Third International Conference on CFD in the Minerals and Process Industries
CSIRO, Melbourne, Australia
10-12 December 2003

COMBINED VERTEX-BASED – CELL-CENTRED FINITE VOLUME METHOD FOR FLOWS IN COMPLEX GEOMETRIES

Diane MCBRIDE, Nick CROFT and Mark CROSS

Centre for Numerical Modelling and Process Analysis,
University of Greenwich,
London SE10 9LS

ABSTRACT

CFD modelling of 'real-life' processes often requires solutions in complex three dimensional geometries, which can often result in meshes where aspects of it are badly distorted. Cell-centred finite volume methods, typical of most commercial CFD tools, are computationally efficient, but can lead to convergence problems on meshes which feature cells with highly non-orthogonal shapes. The vertex-based finite volume method handles distorted meshes with relative ease, but is computationally expensive. A combined vertex-based – cell-centred (VB-CC) technique, detailed in this paper, allows solutions on distorted meshes that defeat purely cell-centred (CC) solutions. The method utilises the ability of the vertex-based technique to resolve the flow field on a distorted mesh, enabling well established cell-centred physical models to be employed in the solution of other transported quantities. The VB-CC method is validated with benchmark solutions for thermally driven flow and turbulent flow. An early application of this hybrid technique is to three-dimensional flow over an aircraft wing, although it is planned to use it in a wide variety of processing applications in the future.

NOMENCLATURE

ϕ	transported variable
p	pressure
$\mathbf{u}$	velocity
Γ	diffusion coefficient
ρ	density
μ	dynamic viscosity
c	specific heat
K	thermal conductivity
T	temperature
S	source
t	time
k	turbulent kinetic energy
ε	dissipation rate
ν_t	kinematic viscosity
β	thermal coefficient of volumetric expansion
g	gravity
T_{ref}	reference temperature

INTRODUCTION

The accuracy of Computational Fluid Dynamics (CFD) analysis not only depends upon the quality of the discretisation approaches within the target code to accurately model the physical process, but also on the ability to solve on a mesh that matches the true geometry of the physical domain. The Cell-Centred - Finite Volume discretisation method (CC-FV) is well established in CFD analysis for modeling physical processes involving the solution of flow and is employed in many CFD codes (e.g. CFX, FLUENT, STAR-CD and PHYSICA). This technique is computationally efficient, on a highly orthogonal mesh, using simple approximations to discretise the terms in the transport equation, it has low memory requirements and fast simulation times. However, the method is not robust on a highly non-orthogonal mesh. Corrections have to be made to the usual discretisation process to account for non-orthogonality in the mesh (Croft, 1995 & 1998). These correction terms can introduce errors into the solution process and lead to difficulties with convergence on highly distorted meshes (Croft, 1998). Modeling 'real-life' processes often requires fitting a mesh to complex geometries. Of course, fitting a highly orthogonal mesh to 'real-life' geometry can be one of the most time consuming aspects of the CFD modelling process. For example, close coupling between different physical phenomena, such as flow and stress, may result in the occurrence of mesh distortion during the solution process. In such *multi-physics* problems, even if one starts with a high quality mesh it may degrade during the solution process. The Vertex-Based – Finite Volume discretisation approach (VB-FV), that utilises element piecewise linear shape functions, (Prakash and Patankar, 1985), handles such distorted meshes with relative ease but is computationally expensive and requires considerable software restructuring for existing CFD software tools. In this paper we outline a novel discretisation approach which utilises the VB approach for the flow solution and employs well established physics models that use cell-centred (CC-FV) techniques for other transported properties. The coupled vertex-based – cell-centred (VB-CC) hybrid approach allows solutions on highly distorted meshes that defeat purely cell-centred solutions and is relatively straightforward to embed within generic CC based CFD tools.

GOVERNING EQUATIONS

The transport equation (1) for the general transport of a scalar variable is used as a starting point for FV procedures.

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot \left(\rho \mathbf{u} \phi\right) = \nabla \cdot \left(\Gamma \nabla \phi\right) + S_\phi \tag{1}$$

The momentum transport equations (2) can be written in the same form as above, with $\phi = u$, v or w and $\Gamma = \mu$. The pressure gradient term that forms the main momentum source term is written separately.

