A novel Cartesian CFD cut cell approach

Mark W. Johnson*

School of Engineering, University of Liverpool, Liverpool L69 3GH, UK

A R T I C L E   I N F O

Article history:
Received 22 October 2011
Received in revised from 14 February 2013
Accepted 22 March 2013
Available online 2 April 2013

Keywords:
CFD
Cut cell
Cartesian mesh

A B S T R A C T

A new approach to evaluating the fluxes on the faces of Cartesian cut cells is presented. In contrast to most established techniques where the value of a flow variable is only defined at specific points within a cell, in the new approach, the variable values are defined throughout the cell using spline functions. This ensures the first derivatives of the flow variables are continuous throughout the computational domain which is not achieved with existing finite volume methods. For cut cells the values of the variables along the cut faces are therefore defined and hence the mass and momentum fluxes are readily evaluated to second order accuracy. The method is validated against three test cases where analytical solutions or extensive experimental and computational data exists. The test cases are flow in an annulus, flow around an isolated cylinder and laminar boundary layer development on a flat plate inclined at a range of angles to the gridlines. For these test cases, the flow is accurately resolved in the cut cells. The method therefore realises the potential of Cartesian CFD as a more efficient computational tool than more commonly used body fitted methods.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

The majority of CFD calculations are currently performed on body fitted meshes, which provide good resolution of the boundary layers providing an adequately fine grid is used and suitable cell types are used in the near wall region. The definition of a suitable grid for a complex engineering geometry can be a laborious task however and frequently requires a high degree of expertise and experience. As the capabilities of CFD have increased and hence the complexity of the engineering geometries which can be computed, simpler alternatives to body fitted meshes have been sought. Two alternatives have shown promise. The first approach is to abandon the mesh altogether and to use a meshless approach, e.g. Shu et al. [1,2]. These methods use a finite difference rather than a finite volume formulation however, which means they have the disadvantages of being non-conservative. These methods have potential, but a full discussion is beyond the scope of the present paper. The second approach is the Cartesian grid approach where the flow domain is divided into simple squares in 2-d or cubes in 3-d.

Cartesian CFD methods have several advantages over methods that use a body fitted mesh. Firstly, the meshing process is considerably more straightforward and easily automated and hence the time taken to mesh a complex engineering geometry is significantly reduced. Secondly, the finite volume flow equations are simpler as the mass flux through each face depends on only one velocity component rather than all components and finally, the additional computational effort involved in problems with moving boundaries is greatly reduced because, after each time step, only the cells adjacent to the moving boundary need to be adapted rather than the whole mesh. Cartesian methods do however have one major drawback, which is that when conventional finite volume procedures are used, accuracy is compromised in the cut cells on the boundaries particularly for Navier Stokes solutions. This is often of critical importance in CFD because the boundary layers are responsible for the aerodynamic losses and hence any error in these regions leads to inaccurate prediction of performance. This problem is not restricted to Cartesian grids and is also apparent in body fitted schemes when cells, other than body fitted quadrilaterals (e.g. triangles), are used adjacent to wall boundaries.

