INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **53**:81–104
Published online 19 June 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1250

A coupled finite volume method for the solution of flow processes on complex geometries

D. McBride*,†, T. N. Croft‡ and M. Cross§

School of Engineering, University of Wales Swansea, Singleton Park, Swansea SA2 8PP, Wales, U.K.

SUMMARY

CFD modelling of 'real-life' thermo-fluid processes often requires solutions in complex three-dimensional geometries, which can result in meshes containing aspects that are badly distorted. Cell-centred finite volume methods (CC-FV), typical of most commercial CFD tools, are computationally efficient, but can lead to convergence problems on meshes that feature cells with highly non-orthogonal shapes. The control volume-finite element method (CVFE) uses a vertex-based approach and handles distorted meshes with relative ease, but is computationally expensive. A combined vertex-based—cell-centre technique (CFVM), detailed in this paper, allows solutions on distorted meshes where purely cell-centred solutions procedures fail. The method utilizes the ability of the vertex-based approach 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 vertex-based flow code is verified against a manufactured 3D solution and error norms are compared on meshes with various degrees of distortion. The CFVM method is validated with benchmark solutions for thermally driven flow and turbulent flow. Finally, the method is illustrated on three-dimensional turbulent flow over an aircraft wing on a distorted mesh where purely cell-centred techniques fail. The CFVM is relatively straightforward to embed within generic CC based CFD tools allowing it to be employed in a wide variety of processing applications. Copyright © 2006 John Wiley & Sons, Ltd.

Received 4 November 2005; Revised 28 March 2006; Accepted 29 March 2006

KEY WORDS: vertex-based; cell-centred; hybrid; distorted mesh; non-orthogonal

1. INTRODUCTION

In the numerical modelling of processes involving fluids the flows of interest generally occur in geometrically complex domains. The accuracy of CFD analysis not only depends upon the quality of the discretization approaches within the target code to model the physical process,

*Correspondence to: D. McBride, School of Engineering, University of Wales Swansea, Singleton Park, Swansea SA2 8PP, Wales, U.K.
†E-mail: d.mcbride@swan.ac.uk
‡E-mail: t.n.croft@swan.ac.uk
§E-mail: m.cross@swan.ac.uk

but also on the ability to solve on a mesh that matches the true geometry of the physical domain. Significant advances have been made in the development of numerical methods for the solution of flow on unstructured meshes. The most frequently used unstructured mesh discretization methods are the finite element method (FEM) [1] and the finite volume method (FV) [2]. However, it is the Finite Volume principles, of applying conservation laws locally to control volumes, which lend themselves to easy physical interpretation (such as, fluxes, source terms and local conservation principles). This has led to the FV method being the preferred technique used in most of the commercially available CFD codes. Even successful finite element CFD codes, such as, FLITE3D [3], originally developed at the University of Wales Swansea and now extensively used within the aerospace industry [4], are formulated to ensure local conservation using a finite volume framework. There are a number of FV approaches, usually either vertex-centred, where the unknowns are defined at the mesh nodes, or cell-centred, where the unknowns are defined at the element centroid. The most widely used is the cell-centred (CC) approach, which is employed in many CFD codes (e.g. CFX [5], FLUENT [6], STAR-CD [7] and PHYSICA [8]). This technique is computationally efficient, on a highly orthogonal mesh, using simple approximations to discretize 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 discretization process to account for non-orthogonality in the mesh, see Section 3. On distorted grids these corrections are only first-order accurate [9]. Local errors appear in the solution dependent on the extent of mesh distortion and the solution may diverge. Peric [10] investigated this effect on distorted meshes and proposed grid refinement and smoothing to reduce the error. Moulinec [11] investigated several different interpolation schemes to improve the derivatives at the cell faces on distorted meshes. Other authors [12, 13] proposed different interpolation schemes based on Taylor series expansions. Lehnhauser [14] reported significantly improved accuracy with a multi-dimensional Taylor series expansion scheme, which ensured second-order accuracy, even on strongly distorted meshes. The problems encountered in discretizing the transport equations on distorted meshes are exacerbated when employing pressure-correction schemes in the solution of the incompressible Navier–Stokes equations. Addition of the non-orthogonal correction terms in the pressure-correction equation is expensive and complex and it is common practice to omit these terms [15, 16]. However, omission or simplification of these terms can introduce stability problems into the solution process and lead to difficulties with convergence on highly distorted meshes. Since in complex geometries it is often not possible to have a good quality mesh over the entire domain, many authors have sought to address this problem [17–20].

Various FV discretization techniques have been developed for unstructured meshes including, edge-based schemes that employ dual control volumes and edge-based data structures [2, 21–23] as well as others who use cell-based gradient reconstruction [24–26] who used cell vertex techniques [27, 28], who employed a vertex-centred scheme for flow past complex geometries and who store solved variables at the cell-circumcentres to enable solutions on unstructured meshes [29, 30]. Any discretization technique that employs the standard linear central differencing scheme can encounter difficulties when approximating the control volume face derivative on a non-orthogonal grid. Standard schemes are only sufficient if the points involved in calculating control volume face values are connected by a straight line, which is normal to the boundary face, as is the case on an orthogonal mesh. On complex grids, this is not usually the case and the accuracy of the scheme deteriorates in areas of poor quality. In multi-physics problems, such as, dynamic fluid–structure interaction, even if one starts with a high quality mesh, distortion can occur during the solution process, see Reference [31], for example, who discusses this problem. CD-Adapco has sought to address this problem by employing polyhedral control volumes with essentially VB discretization schemes. It has recently introduced STAR-CCM+ [32], a multi-physics solver based on polyhedral meshes. They claim that good quality meshes consisting of polyhedral elements can be generated more easily than tetrahedral meshes, whilst they have more neighbouring points and require more storage and compute time per cell, they give higher accuracy in computing gradients of dependent variables. In this context, it is worth noting their claim that polyhedral control volumes can be generated from any type of mesh by defining a dual control volume, potentially providing additional flexibility in mesh generation for very complex geometries.

The control volume-finite element method (CVFE), developed by Prakash [33] and described by Baliga [34] has successfully been applied to complex flow problems with irregular geometries [35]. This method can be viewed in a finite volume context as a vertex-based finite volume method (VB) [36]. A polyhedron control volume is constructed around the mesh vertex and the local variation of a variable within a mesh element is described by simple piecewise polynomial functions allowing distorted meshes to be handled with relative ease. This flexibility of applying the VB discretization technique to arbitrary irregular meshes is appealing. However, the method requires extensive storage/topological information and the computational cost is far more expensive than the cell-centred technique. Combining aspects of both the FEM and FVM has become increasingly popular and has been applied in a number of research areas. A mixed finite element-finite volume method, originally developed by Dervieux [37] has been employed by many in the simulation of turbulent flow, including 3D turbulent compressible flow [38] and Large Eddy simulations [39]. This method uses finite element discretization for the diffusive part of the Navier–Stokes equations and finite volume computations for all other terms. Durlofsky [40] combined finite volume-finite elements in the solution of multi-phase flow in porous media. Here the pressure equation is discretized using finite elements and the fluid phases are computed using the finite volume method. This method has been applied by Mazzia and Putti [41] and Chavent *et al*. [42] amongst others. Hybrid finite volume-finite elements have also been employed for viscoelastic flows [43]. Here the FEM is applied to the continuity and momentum equations and the FVM to the constitutive equations for stress. Chan [44] used a finite volume formulation for the 3D Navier–Stokes equations of incompressible flow and a finite element discretization of the pressure-correction equation. In the foregoing methods, although not an exhaustive list, the cell–vertex finite volume method has commonly been employed on triangle–tetrahedral meshes. The variables are all stored at the same locations, the mesh vertices. McBride [45] investigated the CC and VB approaches in the solution of the incompressible Navier–Stokes equations. Co-located and partially staggered, i.e. velocity CC and pressure VB and *vice versa*, solutions were compared on unstructured skewed meshes. The co-located VB approach was shown to handle the distorted meshes with relative ease. Solutions obtained on the skewed meshes were comparable with solutions obtained on a uniform Cartesian mesh. Whereas, the partially staggered hybrid approaches enabled solutions where purely CC techniques failed, the results obtained on the distorted mesh contained non-orthogonal errors.

This paper employs a combined vertex-based—cell-centre technique (CFVM) in the solution of thermal and turbulent flow. The method utilizes the vertex-based approach to resolve the flow field and employs cell centred discretization for all other transported variables. Resolving the flow field using VB techniques allows flow solutions on a distorted mesh, thus enabling well established cell-centred physical models to be employed in the solution of other transported quantities. The claim is that the CFVM approach allows solutions on highly distorted meshes that defeat purely

DOI: 10.1002/fld

cell-centred solutions and can be employed within generic CC based CFD tools. The CFVM is the vertex based method implemented within the host CFD code PHYSICA, [46] which provides a 3D unstructured mesh FV framework for multi-physics modelling, allowing existing CC physics models to be utilized. The motivation for this paper was to evaluate the CFVM method using benchmark problems for thermally driven flow and turbulent flow under circumstances where the quality of the mesh used is increasingly degraded. The objectives here are to evaluate how well the CFVM approach compares with a conventional CC discretization and to assess how well the former copes with mesh quality degradation in terms of convergence behaviour and solution accuracy. The assessment is completed with a three-dimensional case, flow over an aircraft wing, on a good quality mesh and after mesh degradation.