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \left(\rho \mathbf{u}\mathbf{u}\right) = -\nabla p + \nabla . \mu \nabla \mathbf{u} + S_u \tag{2}$$

The velocity field must also satisfy mass conservation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \mathbf{u}\right) = S_m \tag{3}$$

The general equation governing heat transfer can be written in the same form as above:

$$\frac{\partial \rho c T}{\partial t} + \nabla \cdot \left(\rho c \mathbf{u} T\right) = \nabla \cdot \left(K \nabla T\right) + S_T \tag{4}$$

The popular k-ε model (Launder and Spalding, 1974) involves the solution of the turbulent kinetic energy (k) equation is given by,

$$\frac{\partial \rho k}{\partial t} + \nabla \cdot \left(\rho \mathbf{u} k\right) = \nabla \cdot \left(\left[\mu_{lam} + \frac{\rho \nu_t}{\sigma_k}\right] \nabla k\right) + \rho \nu_t G - \rho \varepsilon \tag{5}$$

and the dissipation rate (ε) equation is as follows:

$$\frac{\partial \rho \varepsilon}{\partial t} + \nabla \cdot \left(\rho \mathbf{u} \varepsilon\right) = \nabla \cdot \left(\left[\mu_{lam} + \frac{\rho \nu_t}{\sigma_\varepsilon}\right] \nabla \varepsilon\right) + C_1 \rho \nu_t G \frac{\varepsilon}{k} - C_2 \rho \frac{\varepsilon^2}{k} \tag{6}$$

where the rate of generation of the turbulent kinetic energy, G, is given by:

$$G = 2\left(\left[\frac{\partial u}{\partial x}\right]^2 + \left[\frac{\partial v}{\partial y}\right]^2 + \left[\frac{\partial w}{\partial z}\right]^2\right) + \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)^2 + \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right)^2 \tag{7}$$

The turbulent viscosity is related to k and ε by,

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{9}$$

The values of empiric constants employed in equations (5) to (9) are:

$$C_\mu = 0.09; \sigma_k = 1.0; \sigma_\varepsilon = 1.3; C_1 = 1.44; C_2 = 1.92$$

CONTROL VOLUMES

In the VB-CC method equations (2) and (3) are discretised over a vertex-based control volume. The mesh element is subdivided into a number of sub-control volumes by connecting the element centroid to the element face centre. The sub-control volumes are assembled around the mesh vertex to form the vertex-based control volume, shown in Figure 1, for a two dimensional quadratic mesh.

The control volume associated with equations (4), (5) and (6) is simply the mesh element.

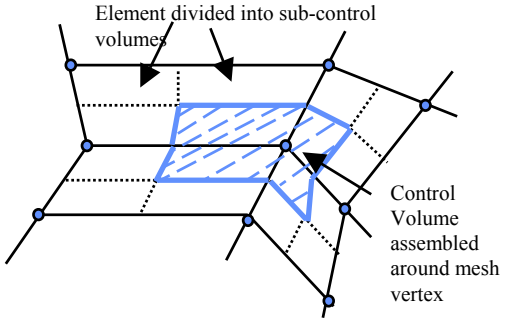


Figure 1: Vertex-based control volume

INTERPOLATION FUNCTIONS

The cell-centred approach uses finite-difference type approximations to describe how ϕ varies between solution points. However non-orthogonal meshes require corrections to be made to the usual discretisation process (Croft, 1995, 1998). The diffusive flux across a face requires the inclusion of a secondary gradient term, this can lead to face fluxes that are no longer computed in terms of neighbouring values. Most of the terms in the discretised transport equation require face values of ϕ, for orthogonal meshes this is achieved through interpolation of adjacent cell values, for non-orthogonal terms meshes an extra term is required based on the gradients of ϕ. The inclusion of these correction terms can introduce errors into the solution procedure leading to difficulties with convergence. These errors are multiplied when solving coupled variables and on arbitrary distorted meshes and divergence is often encountered.

In the vertex-based approach the local variation of a variable ϕ within an element is described by simple piecewise polynomial functions. The interpolation functions employed here are given in (Taylor et al, 2003) who used them in structural analysis. The variables and co-ordinates are approximated, in local co-ordinates as,

$$u_i(s,t,u) = \sum_{j=1}^{n} N_j(s,t,u) u_{ij}$$

$$p(s,t,u) = \sum_{j=1}^{n} N_j(s,t,u) p_j \qquad (10)$$

$$x_i(s,t,u) = \sum_{j=1}^{n} N_j(s,t,u) x_{ij}$$

where n is the number of nodes of an element.