A variety of approaches have been adopted to improve the resolution of boundary layers in cut cells which can be divided into two categories [3]. The first category of methods, which was instigated by Peskin [4], retains full cells straddling the boundaries and introduces a forcing function which modifies the flow adjacent to the boundary in order to satisfy the required boundary condition. These methods have been developed by Peskin and other researchers [5–14] but as the technique is not conservative the disadvantages associated with non-conservative schemes are inherited. The second category of methods uses the unmodified flow equations and so the solutions are conservative, but the fluxes through the cut faces need to be determined. These Cartesian methods have had considerable success for Euler flow solutions, e.g. Nemec et al. [15]. In Euler flows the necessity to resolve the high velocity shear rate, which results within the boundary layers, is avoided and
hence the modest gradients in flow properties across the cut cells adjacent to the boundary lead to acceptably small truncation errors. Boundary layer effects have been added to these calculations, e.g. Aftosmis et al. [16] by undertaking boundary layer calculations on 2-d planes aligned with the flow. This approach is applicable to computation of flow around streamline bodies providing the boundary layers remain attached or only mildly separated and the boundary layer flow is not significantly skewed by 3-d effects. The method is however not more generally applicable to flows where a full Navier Stokes solution is needed. Full Navier Stokes solutions have been obtained using embedded body fitted grids within the boundary layers, e.g. Cai et al. [17]. This displaces the cut cell resolution problem beyond the edge of the boundary layer where the flow variable gradients are once again weaker and hence any inaccuracies where the Cartesian and embedded meshes overlap are greatly reduced. This approach is however difficult to apply to general flow geometries where no a priori knowledge of the extent of the boundary layer regions exists, particularly where boundary layer separation is present. The implementation of the embedded grid is also complex in comparison with a pure Cartesian scheme and so much of the simplicity of the pure Cartesian meshing is lost. Hartmann et al. [18] use a pure Cartesian mesh and developed a conservative method for cut cells which they have applied successfully to 3-d compressible viscous flow. A least squares approach, similar to that used in meshless methods, is adopted for evaluation of the fluxes through the cut faces.

There are strong motives to develop improved finite volume schemes for Cartesian grids which retain high accuracy within the cut cells on the boundaries. The objective of the current work was to develop a second order accurate accurate procedure for determining the fluxes through the cut faces which could be implemented easily and systematically in both two and three dimensions.

1.1. Solution accuracy in cut cells

In developing new methods for cut cells, it is useful to consider why many current approaches lead to poor accuracy. A sloping boundary and the resulting cut cells for a Cartesian grid are shown in Fig. 1. For a simple second order accurate finite volume scheme, the values of velocity at the points w, e and n on the faces are calculated as the average of the values in the two adjacent cells C and W, C and E and C and N. This will only give the correct flux through the face if the velocity varies linearly along the face length and the points w, e and n lie in the centres of their respective faces. When the boundary is parallel with the gridlines, i.e. \( \alpha = 0 \), any error associated with the flux will be approximately the same on the w and e faces and so this will cancel out when the continuity and momentum balances are considered. However, when \( \alpha = 0 \) this is not the case as the deviation of the w and e points from the midpoint will differ. It should be noted that this is not a source of error for the n face in the current example as this face is uncut. To determine the magnitude of the error and its dependence on the cut angle \( \alpha \), consider the algebraic solution to the continuity and momentum equations at infinite Reynolds number

\[
U = 2X^{-1}Y \quad V = X^{-1}Y^2 \quad p = 0
\]

where U and V are the wall parallel and normal velocities in the X and Y direction as indicated in Fig. 1. Close to the wall this solution has the characteristics of a boundary layer. The velocity components at the cell centres can then be determined as

\[
u(i,j) = (\Delta x)^2(i \cos x + j \sin x)^2(-i \sin x + j \cos x)[i(1 + \cos^2 x) + j \sin x \cos x]
\]

\[
u(i,j) = (\Delta x)^2(i \cos x + j \sin x)^2(-i \sin x + j \cos x)[i \sin x \cos x + j(1 + \sin^2 x)]
\]

where i and j are the cell indices. These values can then be used to determine a resultant error in the continuity equation using the 2nd order approximation to the face values (i.e. the average of adjacent cell values). This error is assumed to be similar in magnitude to the error that would result in satisfying the continuity equation if the finite volume equations were solved. The results are shown in Fig. 2 for X = 1000 \( \Delta x \) and 10\(^{\circ} \) increments of the cut face angle \( \alpha \) with different heights of cut through the cell (cell volume). The results clearly show that the error for any non zero angle \( \alpha \) is about two orders of magnitude higher than the \( \alpha = 0 \) case. It is also apparent that to overcome this problem the strong variation in the flow quantities along the cell faces must be taken into account.