2. GOVERNING EQUATIONS

The starting point for FV procedures is the general transport equation for a scalar variable, ϕ, Equation (1). The key step of the FV method is the integration of the transport equation over each control volume, as well as over time, and a solution is sought which makes each of these integrals equal to zero

$$\frac{\partial(\boldsymbol{\rho\phi})}{\partial\mathbf{t}} + \nabla\cdot(\boldsymbol{\rho}\bar{\mathbf{u}}\boldsymbol{\phi}) = \nabla\cdot(\Gamma\nabla\boldsymbol{\phi}) + \mathbf{S}_\phi \tag{1}$$

The momentum transport equations can be cast in the same general form as (1), with ϕ equal to the components of velocity u, v or w and $\Gamma=\mu$. The pressure gradient term that forms the main momentum source term is written separately. The equations are given using tensor notation

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} - \frac{\partial}{\partial x_j}\left(\mu\frac{\partial u_i}{\partial x_j}\right) = S_i - \frac{\partial p}{\partial x_i} \tag{2}$$

The velocity field must also satisfy mass conservation

$$\frac{\partial\rho}{\partial t} + \frac{\partial(\rho u_i)}{\partial x_i} = 0 \tag{3}$$

Applying the FV methodology equations (2) and (3) are integrated over each control volume. The momentum equations (2) are integrated over a vertex-based control volume using CVFE techniques; the components of velocity are solved and stored at the mesh nodes. The continuity equation (3) is transformed into a pressure equation and also integrated over a vertex based control volume, pressure is solved and stored at the mesh nodes.

Equations describing the transport of any other scalar variable need to be discretized over the element-based control volume using well-established cell-centred techniques. The equations for the models used in this paper are as follows.

2.1. *Thermally driven flow*

The general equation governing heat transfer is

$$\frac{\partial(\boldsymbol{\rho}\mathbf{cT})}{\partial\mathbf{t}} + \nabla\cdot(\boldsymbol{\rho}\mathbf{c}\bar{\mathbf{u}}\mathbf{T}) = \nabla\cdot(\mathbf{k}\nabla\mathbf{T}) + \mathbf{S}_T \tag{4}$$

DOI: 10.1002/fld

The buoyancy forces are calculated using the Boussinesq approximation. This approximation results in a source per unit volume which enters the i' momentum equation as

$$\mathbf{S}_i = -\boldsymbol{\rho}\boldsymbol{\beta}\mathbf{g}_i(\mathbf{T} - \mathbf{T}_{\text{ref}}) \tag{5}$$

2.2. *Turbulent flow*

The k–ε model is employed for turbulent flow which uses the following equations for the solution of k and ε. The turbulent kinetic energy equation is given by [47]

$$\frac{\partial \boldsymbol{\rho}\mathbf{k}}{\partial \mathbf{t}} + \nabla \cdot (\boldsymbol{\rho}\mathbf{u}\mathbf{k}) = \nabla \cdot \left(\left[\mu_{\text{lam}} + \frac{\boldsymbol{\rho}\mathbf{v}_t}{\boldsymbol{\sigma}_k}\right]\nabla \mathbf{k}\right) + \boldsymbol{\rho}\mathbf{v}_t\mathbf{G} - \boldsymbol{\rho}\boldsymbol{\varepsilon} \tag{6}$$

and the dissipation rate equation is as follows:

$$\frac{\partial \boldsymbol{\rho}\boldsymbol{\varepsilon}}{\partial \mathbf{t}} + \nabla \cdot (\boldsymbol{\rho}\mathbf{u}\boldsymbol{\varepsilon}) = \nabla \cdot \left(\left[\mu_{\text{lam}} + \frac{\boldsymbol{\rho}\mathbf{v}_t}{\boldsymbol{\sigma}_\varepsilon}\right]\nabla \boldsymbol{\varepsilon}\right) + \mathbf{C}_1\boldsymbol{\rho}\mathbf{v}_t\mathbf{G}\frac{\boldsymbol{\varepsilon}}{\mathbf{k}} - \mathbf{C}_2\boldsymbol{\rho}\frac{\boldsymbol{\varepsilon}^2}{\mathbf{k}} \tag{7}$$

where the rate of generation of the turbulent kinetic energy, $\mathbf{G}$, is given by

$$\mathbf{G} = 2\left(\left[\frac{\partial \mathbf{u}}{\partial \mathbf{x}}\right]^2 + \left[\frac{\partial \mathbf{v}}{\partial \mathbf{y}}\right]^2 + \left[\frac{\partial \mathbf{w}}{\partial \mathbf{z}}\right]^2\right) + \left(\frac{\partial \mathbf{u}}{\partial \mathbf{y}} + \frac{\partial \mathbf{v}}{\partial \mathbf{x}}\right)^2 + \left(\frac{\partial \mathbf{u}}{\partial \mathbf{z}} + \frac{\partial \mathbf{w}}{\partial \mathbf{x}}\right)^2 + \left(\frac{\partial \mathbf{w}}{\partial \mathbf{y}} + \frac{\partial \mathbf{v}}{\partial \mathbf{z}}\right)^2 \tag{8}$$

The turbulent viscosity is related to k and ε by

$$\boldsymbol{\mu}_t = \boldsymbol{\rho}\mathbf{C}_\mu\frac{\mathbf{k}^2}{\boldsymbol{\varepsilon}} \tag{9}$$

The values of empiric constants employed in Equations (6)–(9) are the standard values [47]

$$C_\mu = 0.09, \quad \sigma_k = 1.0, \quad \sigma_\varepsilon = 1.3, \quad C_1 = 1.44, \quad C_2 = 1.92 \tag{10}$$

3. DISCRETIZATION APPROACH

In the CFVM method equations (2) and (3) are discretized 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), (6) and (7) is simply the mesh element. A detailed description of the vertex-based discretization process is given in Reference [48] and cell-centred discretization process is given in Reference [9].

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 Reference [36] who used them in structural analysis. The variables and co-ordinates are

DOI: 10.1002/fld

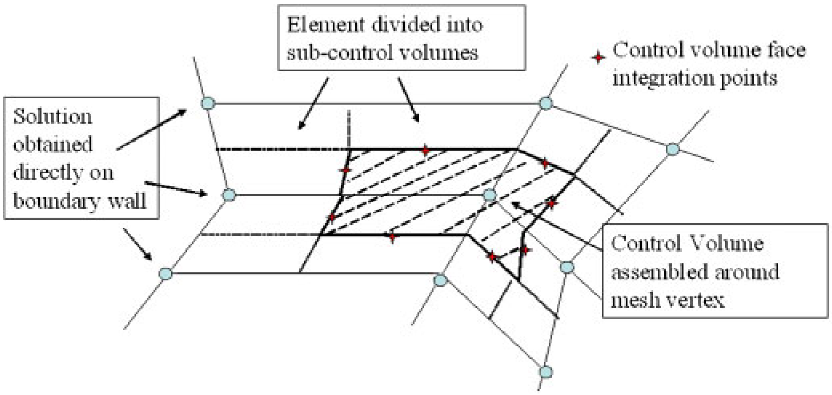
Element divided into sub-control volumes
Control volume face integration points
Solution obtained directly on boundary wall
Control Volume assembled around mesh vertex

Figure 1. Vertex-based control volume.

approximated, in local co-ordinates as

$$u_i(s,t,u)=\sum_{j=1}^{n} N_j(s,t,u)u_{ij}$$
$$p(s,t,u)=\sum_{j=1}^{n} N_j(s,t,u)p_j \tag{11}$$
$$x_i(s,t,u)=\sum_{j=1}^{n} N_j(s,t,u)x_{ij}$$

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. For a general variable, face gradients can easily be calculated using the derivatives of the interpolation functions, thus

$$\left(\frac{\partial \boldsymbol{\phi}}{\partial \mathbf{x}_i}\right)_f=\sum_{j=1}^{n}\frac{\partial \mathbf{N}_j}{\partial \mathbf{x}_i}\boldsymbol{\phi}_i \tag{12}$$

In the cell-centred discretization approach, the face derivatives require special treatment on a non-orthogonal mesh. If the mesh is orthogonal, vectors $\underline{v}$ and $\underline{n}$ in Figure 2 are parallel and the face derivative in the face normal direction is simply

$$\frac{\partial \boldsymbol{\phi}}{\partial \mathbf{n}}=\frac{\partial \boldsymbol{\phi}}{\partial \mathbf{v}} \tag{13}$$

On a non-orthogonal mesh the face normal vector can be written as a component in the $\underline{v}$ direction plus a tangential component of the form

$$\mathbf{n}=(\underline{\mathbf{v}}\cdot\underline{\mathbf{n}})\underline{\mathbf{v}}+\boldsymbol{\xi} \tag{14}$$

DOI: 10.1002/fld

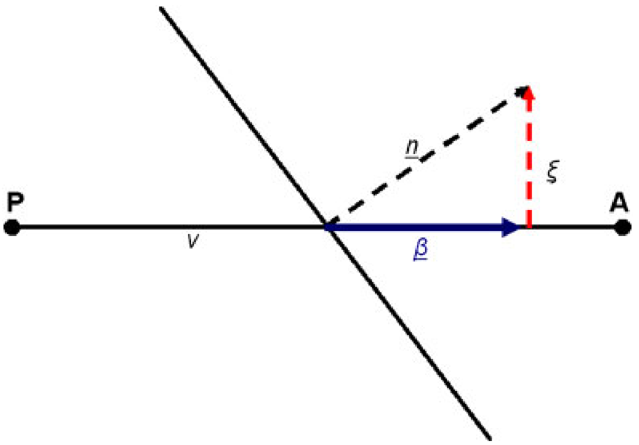

Figure 2. Non-orthogonal correction.

So it is possible to split the differential with respect to the normal in terms of an orthogonal contribution and a non-orthogonal correction term

$$\frac{\partial \boldsymbol{\phi}}{\partial \mathbf{n}} = \boldsymbol{\beta}\frac{\partial \boldsymbol{\phi}}{\partial \mathbf{v}} + \xi\frac{\partial \boldsymbol{\phi}}{\partial \mathbf{t}} \tag{15}$$

where the vector $\boldsymbol{\beta}$ is chosen to be parallel to $\mathbf{v}$ and $\mathbf{t}$ is a unit vector in the direction ξ.

