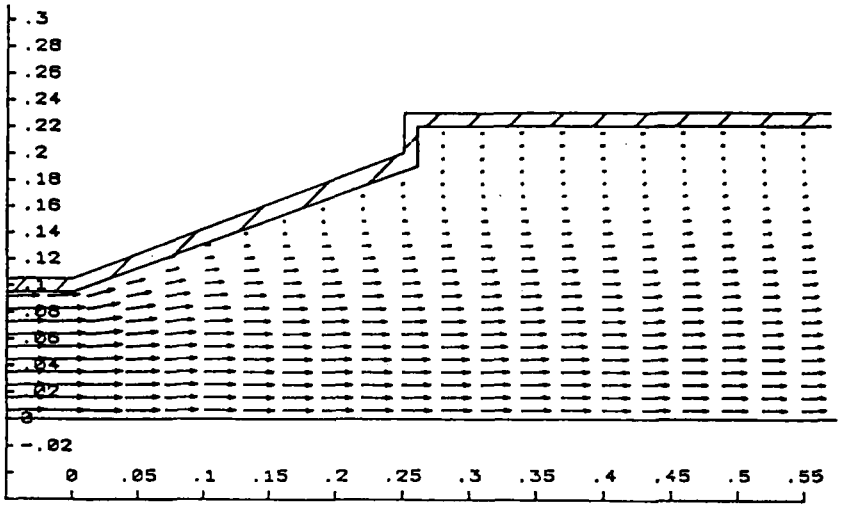

Figure 12. Velocity field in the diffusing part, $S = 0{\cdot}0$

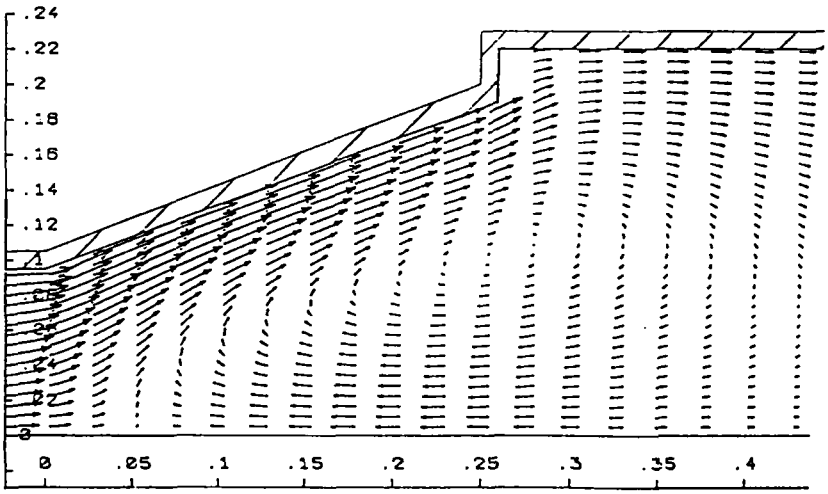

Figure 13. Velocity field in the diffusing part, $S = 0{\cdot}0$

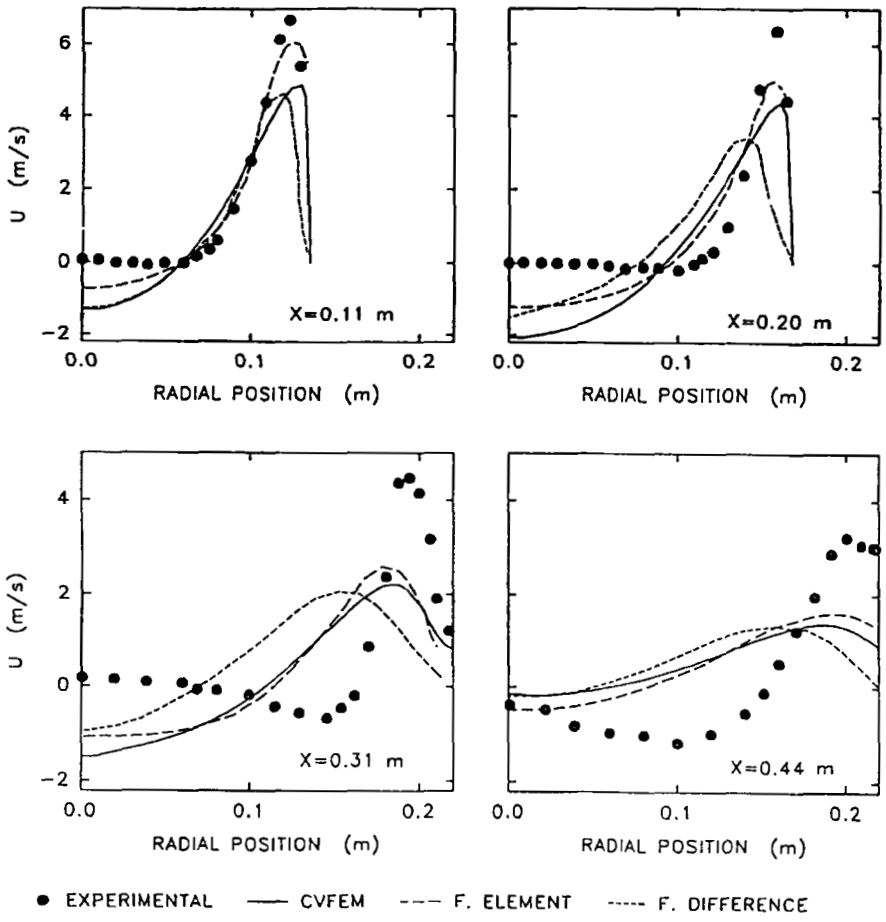

Figure 14. Radial distribution of the axial velocity component, $S = 0{\cdot}7$

corroborated by the FEM predictions of Benim[1] using the standard k–ε model. However, in addition to the central recirculation zone ($S = 0.7$), he also predicted a small vortex in the corner right after the diffusing part. We were not able to reproduce this result even with a finer grid in this region. The distribution of the x-velocity component at several stations downstream is plotted in Figure 14 and compared with experimental and numerical data.[1] Our predictions agree best with the finite element numerical predictions of Benim.[1] The use of the exponential shape function did give the same benefits as the use of the SUPG procedure of the finite element computations. The solution is indeed less diffusive when compared with the finite differences predictions. Nevertheless, the correlation with the experimental data is poor. In fact, Benim has clearly shown[1] that the use of the k–ε turbulence model is deficient for highly swirling flows and that an algebraic stress model makes it possible to obtain much better predictions. The same conclusions may be drawn from an examination of Figure 15, which shows the radial distribution of the tangential velocity component.