2. The new method

2.1. Cartesian mesh generation

The Cartesian meshes for the current study were generated using established procedures. A structured Cartesian mesh in two dimensions can be produced simply by dividing the whole flow domain into a number of identical square cells. Boundary nodes are then defined at each point where a mesh line crosses the flow domain and for simplicity these nodes are connected with straight edges. This leads to rounding of any sharp corners in the domain boundary, but for realistic grids this will not compromise the
definition of the domain boundary significantly. Cells that intersect with the flow domain boundaries (the cut cells) will have a smaller volume than the uncut cells (full cells) which cover the majority of the domain. Cut cells which have much smaller volumes are an undesirable feature as the CFL condition would require a significantly smaller time step during the CFD solution. For this reason, in the current method, cut cells whose cell centre lie outside the flow domain, i.e. which have a volume less than half the full cell volume, are merged with an adjacent cell which shares the longest boundary with the cut cell. When these cells share a boundary node this is deleted, which leads to further rounding of sharp corners in the flow domain boundary and may also lead to a cell centre for the merged cell outside the flow domain. If this is the case the cell undergoes a further merging along its longest shared boundary. Every merged cell therefore ultimately consists of a parent whose cell centre lies within the flow domain and one or two children whose cell centres lie outside the flow domain. An example of this process is shown in Fig. 3. Four cells C, N, NE and E are shown where the boundary defined by the original geometry cuts the N, NE and E cells such that each of their cell centres lies outside the flow domain. The point defining the corner in the original geometry within the NE cell is not retained which, prior to cell merging, leads to the initial boundary shown in the figure. The first pass of the merging algorithm will lead to the combination of cells C and N and cells E and NE. The cell centres of both E and NE are outside the flow domain and hence the combined E/NE cell undergoes a further merging with C/N during a second pass of the algorithm. The combined merging will also lead to the deletion of the two node points between NE and its neighbours resulting in the final boundary in Fig. 3. The four original cells have thus been reduced to a single cell where C is the parent and the remnants of N and E are the children.

It should be noted that, although not implemented here, any structured Cartesian grid can be refined to produce an unstructured grid using a quadtree structure whereby cells are successively subdivided into quarters. Both the merging and subdivision methodologies are easily extended to three dimensions.

2.2. Finite volume equations

In the current work the 2-d unsteady incompressible RANS equations are solved using a conservative finite volume formulation. The presentation of the methodology is therefore limited to 2-d incompressible flow, however extension to 3-d is discussed under future work and the methodology can also be used for compressible flow. The majority of existing approaches to forming finite volume equations consider the value of each flow variable \( q \) at discrete points in space. \( q \) is defined at the storage locations (typically the cell centres or corners) and values required for the fluxes are then determined on the cell faces using Taylor series approximations. The value of \( q \) elsewhere in the cell and in particular the variation in \( q \) over the cell face is not generally considered and so \( q \) is often discontinuous in value and almost always discontinuous in its first derivatives between adjoining cell faces. When the cells are approximately rectangular in shape, the cell face value is effectively evaluated at a point close to the centre of the cell face and is therefore a good approximation (usually second order) to the cell face average value and therefore the cell fluxes are accurately determined. However, as demonstrated above, if the cells deviate significantly from a rectangular shape, there is a consequential degradation in accuracy. This problem can be addressed by using a scheme whereby the flow variables are defined at all points within the flow domain and (at a minimum) their first derivatives are also continuous within the flow domain. This is achieved in the current work by defining the variation of each flow variable \( q \) throughout each cell using spline functions. As the value of \( q \) is also defined in this way within the cut cells then the variation in \( q \) over the cut faces required to determine the flux is known.