Many possible approaches exist to satisfy Equation (15). Jasak [49] outlines and compares the most common approaches. The source code keeps the non-orthogonal correction as small as possible by making $\boldsymbol{\beta}$ and ξ orthogonal.

3.1. Pressure–velocity coupling

Potential problems can be encountered with CFD numerical methods that store both pressure and the components of velocity at the same location. Equal order methods that use linear interpolation can lead to checker boarding in the pressure and velocity fields. Several specialized interpolation schemes for co-located non-staggered arrangements were proposed in the early 1980s with small variations, the most widely used being Rhie Chow [50] which is employed within the host code, PHYSICA, for the cell-centred discretization method. There are unresolved problems with the Rhie Chow scheme. Croft [9] showed that the Rhie Chow interpolation method can lead to non-physical cell centred values despite satisfying continuity. Majumdar [51] and Choi [52] reported that the converged velocity field is slightly dependent on relaxation factor and time step size. Kawaguchi [53] reported that Rhie Chow interpolation could cause checkerboard pressure predictions. Prakash [33] developed a co-located equal-order method for the CVFE method which does not suffer from spurious checkerboard pressure fields. The key idea is similar to using a staggered grid. A new face velocity field is defined which is dependent on the pressure difference between adjacent nodal points. This new velocity field is substituted into the discretized continuity equation so that the generation of a checkerboard pressure field is avoided.

In the solution of the Navier–Stokes equations, the CFVM employs an adaptation of the revised SIMPLER method [54]. Correct pressure and velocity coupling is ensured by the method of Prakash and Patankar [33]. The discretized momentum equations (2) lead to a system of algebraic equations of the form

$$\mathbf{a}_i\mathbf{u}_i = \sum \mathbf{a}_{nb}\mathbf{u}_{nb} + \mathbf{b}_i - \nabla\mathbf{p}_i \tag{16}$$

where i denotes the component of velocity and nb its neighbours. In order to prevent checkerboarding in the final solution the discretization of either the momentum or continuity equations must ensure that face velocity values are defined in terms of adjacent pressure values. In co-located formulations the face velocity is not defined purely as a linear interpolation of adjacent cell values as this would lead to face values defined in terms of adjacent pressure values. Effectively the pressure gradient term resulting from linear interpolation of the adjacent cell values is removed and a new pressure gradient term, written in terms of adjacent pressure values, is added to prevent checkerboard pressure predictions. In the CFVM formulation it is the discretization of the continuity equation that prevents checkerboard pressure predictions.

The discretized momentum equation (16) can be rearranged as follows:

$$u_i = \frac{\sum_{nb} a_{nb}^{i_i}(u_i)_{nb} + b^{i_i}}{a_i} - \frac{(\partial p/\partial x_i)(\nabla V)_{cv}}{a_i} \tag{17}$$

A pseudo-velocity field can be defined from Equation (17) which is used in the discretization of the continuity equation, allowing the vertex-based face velocity to be written in terms of adjacent pressure values, as follows:

Pseudo-velocity field is defined as

$$\hat{u}_i = \frac{\sum_{nb} a_{nb}^{i_i}(u_i)_{nb} + b^{i_i}}{a_i} \tag{18}$$

and a pressure gradient coefficient as

$$(d^{u_i})_i = \frac{\nabla V}{a_i} \tag{19}$$

Using Equations (16)–(19) and from principles of conservation an equation can be defined for velocity on a control volume face (f) in terms of adjacent pressures. Here the face pressure gradient is written in terms of adjacent nodal pressures, using Equation (12), and a face pseudo-velocity and pressure gradient coefficient can be obtained from nodal values using interpolation functions, as Equation (11)

$$(u_i)_f = (\hat{u}_i)_f - (d^{u_i})_f \left(\frac{\partial p}{\partial x_i}\right)_f \tag{20}$$

The continuity equation (3) is discretized using finite volume methodology giving an equation for each vertex-based control volume

$$\frac{1}{\partial t}\sum_{\mathrm{scv}}(\rho_{\mathrm{scv}} - \rho_{\mathrm{scv}}^0)V_{\mathrm{scv}} + \sum_f \rho_f A_f (u_i \cdot n_i)_f = 0 \tag{21}$$

where f is a control volume face, ρ^0 the density at previous time step and scv is the sub-control volume. Equation (20) is then substituted into Equation (21) giving a discretized continuity equation in terms of adjacent pressure values which prevents spurious pressure oscillations

$$\frac{1}{\partial \mathbf{t}} \sum_{\text{scv}} (\boldsymbol{\rho}_{\text{scv}} - \boldsymbol{\rho}^{0}_{\text{scv}}) \mathbf{V}_{\text{scv}} + \sum_{f} \boldsymbol{\rho}_f \mathbf{A}_f \sum_{i=x,y,z} \left[(\hat{\mathbf{u}}_i)_f - (\mathbf{d}^{u_i})_f \left(\frac{\partial \mathbf{p}}{\partial \mathbf{x}_i} \right)_f \right] \cdot (\mathbf{n}_i)_f = 0 \tag{22}$$

and rearranging gives

$$\frac{1}{\partial t} \sum_{\text{scv}} (\rho_{\text{scv}} - \rho^{0}_{\text{scv}}) V_{\text{scv}} + \sum_{f} \rho_f A_f \sum_{i=x,y,z} (d^{u_i})_f \left(\frac{\partial p}{\partial x_i} \right)_f \cdot (n_i)_f = \sum_{f} \rho_f A_f ((\hat{u}_i)_f \cdot (n_i)_f) \tag{23}$$

3.2. Coupled finite volume method

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 [48]. As mass conservation is enforced over the vertex-based control volume, any errors resulting from interpolating for element face values also decrease.

The components of velocity are solved and stored at the mesh vertices. These are located at the corners of the mesh element that forms a CC-FVM control volume. Element face velocity components, required for the cell-centred discretization of Equations (4), (6), (7) and (10) can be easily extrapolated from face corner values, thus

$$(u_i)_f = \sum_{n=1}^{N} \frac{(u_i)_n}{N} \tag{24}$$

where N is the number of corner vertices of the element face. It is this element face velocity that is used to calculate the element mass flux in the convection of a transported scalar.

4. 3D CODE VERIFICATION

Verification of the 3D code has been carried out against Ethier and Steinman's [55] analytical benchmark solutions for flow. The manufactured solution will likely never be physically realized but it is an excellent test case since the solution can be imposed on arbitrarily shaped finite domains by the specification of analytic boundary conditions. The flow field is multidirectional and complex, providing a challenging case for quantifying the effects of mesh distortion. Figure 3 shows the flow field and meshes employed in the simulations. The mesh is centred at (0,0,0) and extended 1 unit in all directions. The cubed domain, mesh 1, was discretized using a uniform nodal spacing from 0.5 to 0.14, giving 6 meshes of varying density. Mesh 1 was then distorted by various degrees, mesh 2 and mesh 3. Mesh 2 resulted in the mesh containing elements with angles ranging from 26.57° to 151.9°. Mesh 3 was more severely distorted with element angles ranging from 3.45° to 176.6°. The analytical solutions used for the Beltrami flow and pressure

DOI: 10.1002/fld

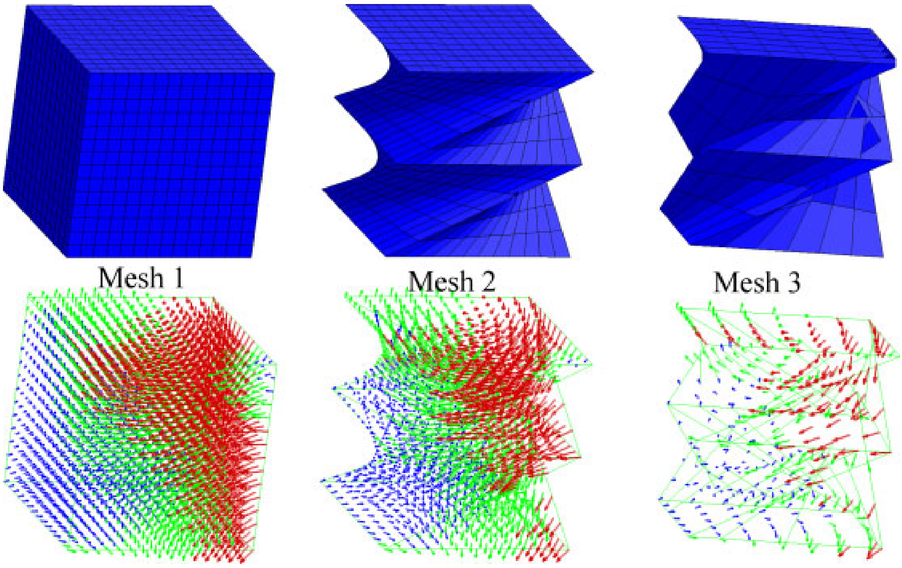

Figure 3. Meshes used for Beltrami flow.