The use of elemental shape functions allows the direct computation of fluxes in the required direction even on a non-orthogonal mesh, see McBride (2003).

COMBINED VB-CC METHOD

In the solution of the Navier Stokes equations the revised SIMPLER method of Patankar (1980) is employed. Correct pressure and velocity coupling is ensured by the method of Prakash and Patankar (1985). Obtaining a flow field using vertex-based techniques allows vertex-based velocities to be employed in the transport of other quantities using cell-centred techniques. Mass is conserved on the boundary of the vertex-based control volume. Since the element face centroid is a point on the boundary of the vertex-based control volume, indirectly mass is also conserved over the mesh element. As mass conservation is enforced over the vertex-based control volume, any errors resulting from interpolating for element face values also decrease. See the work of McBride (2003) for a detailed description of the computational approach.

EVALUATION OF THE METHOD

Results have been developed for a number of benchmark problems, including the thermally driven flow case discussed below. The approach was then applied to a three-dimensional flow over an aircraft wing. The results are shown for a uniform Cartesian mesh and distorted versions of the same mesh.

THERMALLY DRIVEN FLOW

De Vahl Davis and Jones (1983), suggested that buoyancy-driven flow in a square cavity would be a suitable validation test case for CFD codes and published a set of benchmark results for a number of different Rayleigh numbers. Declining quality in solutions is often encountered with increasing Rayleigh number. The fluid contained in the cavity is assumed incompressible and initially stationary. Thermal gradients across the solution domain result from opposing walls of differing temperatures. These thermal gradients lead to buoyancy forces that create flow. The buoyancy forces are calculated using the Boussinesq approximation. This approximation results in a source per unit volume of the form,

$$S_i = -\rho\beta g_i (T - T_{ref}) \qquad (10)$$

The simulations were performed on a Athlon 1.39Ghz processor for a uniform 35 by 35 Cartesian mesh, mesh 1, and distorted versions of mesh 1, shown in Figure 2.

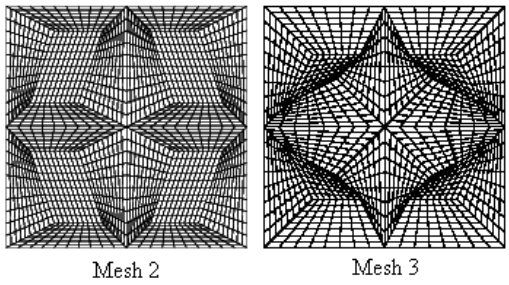
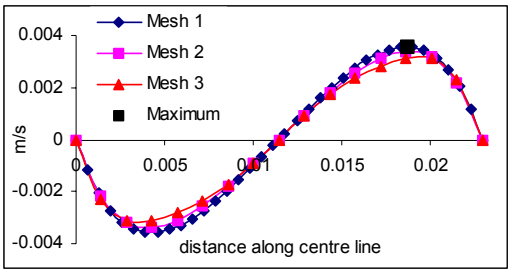


Figure 2: Distorted versions of Cartesian mesh 1

Plots of the u-velocity along the central vertical plane and the v-velocity along the central horizontal plane for each mesh and Rayleigh number are shown along with the benchmark maximum values. Figure 3 for Rayleigh number of 10^3, Figure 4 for Rayleigh number of 10^4, Figure 5 for Rayleigh number of 10^5 and Figure 6 for Rayleigh number of 10^6.

a)



b)

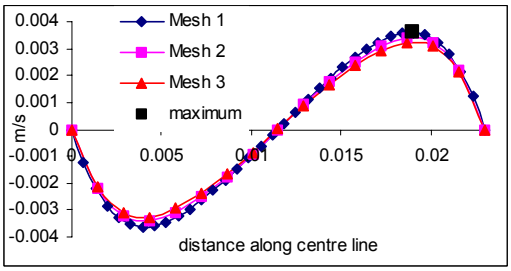
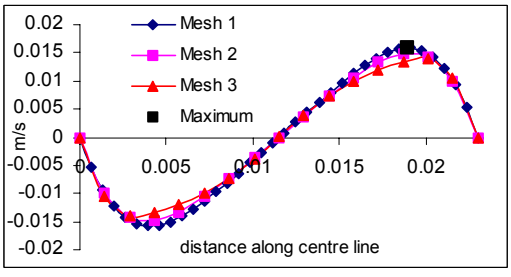