2.2.1. Full cells

To begin with consider a full cell which is surrounded by a complete set of eight full cells as shown in Fig. 4. The value of the variable \( q \) is stored in each cell, but it should be noted that this is not the value of \( q \) given by the spline function at the cell centre. This value will be determined later. The value and first derivatives can now be determined to second order at each mesh point in terms of the values stored in the four surrounding cells using Taylor series. For example for mesh point \( nw \) in Fig. 4.

\[
q_{nw} = \frac{(q_w + q_N + q_{NW} + q_N)}{4}
\]

(4)

\[
\left( \frac{\partial q}{\partial x} \right)_{nw} = \frac{(q_N + q_c - q_{NW} - q_w)}{2\Delta x}
\]

(5)

\[
\left( \frac{\partial q}{\partial y} \right)_{nw} = \frac{(q_N + q_{NW} - q_c - q_w)}{2\Delta y}
\]

(6)

and also

\[
\left( \frac{\partial^2 q}{\partial x \partial y} \right)_{nw} = \frac{(q_N + q_w - q_{NW} - q_c)}{\Delta x \Delta y}
\]

(7)

The value for each variable \( q \) at any point in the cell can now be defined using the cubic spline equation

![Fig. 4. Full cell surrounded by eight cell values.](Image)
\[ q(x, y) = f_0(x)f_1(y)q_{sw} + f_1(x)f_0(y)q_{se} + f_0(x)f_1(y)q_{nw} + g_1(x)f_0(y)q_{ne} + g_0(x)f_1(y)q_{nw} + f_1(x)g_0(y)q_{sw} + f_0(x)g_0(y)q_{ne} + f_1(x)g_1(y)q_{se} + g_0(x)g_1(y)q_{nw} + g_1(x)y\] 

where the functions \( f_0, f_1, g_0 \) and \( g_1 \) are given by 

\[ f_0(x) = \frac{-(x-1)^2(2x+1)}{(\Delta x)^2}, \quad f_1(x) = x^2(3-2x), \quad g_0(x) = \frac{x(x-1)^2}{(\Delta x)^2} \quad \text{and} \quad g_1(x) = \frac{x^2(x-1)}{(\Delta x)^2}. \]

and the origin for \( x \) and \( y \) is the sw mesh point.

The value of \( q \) determined from the spline functions at the cell centre is therefore

\[
\frac{1}{4}(q_{sw} + q_{se} + q_{nw} + q_{ne}) + \frac{\Delta x}{16} \frac{\partial q}{\partial x}_{sw} + \Delta y \frac{\partial q}{\partial y}_{se} + \frac{\Delta y}{15} \left( \frac{\partial^2 q}{\partial y^2}_{se} - \frac{\partial^2 q}{\partial y^2}_{nw} - \frac{\partial^2 q}{\partial y^2}_{ne} \right) + \frac{\Delta x \Delta y}{64} \left( \frac{\partial^2 q}{\partial x \partial y}_{se} - \frac{\partial^2 q}{\partial x \partial y}_{nw} - \frac{\partial^2 q}{\partial x \partial y}_{ne} \right) = \frac{9}{16} (q_{sw} + q_{se} + q_{nw} + q_{ne}) + \frac{3}{32} (q_{sw} + q_{se} + q_{nw} + q_{ne})
\]

The cubic spline formulation ensures that the value and first derivatives are continuous across cell boundaries. The volume flux through the vertical face between the mesh points \( sw \) and \( nw \) is therefore given by

\[
\int_{y=0}^{y=\Delta y} u dy = \int_{y=0}^{y=\Delta y} \left( f_0(y)u_{sw} + f_1(y)u_{nw} + g_0(y)\frac{\partial u}{\partial y}_{sw} + g_1(y)\frac{\partial u}{\partial y}_{nw} \right) dy
\]

\[
= \frac{1}{2} \left( u_{sw} + u_{ne} \right) + \frac{\Delta y}{12} \left( \frac{\partial u}{\partial y}_{sw} - \frac{\partial u}{\partial y}_{nw} \right) \Delta y
\]

\[
= \frac{1}{3} \left( u_{sw} + u_{se} - \frac{1}{2} \left( u_{sw} + u_{s} + u_{nw} + u_{ne} \right) \right) \Delta y
\]

and so is dependent on the six cell values nearest to that face.