field are as follows:

$$
\begin{aligned}
\mathbf{u} &= -\mathbf{a}[\mathbf{e}^{ax}\sin(\mathbf{ay}\pm\mathbf{dz}) + \mathbf{e}^{az}\cos(\mathbf{ax}\pm\mathbf{dy})]\,\mathbf{e}^{-d^2t}\\
\mathbf{v} &= -\mathbf{a}[\mathbf{e}^{ay}\sin(\mathbf{az}\pm\mathbf{dx}) + \mathbf{e}^{ax}\cos(\mathbf{ay}\pm\mathbf{dz})]\,\mathbf{e}^{-d^2t}\\
\mathbf{w} &= -\mathbf{a}[\mathbf{e}^{az}\sin(\mathbf{ax}\pm\mathbf{dy}) + \mathbf{e}^{ay}\cos(\mathbf{az}\pm\mathbf{dx})]\,\mathbf{e}^{-d^2t}\\
\mathbf{p} &= -\frac{\mathbf{a}^2}{2}\left[\begin{array}{l}\mathbf{e}^{2ax}+\mathbf{e}^{2ay}+\mathbf{e}^{2az}+2\sin(\mathbf{ax}\pm\mathbf{dy})\cos(\mathbf{az}\pm\mathbf{dx})\,\mathbf{e}^{a(y+z)}\\ \quad +2\sin(\mathbf{ay}\pm\mathbf{dz})\cos(\mathbf{ax}\pm\mathbf{dy})\,\mathbf{e}^{a(z+x)}\\ \quad +2\sin(\mathbf{az}\pm\mathbf{dx})\cos(\mathbf{ay}\pm\mathbf{dz})\,\mathbf{e}^{a(x+y)}\end{array}\right]\mathbf{e}^{-2d^2t}
\end{aligned}
\tag{25}
$$

In all tests the independent constants in Equations (25) were set as $a=\pi/4$ and $d=\pi/2$. The initial velocity field at $t=0$ was set by Equations (25), resulting in velocities ranging from 1.59 to -3.31 with a 22% decay after time $t=0.1$. Dirichlet boundary conditions, also based on Equations (25) were applied to all external faces. Figure 4 shows a graph of the normalized L_2 error, defined as

$$
\text{error} = \frac{\|\mathbf{u}-\mathbf{u}_{\text{exact}}\|_{L_2}}{\|\mathbf{u}_{\text{exact}}\|_{L_2}} \tag{26}
$$

On the orthogonal mesh the vertex-based code resulted in a velocity error in the order of 10^{-4} for the fine mesh and 10^{-3} for the coarser meshes. The cell-centred code resulted in a slightly larger error of 10^{-2} for the velocity field and 10^{-1} for the pressure field. The cell-centred discretization results in a weaker formulation, as the solution points are located internal to the boundary and the analytic boundary conditions enter the solution in the source term. On the distorted mesh 2, the

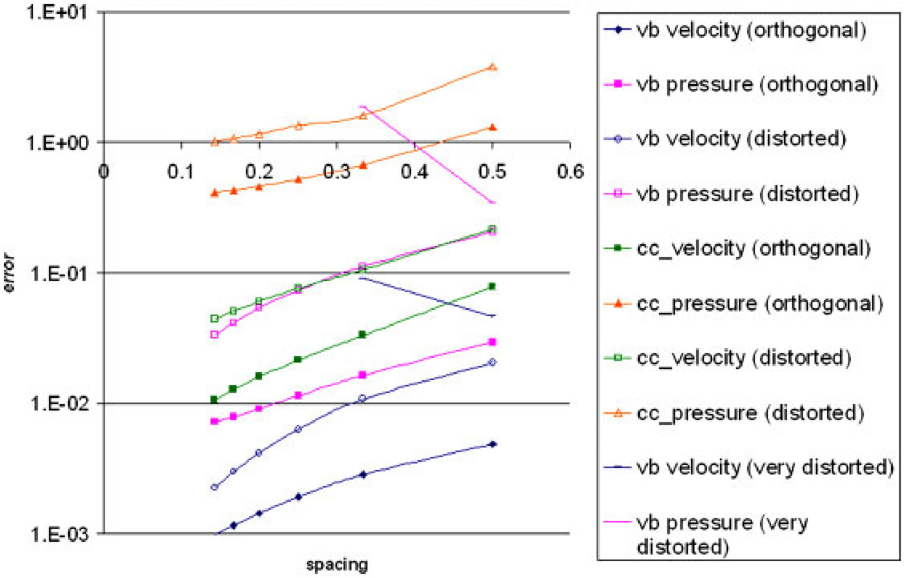

Figure 4. Beltrami flow: error solving vertex-based (vb) and cell-centred (cc).

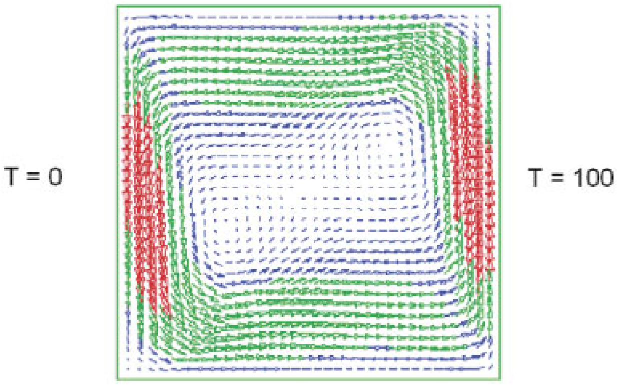

Figure 5. Thermally driven flow, Rayleigh number 10^5.

vertex-based error was less than the error obtained with the cell-centred code on the orthogonal mesh. The cell-centred formulation failed to converge on mesh 3, whereas the vertex-based method gave solutions on the coarse mesh with element angles ranging from 9.5° to 170.5° with an error of the order of 10^{-2} for the velocity field and 10^{-1} for the pressure field. As the density of mesh 3 was increased, the minimum element angle reduced and the vertex-based code failed to converge.

This verification case illustrates the ability of the vertex-based flow code to handle distortion in a mesh with greater accuracy than the cell-centred approach. Although the vertex-based code has a breaking point, in this case with element angles below 9.5°, the method continues to give

DOI: 10.1002/fld

solutions beyond the non-orthogonal capabilities of the cell-centred code. The question here is at what cost? On a mesh comprising of 3375 nodes and 2744 elements, the memory requirements per solution point for the flow variables were 879 bytes for the vertex-based code and 249 bytes for the cell-centred code. The computer time per iteration per solution point was 7.35×10^{-5} s and 2.00×10^{-5} s for vertex-based and cell-centred, respectively. In summary the vertex-based required approximately 3.5 times more memory than the cell-centred code and approximately 4 times more compute time for simulations with similar convergence. The number of iterations required to achieve convergence is case dependent. The extra memory required for the vertex-based code can be explained by the increase in stored face quantities. On a hexahedral mesh the vertex-based control volume comprises of 24 faces compared to 6 cell-centred. This increased discretization requires greater computational effort and produces a solution matrix with a possible 27 non-zero coefficients compared to a possible 7 non-zero coefficients in the cell-centred discretized equation.

5. EVALUATION OF THE METHOD

Results have been evaluated on a number of benchmark problems, including thermally driven flow, where the effect of mesh distortion is investigated and turbulent flow, where the effect of mesh resolution is investigated. The approach is then applied to three-dimensional turbulent flow over an aircraft wing. Results are compared for a uniform mesh and distorted versions of the same mesh.

5.1. Thermally driven flow

De Vahl Davis [56] 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, see Figure 5. The simulations were performed on a Athlon 1.39 Ghz processor for a uniform 35×35 Cartesian mesh, mesh 1, and distorted versions of mesh 1, shown in Figure 6. 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. For a Rayleigh number of 10^3, Figure 7 shows the velocity plots for both the CFVM and CC-FV method. As can be seen from these plots the error due to mesh distortion is much smaller in the CFVM results. On the distorted meshes the CFVM slightly under predicts the maximum value due to the coarseness of the mesh in this region

Table I shows $U_{\max}$ and $V_{\max}$, the maximum value of the normalized velocity component along the central planes for mesh 1. The $Y_{\max}$ and $X_{\max}$, are the normalized positions of this maximum value. The percentage errors of the simulation results against benchmark solutions are shown in brackets for each Rayleigh number (Ra). A measure of the error, due to mesh distortion, for meshes 2 and 3 is shown in Table II, using mesh 1 as the base result. Divergence was encountered on mesh 3 for a Rayleigh number of 10^6. Good agreement with benchmark solutions were obtained on the uniform mesh and the solutions only slightly degraded on the distorted meshes (Figure 8).

DOI: 10.1002/fld

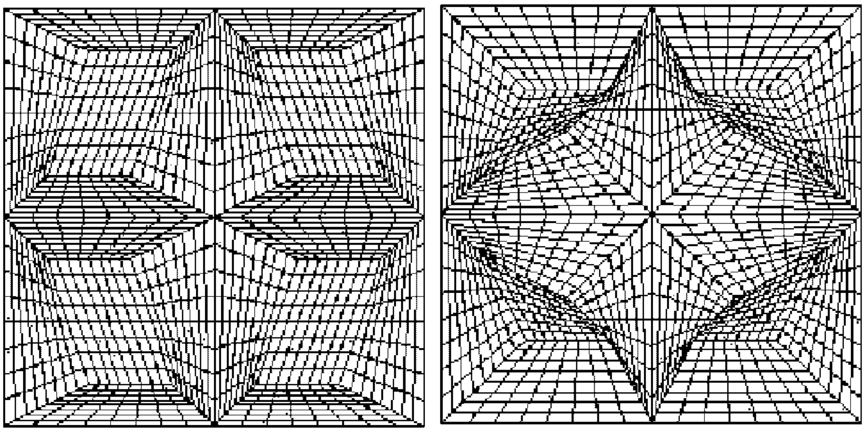

Figure 6. Distorted versions of Cartesian mesh 1, mesh 2 (left), mesh 3 (right).