Figure 3: Rayleigh number of 10^3, a) u-velocity b) v-velocity

a)



b)

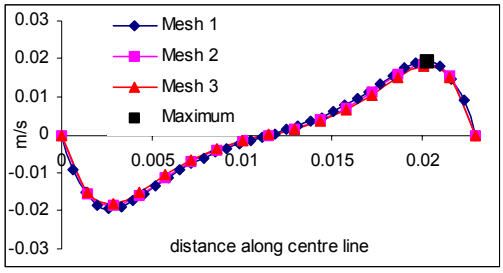
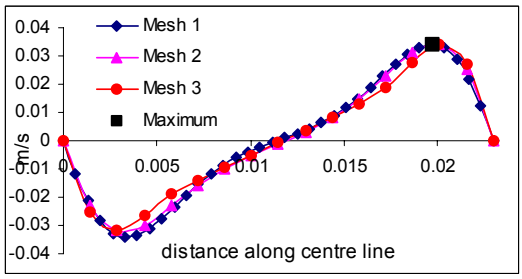


Figure 4: Rayleigh number of 10^4, a) u-velocity b) v-velocity

a)



b)

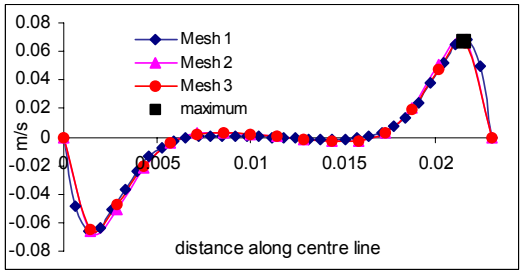
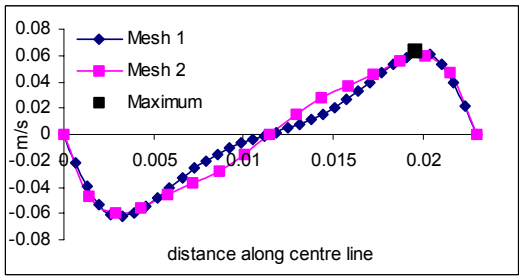


Figure 5: Rayleigh number of 10^5, a) u-velocity b) v-velocity

a)



b)

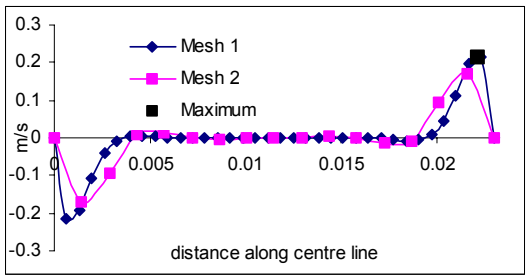


Figure 6: Rayleigh number of 10^6, a) u-velocity b) v-velocity

For the uniform mesh, Table 1 shows VB-CC U_{max} and V_{max}, the maximum value of the normalised velocity component along central planes. The Y_{max} and X_{max}, are the normalised positions of this maximum value. The percentage errors of the simulation results against benchmark solutions are shown in brackets for each Rayleigh number (Ra). VB-CC and CC results, on a 35 by 35 uniform mesh, compare well with benchmark solutions from non-uniform optimised meshes.

Ra	U_{max}	Y_{max}	V_{max}	X_{max}
10^3	3.638 (0.29%)	0.189 (1.0%)	0.686 (0.27%)	0.178 (0.12%)
10^4	16.194 (0.10%)	0.178 (0.43%)	19.570 (0.24%)	0.122 (2.6%)
10^5	34.780 (0.14%)	0.144 (0.38%)	69.600 (1.47%)	0.067 (0.97%)
10^6	63.867 (1.18%)	0.144 (3.71%)	219.85 (0.22%)	0.0333 (12.1%)

Table 1: VB-CC: Percentage error with benchmark values

A measure of the error, due to mesh distortion, for mesh 2 and mesh 3 is shown in Table 2 for VB-CC solutions, using mesh 1 as the base result. Divergence was encountered on mesh 3 for a Rayleigh number of 10^6.