This formulation therefore differs from a conventional second order finite volume approximation as the variation of \( u \) in both the \( x \) and \( y \) direction is considered. It seems entirely appropriate however to improve accuracy by including additional cell values in their order of proximity to the face being considered. The full finite volume equations for mass and momentum will therefore use a nine point template consisting of the current cell value and the values from the surrounding eight cells. The template involves more points than the seven points required for a conventional second order upwind scheme however the coefficients are simpler to evaluate than for a body fitted grid and for full cells in 2-d there are only three coefficient values, for the centre point \( C \), offset points \( W, E, N \) and \( S \) and diagonal points \( NW, NE, SW \) and \( SE \). This increases to five coefficient values in 3-d. There is therefore an increased computational overhead when compared with a conventional Cartesian grid formulation, however the computational cost is still considerably less than for a body fitted grid system. For consistency and simplicity in coding, this nine point template is maintained for cells close to the boundaries where some of the surrounding eight cell values are no longer available.

2.2. Boundaries

For mesh points close to the boundaries, Eqs. (4)–(7) cannot be used because either one or two of the surrounding cells are cut such that they no longer store a value. These situations are shown in Fig. 5 where for the mesh point \( nw \) the value in the cell \( NW \) (Fig. 5a) or the values in the cells \( NW \) and \( N \) (Fig. 5b) are not stored. If a boundary condition for \( q \) is available on the cut face for all the cut cells surrounding the mesh point then these boundary condition values at the points where the diagonal intersects the boundary can be used in place of each missing cell stored value as shown in Fig. 6. The revised equations which replace Eqs. (4)–(7) can then be derived from the Taylor series to obtain

\[
q_{nw} = \frac{1}{(f_{sw} + f_{se})} \left( f_{sw}(q_{sw} + q_{se}) + f_{se}(q_{se} + q_{sw}) + f_{nw}(q_{nw} + q_{ne}) + f_{ne}(q_{ne} + q_{nw}) \right)
\]

\[
\frac{\partial q}{\partial x}_{nw} = \frac{1}{2} \frac{\Delta x}{f_{sw} + f_{se}} \left( \frac{f_{sw}}{f_{sw} + f_{se}} q_{nw} + \frac{f_{se}}{f_{sw} + f_{se}} q_{nw} + \frac{f_{nw}}{f_{sw} + f_{se}} q_{nw} + \frac{f_{ne}}{f_{sw} + f_{se}} q_{nw} \right)
\]

\[
\frac{\partial q}{\partial y}_{nw} = \frac{1}{2} \frac{\Delta y}{f_{sw} + f_{se}} \left( \frac{f_{sw}}{f_{sw} + f_{se}} q_{nw} + \frac{f_{se}}{f_{sw} + f_{se}} q_{nw} + \frac{f_{nw}}{f_{sw} + f_{se}} q_{nw} + \frac{f_{ne}}{f_{sw} + f_{se}} q_{nw} \right)
\]

and

**Fig. 5.** Determining values at the \( nw \) mesh point near a boundary where (a) one surrounding cell value is absent and (b) two surrounding cell values are absent.
Fig. 6. Mesh point value determination using a boundary condition value.

\[
\left( \frac{\partial^2 q}{\partial x \partial y} \right)_{nw} = \frac{f_W}{(f_N+f_W)} q_{n} - \frac{f_c}{(f_{NW}+f_c)} q_{NW} + \frac{f_N}{(f_N+f_W)} q_{W} - \frac{f_{NW}}{(f_{NW}+f_c)} q_{C}
\]

where \( f = 0.5 \) where a cell value is stored and otherwise, when a boundary condition value is used, \( f \) is the proportion of the distance along the cell diagonal where the boundary is crossed. \( f \) is therefore always less than or equal to 0.5. It should be noted that when \( f_w = f_{NW} = f_n = f_c = 0.5 \), i.e. the limit case when the boundary passes through the cell centre, these equations reduce to Eqs. (4)–(7).