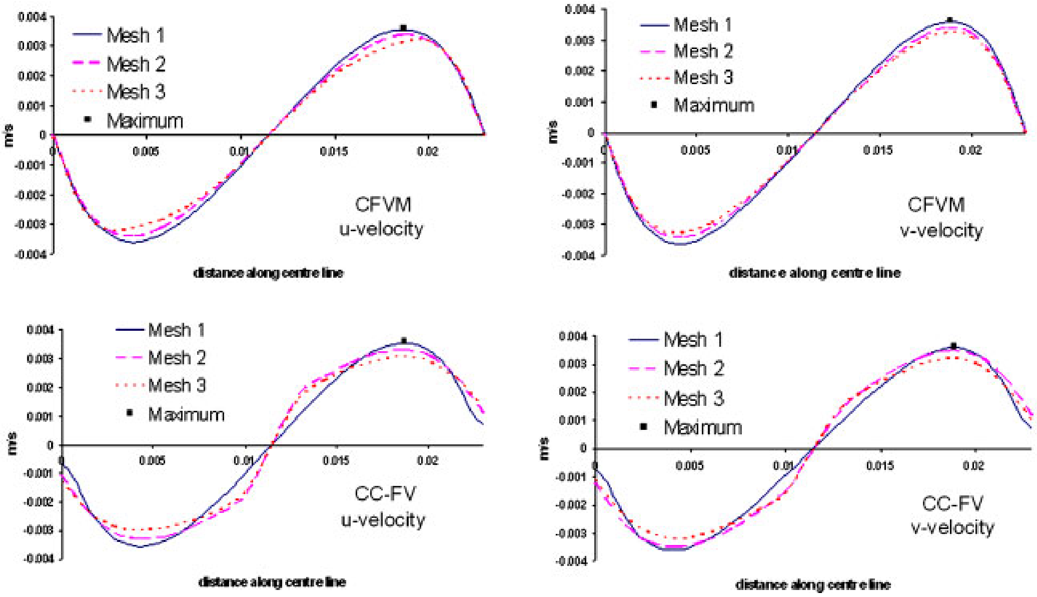

Figure 7. Velocity plots for Rayleigh number of 10^3.

5.2. *Turbulent flow—backward facing step*

In this case fluid enters the domain through a channel of height H and flows over a step of height h into a channel of height $(H + h)$. The case simulated here employs an outflow channel that is three times the height of the step, where the step height is set to 0.1 m. The geometry and boundary conditions are shown in Figure 9, where $H = 2h$. The material properties are set to those of air and the inflow velocity U_{in} is varied to give the required Reynolds number. A no-slip boundary

DOI: 10.1002/fld

Table I. Percentage error with benchmark values.

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 II. Error due to mesh distortion.

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

condition $u = v = 0.0$ is applied to the wall. The outflow boundary location is positioned 20 step heights downstream of the step, to limit its influence on the simulation. Pressure is set to zero at the outlet. The inflow boundary is positioned five step heights upstream of the step to allow the formation of a fully turbulent velocity profile before the step is reached. Positioning the inlet some distance away from the step avoids having precise inlet conditions for the turbulent quantities. The inflow turbulent quantities were estimated as

$$\begin{aligned} K_{\text{in}} &= \tfrac{1}{4}\,0.018U_{\text{in}}^2 \\ \varepsilon_{\text{in}} &= \frac{0.1643k_{\text{in}}^{1.5}}{0.09h} \end{aligned} \tag{27}$$

As the fluid flows over the step it detaches from the wall and reattaches on the bottom wall at a distance x_R from the step. It is this reattachment length x_R that is commonly used for validation purposes. The experimental data for this case [57] indicates a reattachment length of approximately $(7.0 \pm 0.5)h$ with only slight variation for flows of differing Reynolds numbers. The standard k–ε model has been shown by many authors, including Nallasamy and Chen [58] to underpredict the reattachment point, giving a value of x_R within the range 5.8–6.1 step heights along the bottom wall.

The simulations were performed initially on a mesh consisting of 5200 elements. The u-velocity at the inlet was set to give Reynolds numbers of 30 000, 50 000, 70 000 and 90 000. For all the Reynolds numbers investigated, the reattachment length, x_R, is at a distance of 6 step heights

DOI: 10.1002/fld

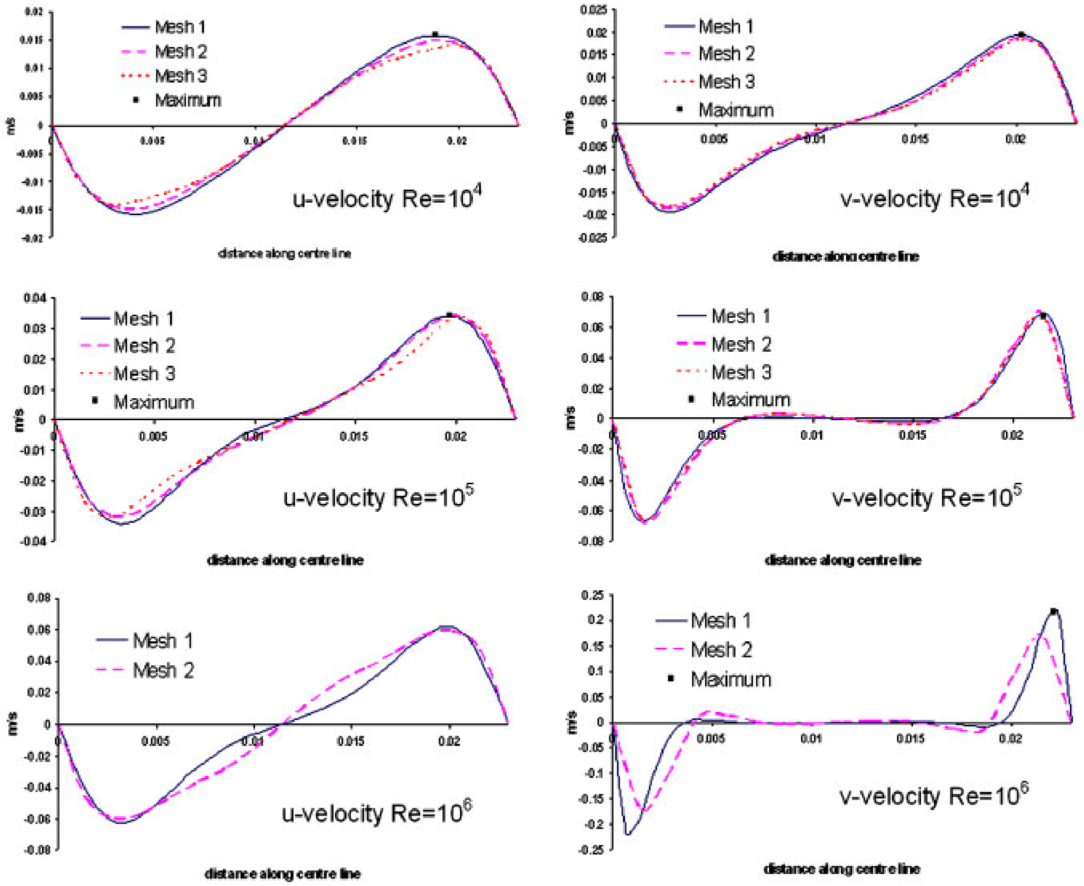

Figure 8. Velocity plots for CFVM, Rayleigh numbers 10^4, 10^5 and 10^6.

downstream. This value is well within the expected range. There appears to be very little difference in the results of the CFVM and CC-FVM discretization methods. To investigate the effect of mesh resolution simulations were also performed on coarser meshes of 1400 and 3588 elements for a Reynolds number of 50 000. As can be seen from the plots of the wall u-velocity values, Figure 10, the CFVM method still gave good predictions of the reattachment point, $x_R = 5.7h$ on the 1400 element mesh and, $x_R = 5.8h$ on the 3588 element mesh. Whereas, the purely cell-centred method under-predicted x_R on the coarser meshes, $x_R = 5.0h$ and $x_R = 5.2h$ for the 1400 and 3588 element meshes, respectively. Simulations were also performed on a mesh containing regions of distorted quadrilateral and triangle elements. An angle test was performed on the elements to measure the mesh quality, an ideal angle is dependent on the element type, ideally a quadrilateral element should have 4 angles of 90° whereas a triangle element should have 3 angles of 60°. An acceptable angle for a triangle element was set as; minimum angle of 35° and maximum angle of 145°, for a quadrilateral element as; minimum angle of 30° and maximum angle of 150°. Twenty percent of the elements in the mesh failed the quality test with angles ranging from 10.6° to 170.7°.

DOI: 10.1002/fld

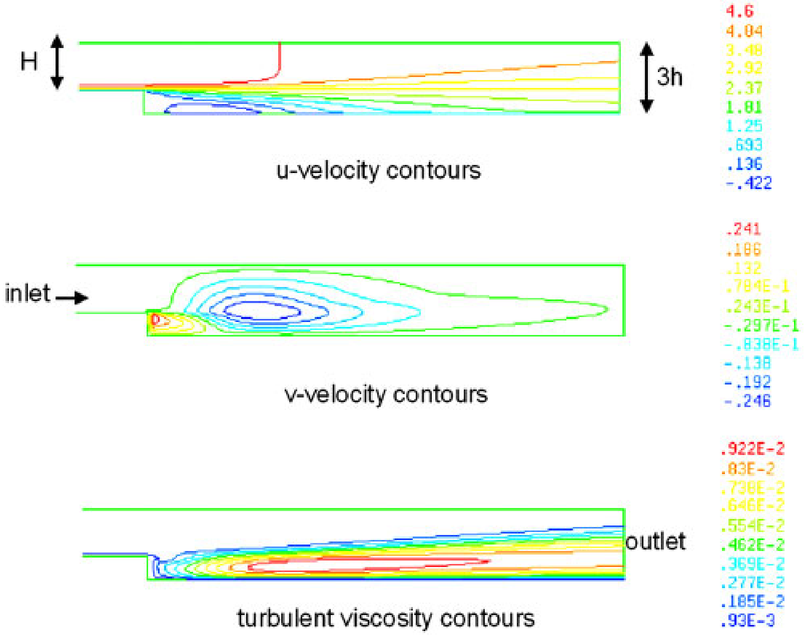

Figure 9. Geometry, boundary conditions and turbulent flow results.