Ra	Mesh 2 u-velocity	Mesh 2 v-velocity	Mesh 3 u-velocity	Mesh 3 v-velocity
10^3	$4.74\text{x}10^{-3}$	$4.71\text{x}10^{-3}$	$3.68\text{x}10^{-2}$	$3.32\text{x}10^{-2}$
10^4	$7.82\text{x}10^{-3}$	$8.39\text{x}10^{-3}$	$3.24\text{x}10^{-2}$	$3.72\text{x}10^{-2}$
10^5	$1.94\text{x}10^{-2}$	$1.58\text{x}10^{-2}$	$4.78\text{x}10^{-2}$	$1.28\text{x}10^{-2}$
10^6	$9.28\text{x}10^{-2}$	$3.94\text{x}10^{-2}$	-	-

Table 2: VB-CC: Error due to mesh distortion

Good agreement with benchmark solutions was obtained on the uniform mesh and the solutions were only slightly degraded on the distorted mesh. The question here is at what cost? In this two-dimensional thermally driven problem, the memory demands per solution point are 73 Bytes per element-based variable and 280 Bytes per vertex-based variable. Moreover, the convergence behaviour and compute time for the problem may be summarised as in Table 3 below.

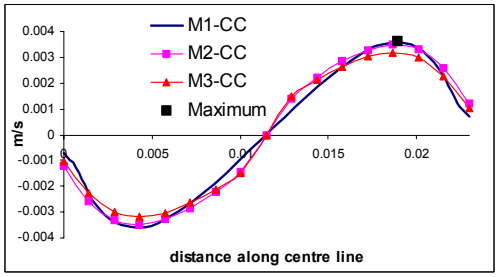
Perf. meas.	Ra	M1-CC	M1-VB	M2-CC	M2-VB	M3-CC	M3-VB
Its	10^3	56	122	75	190	125	213
Time(s)	10^3	1	18	2	32	5	35
Its	10^4	204	211	265	246	Fail	250
Time(s)	10^4	5	35	7	52	Fail	51
Its	10^5	229	187	284	206	Fail	207
Time(s)	10^5	5	31	7	42	Fail	42
Its	10^6	381	301	498	492	Fail	Fail
Time(s)	10^6	9	48	12	93	Fail	Fail

Table 3: Computational times: for a mesh consisting of 1225 elements and 2592 nodes.

It can be deduced from these results that the hybrid method is approximately a factor of 4 more expensive in compute time, per solution point, on a good quality mesh. On the distorted meshes CC solutions could only be

achieved without the inclusion of the non-orthogonal correction terms, consequently as the mesh degrades CC solutions deteriorate. For example, the error on mesh 2 for a Rayleigh number of 10^3 was approximately 10%, see Figure 7, compared to 0.5% for VB-CC solutions. The hybrid scheme continues to produce reasonably accurate solutions as the mesh degrades with a further compute cost of about a factor of 2.

a)



b)

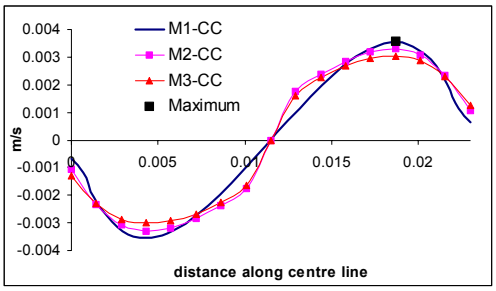
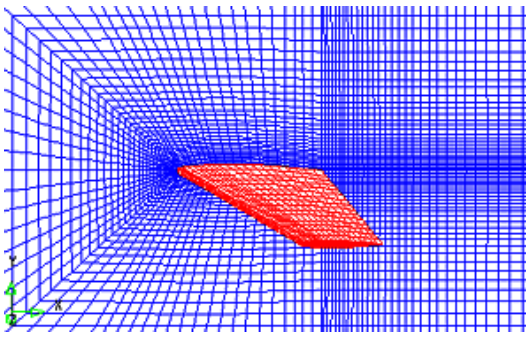


Figure 7: Cell-centred solutions: Rayleigh number of 10^3, a) u-velocity b) v-velocity

FLOW OVER AN AIRCRAFT WING

One reasonably challenging case study to assess the potential of the hybrid scheme involves turbulent incompressible flow over an aircraft wing using the *k-ε* model. The geometry of the wing was taken from ONERA M6 specifications; it is a swept, semi-span wing with no twist. The leading-edge sweep is 30 degrees, trailing edge sweep 15.8 degrees and the taper ratio is 0.562. The simulation carried out employs a low speed Mach number of 0.3, giving a Reynolds number of about 5 million. A wall boundary condition was applied to the wing surface for the flow and turbulence model variables. The simulation was performed on a uniform C-Mesh of approximately 100,000 elements and a distorted version of the mesh, both shown in Figure 8. This is a relatively coarse mesh, with complex flow simulations of flow over the ONERA wing a mesh of at least three to four hundred thousand elements is normally employed. However, the mesh density here is sufficient to explore the performance of the VB-CC hybrid method on a 'real-life' distorted mesh.