If a boundary condition for \( q \) is not available on the cut face boundary then an alternative approach is needed to obtain the values at the mesh point. When three cell values are available and used in the Taylor series, the values and derivatives are given by

\[
q_{nw} = \frac{(q_n + q_W)}{2}
\]

(16)

\[
\left( \frac{\partial q}{\partial x} \right)_{nw} = \frac{(q_c - q_w)}{\Delta x}
\]

(17)

\[
\left( \frac{\partial q}{\partial y} \right)_{nw} = \frac{(q_N - q_c)}{\Delta y}
\]

(18)

with

\[
\left( \frac{\partial^2 q}{\partial x \partial y} \right)_{nw} = 0
\]

(19)

If just two storage cell values are available, two additional cell values can be introduced whilst maintaining a nine point template for each finite volume equation. For the example in Fig. 5b these are the values from the SW and S cells. The formulae are then

\[
q_{nw} = \frac{3(q_W + q_c) - (q_{SW} + q_S)}{4}
\]

(20)

\[
\left( \frac{\partial q}{\partial x} \right)_{nw} = \frac{3(q_c - q_w) - (q_S - q_{SW})}{2\Delta x}
\]

(21)

\[
\left( \frac{\partial q}{\partial y} \right)_{nw} = \frac{(q_W + q_c - q_{SW} - q_S)}{2\Delta y}
\]

(22)

and

\[
\left( \frac{\partial^2 q}{\partial x \partial y} \right)_{nw} = \frac{(q_c + q_{SW} - q_W - q_S)}{\Delta x \Delta y}
\]

(23)

The final case that needs to be considered is where two cell centre values are available (Fig. 5b) and a boundary condition is available for only one of the cut cell boundaries. The mesh point values are then derived from these three values. For the example in Fig. 5b, the value and derivatives at the \( nw \) mesh point when the NW value is missing are given by

\[
q_{nw} = \frac{f_W}{(f_N+f_W)} q_{n} + \frac{f_N}{(f_N+f_W)} q_{W}
\]

(24)

\[
\left( \frac{\partial q}{\partial x} \right)_{nw} = \frac{1}{2\Delta x(f_W+f_N)}[(f_c-f_W)q_n -(f_N+f_C)q_W+(f_N+f_c)q_C]
\]

(25)

\[
\left( \frac{\partial q}{\partial y} \right)_{nw} = \frac{1}{2\Delta y(f_W+f_N)}[(f_c+f_W)q_n+(f_n-f_C)q_W-(f_W+f_n)q_C]
\]

(26)

and

\[
\left( \frac{\partial^2 q}{\partial x \partial y} \right)_{nw} = 0
\]

(27)

These mesh point values can then be used with Eq. (8) to obtain the flux through each uncut face.

2.2.3. Cut cell faces

It should be noted that the term ‘cut cell’ is used here for any cell which has a cut face and hence may be a merged cell consisting of a parent cell and up to two child cells. For cut cells one or two mesh point corners are missing resulting in two bisected faces and a single diagonal face as shown in Fig. 7. In the case of merged cells where one or more of the faces will exceed the full cell dimension, the bisected face is considered as the portion of the face extending beyond the full face dimension. The values and derivatives of the variables are required along the length of both the bisected faces and the diagonal face. Where a boundary condition is not available, this is achieved by extrapolating values from the nearest mesh point and so the values along the bisected face are given by

\[
q = q_M + x\left( \frac{\partial q}{\partial x} \right)_{bt}
\]

(28)

\[
\frac{\partial q}{\partial x} = \frac{\partial q}{\partial x}_{M}
\]

(29)

\[
\frac{\partial q}{\partial y} = \frac{\partial q}{\partial y}_{M} + \left( \frac{\partial^2 q}{\partial x \partial y} \right)_{c} x
\]