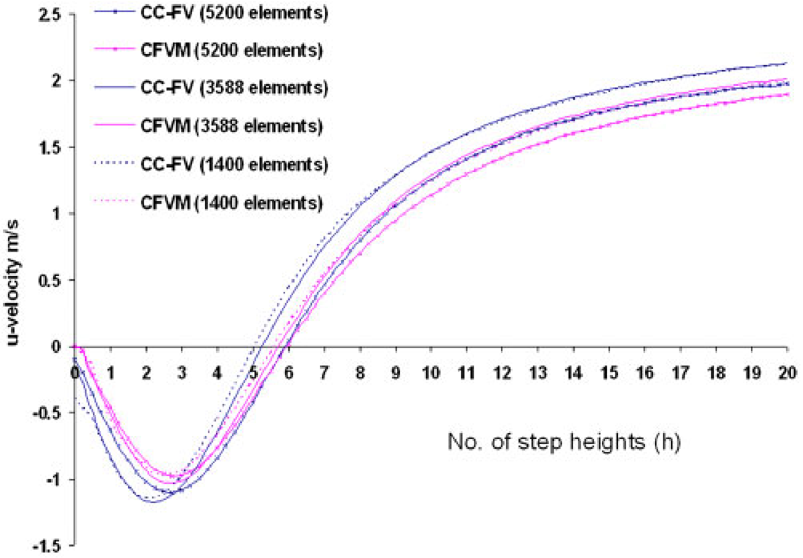

Figure 10. u-velocity along wall.

DOI: 10.1002/fld

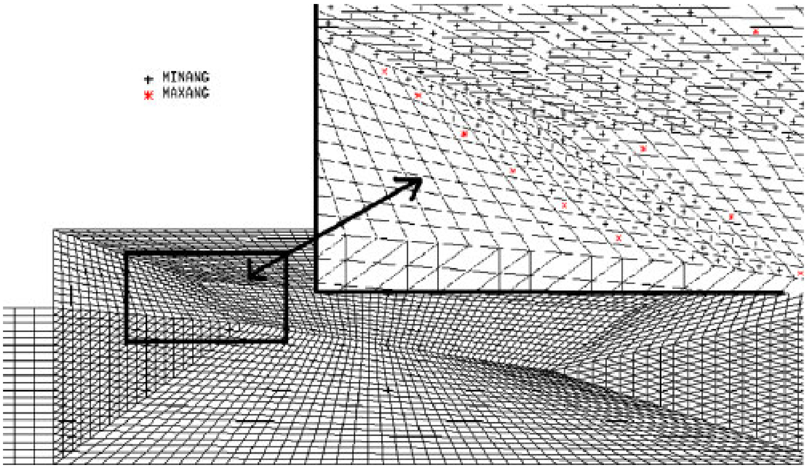

Figure 11. Unstructured distorted mesh.

Figure 11 shows a section of the mesh in the step expansion region, the enlarged area shows an example of bad quality mesh. Contour plots of the resultant velocity field on the distorted unstructured mesh, are shown in Figure 12 for CC-FVM and CFVM, respectively. The solutions were considered converged when the L_2 norm of the change in the solution of all the variables fell to 10^{-4}. Employing the CC-FVM failed to give converged solutions and oscillating residuals were encountered. The solutions obtained degraded in regions of bad mesh quality. The CFVM gave converged solutions with minimal influence of the mesh quality.

5.3. Turbulent flow over a three-dimensional aircraft wing

Finally the CFVM is employed in the simulation of flow over an aircraft wing. The geometry of the wing was taken from ONERA M6 specifications [59]. Many authors have investigated transonic flow over the ONERA M6 wing, at Mach numbers of 0.7 and above, i.e. supersonic flow, shock and turbulent boundary layer separation at various angles of attack. It is beyond the scope of this paper to investigate the complexities of transonic flow, the simulations carried out employ low-speed Mach number of 0.3. The aim of the simulations is to illustrate the CFVM on a three-dimensional problem and investigate its performance when the mesh is distorted. The ONERA M6 is a swept, semi-span wing with no twist. The leading-edge sweep is 30°, trailing edge sweep 15.8° 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, Figure 13. 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 CFVM on a 'real-life' distorted mesh.

At the outset it is worth saying that on the C-mesh, the CFVM method and the conventional CC discretization results are very similar in most respects. However, as the mesh is degraded the purely CC method fails, whereas the CFVM continues to produce solutions. A mesh quality

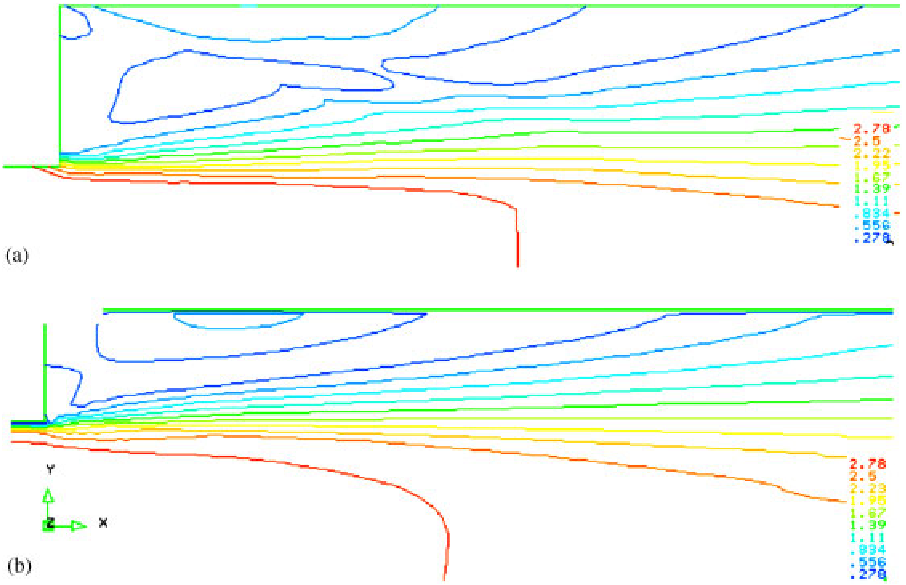

Figure 12. Resultant velocity contour plots on distorted mesh: (a) CC-FVM; and (b) CFVM.

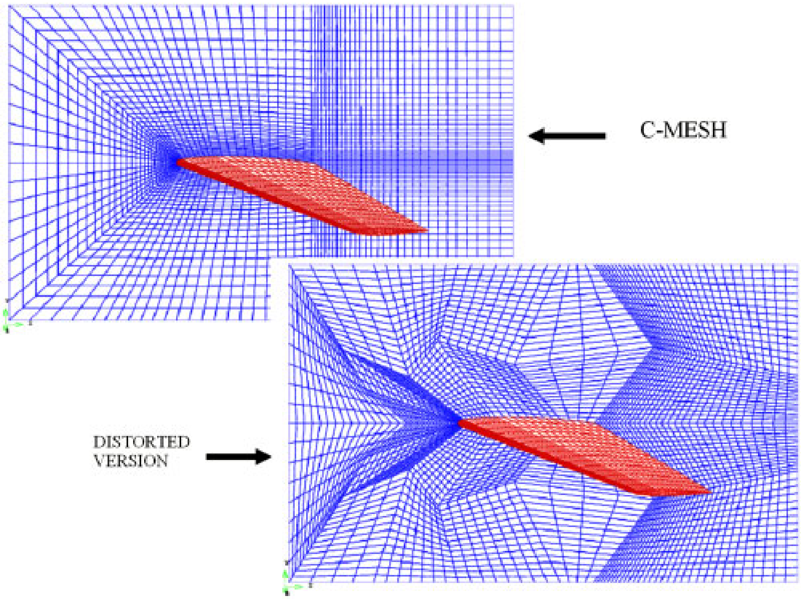

Figure 13. C-mesh and distorted version C-mesh.

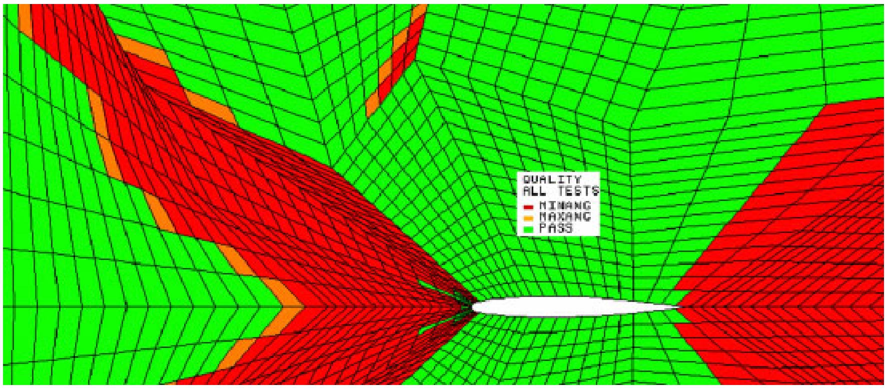

Figure 14. Section of the distorted C-mesh showing mesh quality.