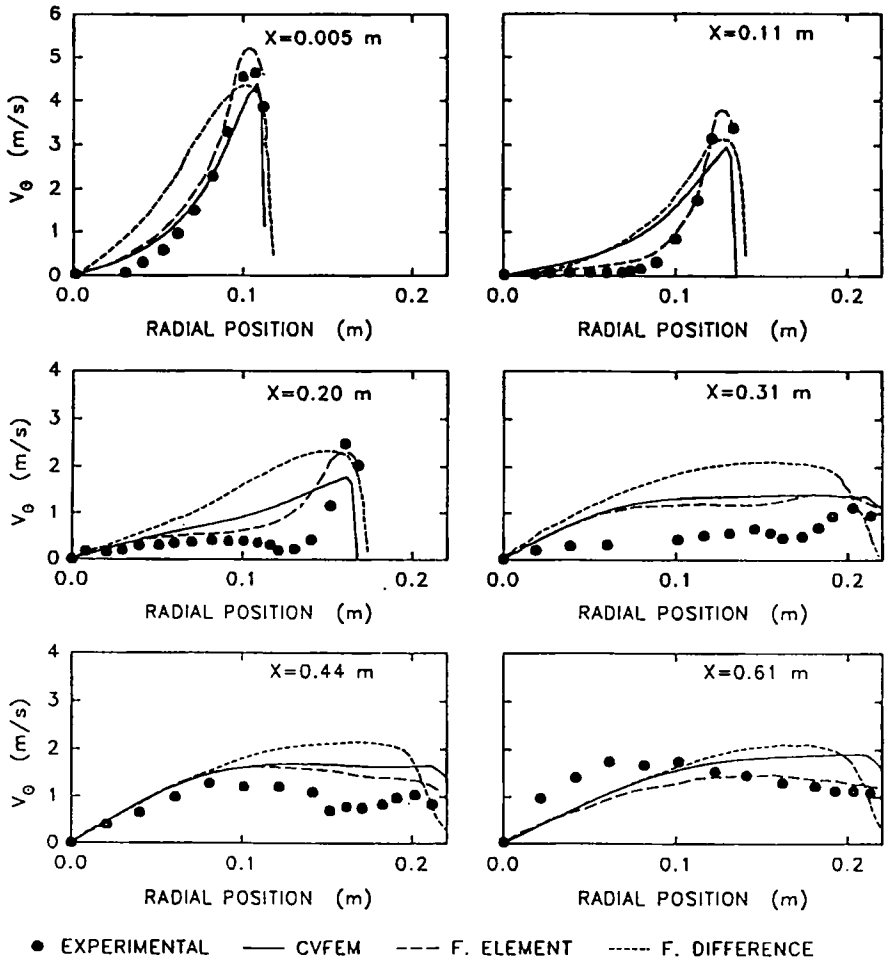

Figure 15. Radial distribution of the tangential velocity component, $S = 0{\cdot}7$

5.3. *Mesh refinements and CPU time*

In order to test the effects of mesh refinement on the solution of the step diffuser swirling test case, computations were performed using a coarser grid containing 409 nodes as well as a finer grid with 2030 nodes. The effect of mesh refinement on the axial and tangential velocities is shown in Figures 16 and 17. No significant improvement is gained because of mesh refinement.

A typical running CPU time for a swirling case such as the step diffuser presented above (around 900 nodes) is 150 s on an IBM 3090 180 VF. Convergence is tested by computing the maximum relative error of all the variables between two successive iterations. To reach a convergence criterion of 1×10^{-3}, 200 iterations are needed for the step diffuser containing around 90 nodes. Relaxation factors of 0.2 for k, ε and v_θ had to be used during the iterative procedure.