a)



b)

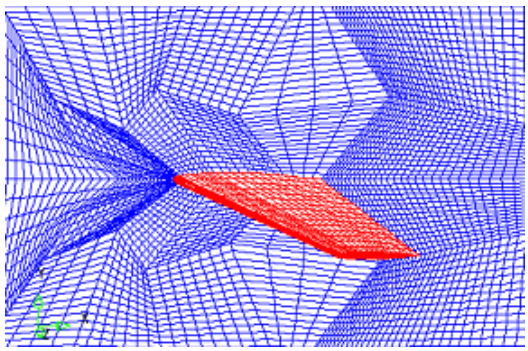


Figure 8: a) C-Mesh, b) Distorted C-Mesh

At the outset it is worth saying that on the C-mesh, the VB-CC hybrid method and the conventional CC discretisation results are very similar in most respects. For the VB-CC method on both the C-mesh and the distorted mesh, plots are shown on two planes, z= 0.558 which is approximately half the wing span, and y=0 which is the symmetry plane. Figures 9 and 10 show the mach contour plots for the Z-plane and Y-plane respectively. Although there is some smearing of values on the distorted mesh, the results have captured the overall trend, identifying local minimum and maximum values. The u- and w-velocity value range remained unchanged, being [0m/s to 107m/s] and [-6.37m/s to 39m/s] respectively. The minimum and maximum v-velocity values decreased slightly from [-49.3m/s, 49.3m/s] to [-40.4m/s, 40.4m/s] on the distorted mesh.

The turbulent viscosity contours, Figure 10, show some smearing of values on the distorted mesh. The lower velocity values downstream of the wing's trailing edge, on the distorted mesh, result in higher turbulent generation rates in this region and hence higher viscosity values. This is caused by the way that the turbulence generation rate is represented numerically at the trailing edge of the wing shape (McBride, 2003) and this problem can be eliminated with a more careful approximation. The maximum turbulent viscosity obtained on the C-mesh was $0.03m^2/s$ compared to $0.06m^2/s$ on the distorted mesh.

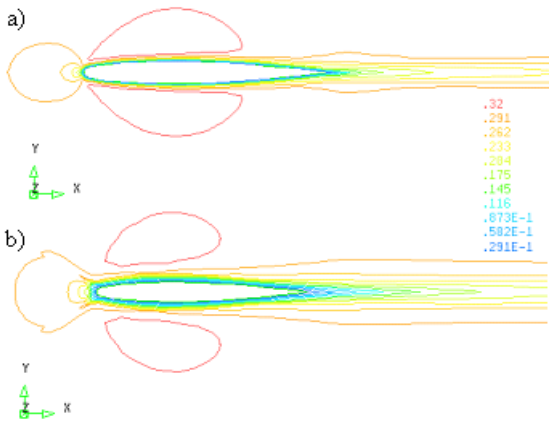


Figure 9: Mach no. Contour plots on plane z=0.558
a) C-mesh, b) Distorted C-mesh

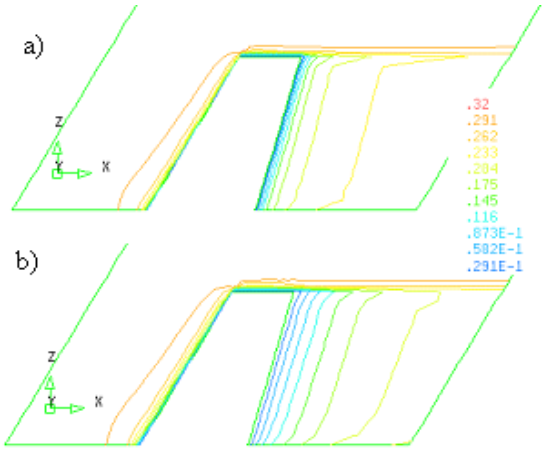


Figure 9: Mach no. Contour plots on plane y=0
a) C-mesh b) Distorted C-mesh

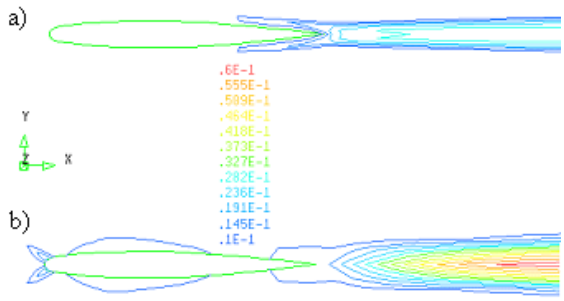


Figure 10: Viscosity contour plots on plane z=0.558
a) C-mesh b) Distorted C-mesh