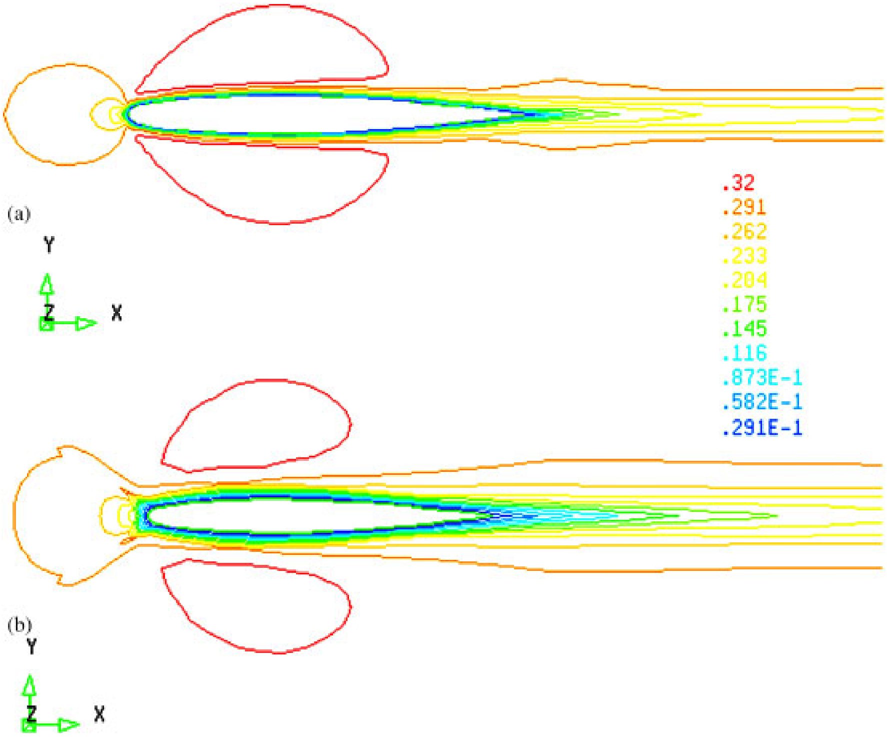

Figure 15. Mach number contour plots on plane $z = 0.558$: (a) C-mesh; and (b) distorted C-mesh.

test was performed on the distorted mesh and of the 101 412 elements 26 100 were considered to be of poor quality with angles below 30° or above 150°, with angles ranging from 3° to 178°. Figure 14 shows a section of the distorted mesh showing elements with angles below a minimum value of 30° or above a maximum value of 150°. The results are shown for the CFVM

DOI: 10.1002/fld

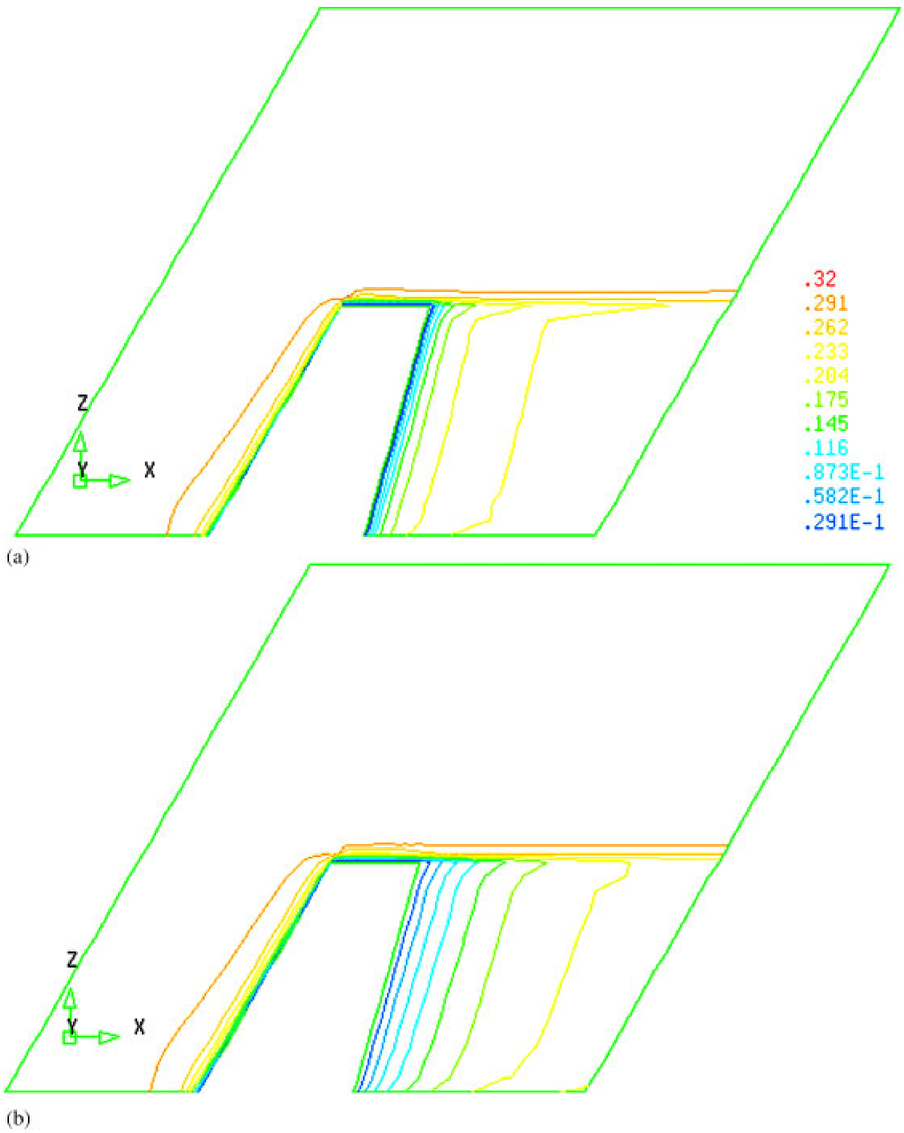

Figure 16. Mach number contour plots on plane $y = 0$: (a) C-mesh; and (b) distorted C-mesh.

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 15 and 16 show the mach contour plots for the z- 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 [0–107 m/s] and [−6.37–39 m/s], respectively. The minimum and maximum v-velocity values decreased slightly from [−49.3 m/s, 49.3 m/s] to [−40.4 m/s, 40.4 m/s] on the distorted mesh.

DOI: 10.1002/fld

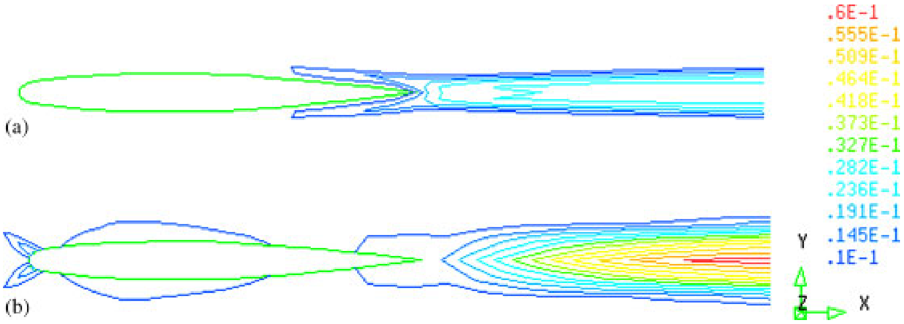

Figure 17. Viscosity contour plots on plane $z = 0.558$: (a) C-mesh; and (b) distorted C-mesh.

The turbulent viscosity contours, Figure 17, 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 sharp rear of the wing shape (McBride, 2003) and can be eliminated with a more careful approximation. The maximum turbulent viscosity obtained on the C-mesh was $0.03\,\mathrm{m}^2/\mathrm{s}$ compared to $0.06\,\mathrm{m}^2/\mathrm{s}$ on the distorted mesh.

5.3.1. Run times and memory requirements. The simulations were performed on a Pentium 4 CPU 2.54 GHz. The mesh employed comprised of 101 412 elements and 108 314 vertices. To achieve the convergence criteria that the L_2 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 3.3×10^{-5} s for each variable being solved vertex-based and 7.0×10^{-6} s for each variable being solved cell-centred. This gives a time per iteration of 15.72 s for CFVM solutions and 4.26 s 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.

6. CONCLUSIONS

The coupling of the vertex-based discretization method with well established cell-centred methods for CFD analysis of flow processes has been presented. Employing the vertex-based technique to obtain good resolution of the flow field has been shown to enable well established cell-centred algorithms to be employed in the solution of other transported quantities (e.g. thermal, turbulent variables). The cell-centred discretization of the transported quantities still includes non-orthogonal errors that may in turn introduce some error into the flow field. However, it is encouraging that the non-orthogonal errors do not appear to significantly affect the final solution. Comparison with benchmark problems indicate that the method degrades only slightly on highly distorted meshes and handles areas of poor mesh resolution reasonably well. The method can be applied to any kind of mixed polyhedral mesh up to hexahedral elements. It has the potential to deliver solutions

DOI: 10.1002/fld

on extremely complex geometries with meshes that necessarily have some poor quality elements and so are beyond the capabilities of conventional cell-centred based CFD analysis.

Although the CFVM is approximately 4 times more expensive in compute time than the CC-FV method, the coupled method does enable solutions on extremely complex geometries with meshes of poor quality making this approach a particularly useful CFD tool. However, with the flexibility of using meshes which tolerate elements with poor quality then one might argue, that a reduced mesh is required which will compensate for any increased compute time per node. In the form described here, the CFVM method can be straightforwardly used as an 'add in' on existing CC-FV software, exploiting all existing models that have been developed with in this context.