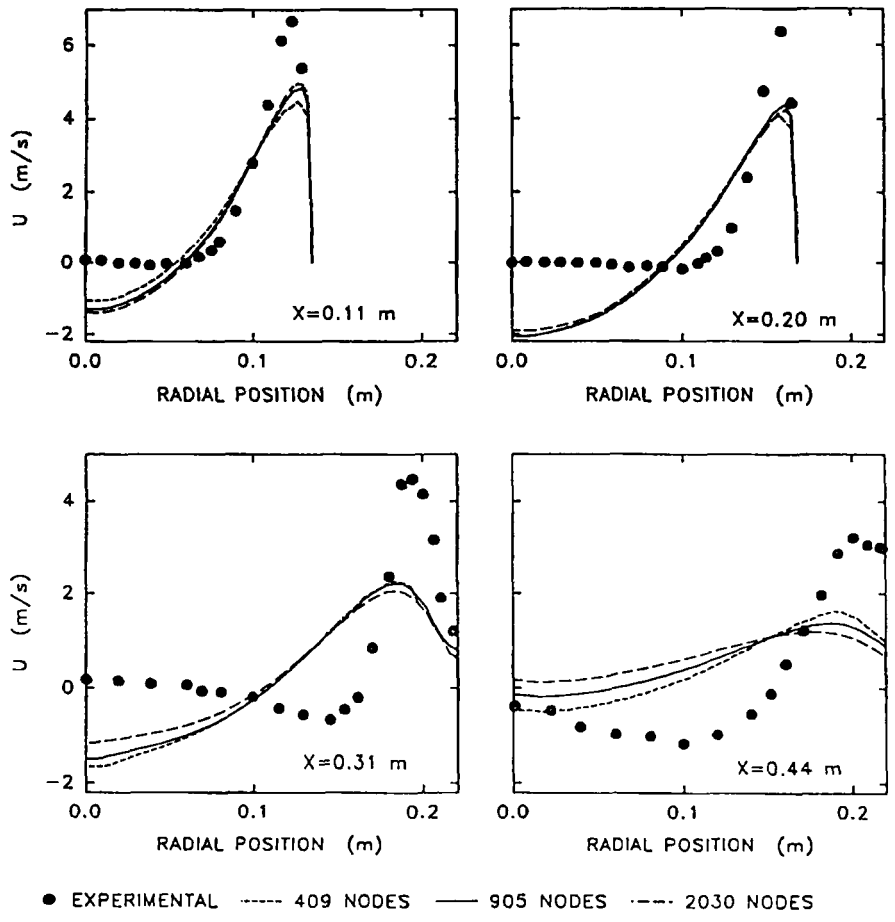

Figure 16. Effect of mesh refinement on the axial velocity component

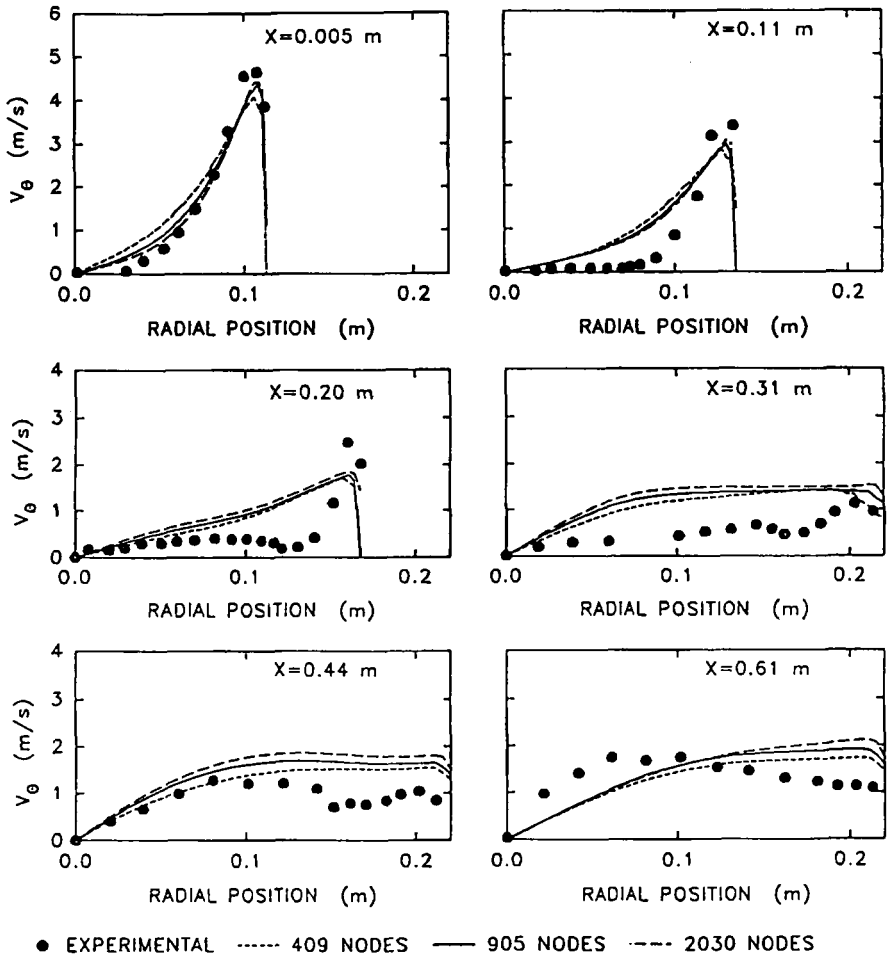

Figure 17. Effect of mesh refinement on the tangential velocity component

6. CONCLUSIONS

The control volume finite element method with the vorticity–streamfunction formulation has been used to solve turbulent swirling flows. An exponential interpolation function helped reduce the false diffusion effects on the solution. There was a resemblance between the results of the simulation and the results of an SUPG finite element procedure. The near-wall region was treated via the wall function method with a particular application to our formulation. To improve the quality of the predicted results, a better physical modelling of the turbulence is necessary. The computational behaviour of the proposed procedure was observed to be stable and accurate within the limits of the physical models.

ACKNOWLEDGEMENTS

The authors acknowledge the financial support provided for this research by the CRIM (Centre de Recherche Informatique de Montréal), the FP (Fondation de Polytechnique), Gaz Métropolitain Inc. of Québec and the NRCC (National Research Council Canada).