Run Times and Memory Requirements

The simulations were performed on a Pentium 4 CPU 2.54GHz. The mesh employed comprised of 101,412 elements and 108,314 vertices. To achieve the convergence criteria that the *L2*norm of the change in the solution dropped by 5 orders-of-magnitude required 254 iterations on the uniform C-mesh and 302 iterations on the distorted mesh. The time per iteration/per solution point was approximately 3.3 x 10^{-5} seconds for each variable being solved vertex-based and 7.0 x 10^{-6} seconds for each variable being solved cell-centred. This yields, 15.72 seconds per iteration for VB-CC solutions, and 4.26 seconds per iteration for CC solutions. The vertex-based method has considerably more memory requirements than the cell-centred method. The approximate memory required per solution point is 373 bytes vertex-based compared to 42 bytes cell-centred.

CONCLUSION

The coupling of the VB-CC hybrid FV discretisation method for CFD involving vertex-based flow coupled with other transported quantities at the cell-centre (e.g. thermal and turbulent variables) has been presented. The cell-centred discretisation of transported quantities still includes non-orthogonal errors that may in turn introduce some error in to the flow field. However, the non-orthogonal errors do not appear to significantly affect the final solution and local minimum and maximum values are identified. Obtaining a good flow field on a distorted mesh using a VB method aids the solution of other transported quantities using efficient cell-centred techniques. As such, the results obtained on benchmark problems, using the VB-CC technique are encouraging.

Although the VB-CC method is approximately 4 times more expensive in compute time and requires 8 times as much memory than the conventional CC method, this new hybrid approach does enable solutions on distorted meshes that defeat purely cell-centred techniques whilst enabling well established cell-centred physical models to be retained. This approach is particularly useful in the generic CFD tool context, because it enables users to exploit all the existing models that they have already developed in CC context within the VB-CC framework and to obtain solutions in complex geometry meshes which have some zones of poor mesh quality.

REFERENCES

CROFT, N., PERICLEOUS, K.A. and CROSS, M., (1995), "PHYSICA: A multi-physics environment for complex flow processes", In C. Taylor and P. Durbetaki, editors, *Numerical Methods in Laminar and Turbulent Flow '95*, **2**, 1269-1280

CROFT, T.N., (1998), "Unstructured Mesh – Finite Volume Algorithms for Swirling, Turbulent, Reacting Flows", *Ph.D. thesis*, The University of Greenwich, London, England.

PRAKASH, C. and PATANKAR, S.V., (1985), "A Control Volume-Based Finite-Element Method for solving the Navier-Stokes equations using Equal-order velocity-pressure interpolation", *Numerical Heat Transfer,* **8**, 259-280

TAYLOR, G.A., BAILEY, C. and CROSS, M., (2003), "A vertex-based finite volume method applied to non-linear material problems in computational solid mechanics", *International journal for Numerical Methods in Engineering,* **56**, 507-529

PATANKAR, S.V., (1980), "Numerical Heat Transfer and Fluid Flow", *Hemisphere Publishing Corporation*.

McBRIDE, D., (2003), "Vertex-Based Discretisation Methods for Thermo – Fluid Flow in a Finite Volume-Unstructured Mesh Context", *Ph.D. thesis*, The University of Greenwich, London, England.

DE VAHL DAVIS, G. and JONES, I.P., (1983), "Natural Convection in a Square Cavity: A Comparison Exercise", *International Journal for Numerical Methods in Fluids,* **3**, 227-248

LAUNDER, B.E. and SPALDING, D.B., (1974), "The Numerical Computation of Turbulent Flows", *Comput. Meth. Appl. Mech. Eng.,* **3**, 269-289

FLUENT, see www.fluent.com

CFX, see www.ansys.com

STAR-CD, see www.cd-adapco.com

PHYSICA, see www.multi-physics.com