REFERENCES

1. Zienkiewicz OC, Morgan K. *Finite Elements and Approximation*. Wiley: New York, 1983.
2. Barth TJ. Aspects of unstructured grids and finite-volume solvers for the Euler and Navier–Stokes equations. *AGARD Report*, 1992; **787**:6.1–6.61.
3. Peraire J, Peiro J, Morgan K. Multigrid solution of the 3-D compressible Euler equations on unstructured tetrahedral grids. *International Journal for Numerical Methods in Engineering* 1994; **36**:1029–1044.
4. Appa J. *FLITE3D Version 4.0 User Guide*. British Aerospace Plc: Sowerby Research Centre, Filton, Bristol, U.K., 1997.
5. CFX, see www.ansys.com
6. FLUENT, see www.fluent.com
7. STAR-CD, see www.cd-adapco.com
8. PHYSICA, see www.physica.co.uk
9. Croft TN. Unstructured mesh-finite volume algorithms for swirling, turbulent, reacting flows. *Ph.D. Thesis*, The University of Greenwich, 1998.
10. Peric M. A finite volume method for the prediction of three-dimensional fluid flow in complex ducts. *Ph.D. Thesis*, University of London, 1985.
11. Moulinec C, Wesseling P. Colocated schemes for the incompressible Navier–Stokes equations on non-smooth grids for two-dimensional problems. *International Journal for Numerical Methods in Fluids* 2000; **32**:349–364.
12. Barth TJ, Jespersen DC. The design and application of upwind schemes on unstructured meshes. *AIAA Paper 89-0366*, 1989.
13. Weiss JM, Maruszewski JP, Smith WA. Implicit solution of preconditioned Navier–Stokes equations using algebraic multigrid. *AAIA Journal* 1999; **37**:29–36.
14. Lehnhauser T, Schafer M. Improved linear interpolation practice for finite-volume schemes on complex grids. *International Journal for Numerical Methods in Fluids* 2002; **38**:625–645.
15. Demirdzic I. A finite volume method for the computation of fluid flow in complex geometries. *Ph.D. Thesis*, University of London, 1982.
16. Braaten M, Shyy W. A study of recirculating flow computation using boundary-fitted co-ordinates: consistency aspects and mesh skewness. *Numerical Heat Transfer* 1986; **9**:559–574.
17. Peric M. Analysis of pressure–velocity coupling on nonorthogonal grids. *Numerical Heat Transfer* 1990; **17**(Part B):63–82.
18. Cho MJ, Chung MK. New treatment of nonorthogonal terms in the pressure correction equation. *Numerical Heat Transfer* 1994; **26**(Part B):133–145.
19. Lehnhauser T, Schafer M. Efficient discretisation of pressure-correction equations on non-orthogonal grids. *International Journal for Numerical Methods in Fluids* 2003; **42**:211–231.
20. Zhu M, Shimizu Y, Nishimoto N. Calculation of curved open channel flow using physical curvilinear non-orthogonal co-ordinates. *International Journal for Numerical Methods in Fluids* 2004; **44**:55–70.
21. Lyra PRM, Morgan K, Peraire J, Peiro J. TVD algorithms for the solution of the compressible Euler equations on unstructured meshes. *International Journal for Numerical Methods in Fluids* 1994; **19**:827–847.
22. Crumpton PI, Giles MB. Aircraft computations using multigrid and an unstructured parallel library. *AIAA 95-0210*, 1995.
23. Sorensen KA, Hassan O, Morgan K, Weatherill NP. An agglomeration unstructured hybrid mesh method for 2D turbulent compressible flows. *ISCFD*, Bremen, 1999.

DOI: 10.1002/fld

24. Coirier WJ. An adaptively-refined, Cartesian, cell-based scheme for the Euler and Navier–Stokes equations. *NASA*, TM-106754, 1994.
25. Mavriplis DJ. Three-dimensional multigrid Reynolds-averaged Navier–Stokes solver unstructured meshes. *AIAA Journal* 1995; **33**:445–453.
26. Haselbacher A, Blasek J. On the accurate and efficient discretization of the Navier–Stokes equations on mixed grids. *AIAA Journal* 2000; **38**:2094–2102.
27. Jameson A, Baker TJ, Weatherill NP. Calculation of inviscid transonic flow over a complete aircraft. *AIAA 86-0103*, 1986.
28. Chakrabartty SK. Vertex-based finite-volume solution of the two-dimensional Navier–Stokes equations. *AIAA Journal* 1990; **28**(10):1829–1831.
29. Boivin S, Cayré F, Hérard J-M. A finite volume method to solve the Navier–Stokes equations for incompressible flows on unstructured meshes. *International Journal of Thermal Sciences* 2000; **39**:806–825.
30. Perron S, Boivin S, Hérard J-M. A new finite volume discretisation scheme to solve 3D incompressible thermal flows on unstructured meshes. *International Journal of Thermal Sciences* 2004; **43**:833–848.
31. Slone AK, Pericleous K, Bailey C, Cross M, Bennett C. A finite volume unstructured mesh approach to dynamic fluid-structure interaction: an assessment of the challenge of predicting the onset of flutter. *Applied Mathematical Modelling* 2004; **28**:211–239.
32. Peric M. Simulation of flows in complex geometries: new meshing and solution methods. *NAFEMS Seminar*; *Simulation of Complex Flows* (*CFD*)—*Applications and Trends*. Niedernhausen/Wiesbaden, Germany, 2004.
33. Prakash C, Patankar SV. A control volume-based finite-element method for solving the Navier–Stokes equations using equal-order velocity-pressure interpolation. *Numerical Heat Transfer* 1985; **8**:259–280.
34. Baliga BR. *Advances in Numerical Heat Transfer*, vol. 1. Hemisphere Publishing Corporation: New York, 1996.
35. Reyes M, Rincon J, Damia J. Simulation of turbulent flow in irregular geometries using a control-volume finite-element method. *Numerical Heat Transfer* 2001; **39**(Part B):79–89.
36. Taylor GA, Bailey C, Cross M. A vertex-based finite volume method applied to non-linear material problems in computational solid mechanics. *International Journal for Numerical Methods in Engineering* 2003; **56**:507–529.
37. Dervieux A. Steady Euler simulations using unstructured meshes. *VKI Lecture Series*, 1884-04, 1985.
38. Hallo L, Le Ribault C, Buffat M. An implicit mixed finite-volume-finite-element method for solving 3D turbulent compressible flows. *International Journal for Numerical Methods in Fluids* 1997; **25**:1241–1261.
39. Koobus B, Farhat C. A variational multiscale method for the large eddy simulation of compressible turbulent flows on unstructured meshes—application to vortex shedding. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:1367–1383.
40. Durlofsky LJ. A triangle based mixed finite-element-finite volume technique for modelling two phase flow through porous media. *Journal of Computational Physics* 1993; **105**:252–266.
41. Mazzia A, Putti M. High order Godunov mixed methods on tetrahedral meshes for density driven flow simulations in porous media. *Journal of Computational Physics* 2005; **208**:154–174.
42. Chavent G, Younes A, Ackerer P. On the finite volume reformulation of the mixed finite element method for elliptic and parbolic PDE on triangles. *Computer Methods in Applied Mechanics and Engineering* 2003; **192**:655–682.
43. Wapperom P, Webster MF. A second-order hybrid finite-element-volume method for viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics* 1998; **79**(2–3):405–431.
44. Chan AJ, Kallinderis Y. Adaptive hybrid (prismatic–tetrahedral) grids for incompressible flows. *International Journal for Numerical Methods in Fluids* 1998; **26**:1085–1105.
45. McBride D, Croft TN, Cross M. Finite volume method for the solution of flow on distorted meshes. *International Journal for Numerical Methods for Heat and Fluid Flow* 2006, in press.
46. Croft N, Pericleous KA, Cross M. PHYSICA: a multiphysics environment for complex flow processes. In *Numerical Methods in Laminar and Turbulent Flow'95*, vol. 2, Taylor C, Durbetaki P (eds). 1995; 1269–1280.
47. Launder BE, Spalding DB. The numerical computation of turbulent flows: a review. *Computers and Fluids* 1974; **15**(2):269–289.
48. McBride D. Vertex-based discretisation methods for thermo-fluid flow in a finite volume-unstructured mesh context. *Ph.D. Thesis*, The University of Greenwich, 2003.
49. Jasak H. Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows. *Ph.D. Thesis*, The University of London, 1996.
50. Rhie CM, Chow WL. Numerical study of the turbulent flow past an aerofoil with trailing edge separation. *AIAA Journal* 1983; **21**(11):1525–1532.

51. Majumdar S. Role of underrelaxation of momentum interpolation for flow with nonstaggered grids. *Numerical Heat Transfer* 1988; **13**:125–132.
52. Choi SK. Note on the use of momentum interpolation method for unsteady flows. *Numerical Heat Transfer* 1999; **36**(Part A):545–550.
53. Kawaguchi BY, Tao W, Wzoe H. Checkerboard pressure predictions due to the underrelaxation factor and time step size for a nonstaggered grid with momentum interpolation method. *Numerical Heat Transfer* 2002; **41**(Part B):85–94.
54. Patankar SV. *Numerical Heat Transfer and Fluid Flow*. Hemisphere Publishing Corporation: New York, 1980.
55. Ethier CR, Steinman DA. Exact fully 3D Navier–Stokes solutions for benchmarking. *International Journal for Numerical Methods in Fluids* 1994; **19**:369–375.
56. De Vahl Davis G, Jones IP. Natural convection in a square cavity: a comparison exercise. *International Journal for Numerical Methods in Fluids* 1983; **3**:227–248.
57. Kim J, Kline SJ, Johnston JP. Investigation of a reattachment turbulent shear layer: flow over a backward-facing Step. *Transaction of the Journal of Fluids Engineering* (ASME) 1980; **102**:302–308.
58. Nallasamy M, Chen CP. Studies on effects of boundary conditions in confined turbulent flow predictions. *Technical Report CR-3929*, NASA, 1985.
59. Schmitt V, Charpin F. Pressure Distributions on the ONERA-M6-wing at transonic mach numbers. Experimental data base for computer program assessment. *Report of the Fluid Dynamics Panel Working Group 04*, AGARD AR, 1979.

DOI: 10.1002/fld