REFERENCES

1. A. C. Benim, 'Finite element analysis of confined turbulent swirling flows', *Int. j. numer. methods fluids*, **11**, 697–717 (1990).
2. J. G. Rice and R. J. Schnipke, 'A monotone streamline upwind finite element method for convection-dominated flows', *Comput. Methods Appl. Mech. Eng.*, **48**, 313–327 (1985).
3. T. J. R. Hughes, M. Mallet and A. Mikuzami, 'A new finite element formulation for computational fluid dynamics: II. Beyond SUPGU, *Comput. Methods Appl. Mech. Eng.*, **54**, 341–355 (1986).
4. B. R. Baliga and S. V. Patankar, 'A new finite-element formulation for convection–diffusion problems', *Numer. Heat Transfer*, **3**, 393–409 (1980).
5. G. D. Raithby, 'A critical evaluation of upstream differencing applied to problems involving fluid flow', *Comput. Methods Appl. Mech. Eng.*, **9**, 75–103 (1976).
6. G. De Vahl Davis and G. D. Mallinson, 'An evaluation of upwind and central difference approximations by a study of recirculating flow', *Comput. Fluids*, **4**, 24–43 (1976).
7. N. A. Hookey and B. R. Baliga, 'Evaluation and enhancements of some control volume finite-element methods —Part 1. Convection–diffusion problems', *Numer. Heat Transfer*, **14**, 255–272 (1988).
8. C. Liu and S. McCormick, 'The finite volume-element method (FVE) for planar cavity flow', *Proc. 11th Int. Conf. on Numerical Methods in Fluid Dynamics*, Williamsburg, VA, June–July 1988, Lecture Notes in Physics, vol. 323, D. L. Dwoyer, M. Y. Hussaini and R. G. Voigr (Eds), Springer-Verlag, Berlin, p. 622, (1989).
9. C. F. Kettleborough, S. R. Husain and C. Prakash, 'Solution of fluid flow problems with the vorticity–streamfunction formulation and the control-volume-based finite-element method', *Numer. Heat Transfer B*, **16**, 31–58 (1989).
10. B. E. Launder and D. B. Spalding, 'The Numerical computation of turbulent flows', *Comput. Methods Appl. Mech. Eng.*, **3**, 269–289 (1974).
11. A. C. Benim and W. Zinser, 'Investigation into the finite element analysis of confined turbulent flows using a k–ε model of turbulence', *Comput. Methods Appl. Mech. Eng.*, **51**, 507–523 (1985).
12. A. D. Gosman, W. M. Pun, A. K. Runchal, D. B. Spalding and M. Wolfshtein, *Heat and Mass Transfer in Recirculating Flow*, Academic, New York, 1969.
13. H. Schlichting, *Boundary Layer Theory*, 7th edn, McGraw-Hill, New York, 1979.
14. B. R. Baliga and S. V. Patankar, 'A control volume-based finite element method for two-dimensional fluid flow and heat transfer', *Numer. Heat Transfer*, **3**, 245–261 (1983).
15. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, Washington, DC, 1980.
16. C. Prakash and S. V. Patankar, 'A control volume-based finite element method for solving the Navier–Stokes equations using equal-order velocity–pressure interpolation', *Numer. Heat Transfer*, **8**, 259–280 (1985).
17. N. A. Hookey and B. R. Baliga, 'Evaluation and enhancements of some control volume finite-element methods— Part 2. Incompressible fluid flow problems', *Numer. Heat Transfer*, **14**, 273–293 (1988).
18. D. Elkaim, M. Reggio and R. Camarero, 'Numerical solution of reactive laminar flow by a control volume-based finite-element method and the vorticity–streamfunction formulation', *Numer. Heat Transfer B*, **20**, 223–240 (1991).
19. *Engineering and Scientific Subroutine Library*, 4th edn, Kingston, N.Y., IBM, 1983.
20. J. W. Richman and R. S. Azad, 'Developing turbulent flow in smooth pipes', *Appl. Sci. Res.*, **28**, 419–441 (1973).
21. R. B. Dean and P. Bradshaw, 'Measurements of interacting shear layers in a duct', *J. Fluid Mech.*, **78**, 641–676 (1976).