(30)

where \( M \) refers to the mesh point and the cut point and \( x \) is the distance from the mesh point. The values at the cut point are obtained when \( x = x_c \), where the subscript \( c \) refers to the cut point. The value at a point a distance \( x \) along the diagonal face is obtained from the cubic spline

\[
q = f_0(x)q_{cb} + f_1(x)q_{c1} + g_0(x)\left( \frac{\partial q}{\partial x} \right)_{c0} + g_1(x)\left( \frac{\partial q}{\partial x} \right)_{c1}
\]

(31)

where here

\[
f_0(x) = \frac{(x-1)^2(2x+1)}{x^2}, \quad f_1(x) = \frac{x^2(3-2x)}{x^2},
\]

\[
g_0(x) = \frac{x(x-1)^2}{x^2} \quad \text{and} \quad g_1(x) = \frac{x^4(x-1)}{x^2}
\]

(32)

When a boundary condition value is available, a quadratic equation is used to give the variation of the variable along the bisected face, so
the calculation, were evaluated at the start and stored in memory terms in the equation \( s \), which remained invariant throughout. Therefore, more laborious to compute. The template \( s \) for the linear points than a conventional 2nd order method and are solved using point over-relaxation.

The resulting Poisson equation in pressure is solved, but the methodology is readily applied to any form of the continuity equation. A conventional two-step pressure correction method following Voke and Yang [19] is used to solve the equations. In the first step, the velocity is marched explicitly in time using the velocity terms in the momentum equations. The time derivative is evaluated to second order accuracy in time by utilising the momentum flux from the previous time level.

By utilising the momentum flux from the previous time level.

In the second step, the pressure is corrected such that the residual change in the velocity from the momentum equation satisfies the continuity equation. The resulting Poisson equation in pressure is solved using point over-relaxation.

The finite volume templates using the new method involve the boundary condition value is used on the diagonal face.

### 2.2.4. Cut cell errors

The error in the continuity equation for the range of cut cells, already considered using a simple 2nd order finite volume scheme, was determined using the new method. The results are also shown in Fig. 2 and indicate that, for all cut angles, the resultant error remains below that achieved at \( \alpha = 0 \) for the conventional 2nd order scheme. The objective of devising an accurate finite volume formulation for cut cells therefore appears to have been achieved. It has been assumed though that solution of the flow equations using the new method will result in errors similar to the ones calculated in Fig. 2 from an exact flow solution. Full CFD solutions are required to validate the new method.

### 2.3. Solution procedure

In the current work, the unsteady 2-d RANS equations are solved, but the methodology is readily applied to any form of conservation equations. A conventional two-step pressure correction method following Voke and Yang [19] is used to solve the equations. In the first step, the velocity is marched explicitly in time using the velocity terms in the momentum equations. The time derivative is evaluated to second order accuracy in time by utilising the momentum flux from the previous time level. Thus

\[
q(x) = \left( 1 - \left( \frac{x}{X_C} \right)^2 \right) q_M + \left( \frac{x}{X_C} \right)^2 q_C + x \left( 1 - \frac{x}{X_C} \right) \left( \frac{\partial q}{\partial x} \right)_M
\]

and the boundary condition value is used on the diagonal face.

### 3. Test cases

Current Cartesian grid formulations lead to poor resolution of the flow in the cut cells and so test cases were chosen specifically to determine whether the current formulation overcomes this deficiency.

#### 3.1. Test case 1

The first test case was chosen to test the accuracy of the current method for highly curved boundaries. The geometry is shown in Fig. 8 and consists of an annular channel of inner and outer radii, \( r_i \) and \( r_o \), respectively. Both the inner and outer boundaries are stationary walls and flow is induced by introducing a discontinuity in pressure along the right hand horizontal radius. For laminar flow this flow has the analytical solution

\[
\frac{u_0}{u} = \left[ \left( r_o^2 \ln \left( \frac{r_o}{r_i} \right) - r_i^2 \ln \left( \frac{r_i}{r_o} \right) \right) r + r_i^2 r_o^2 \ln \left( \frac{r_o}{r_i} \right) \frac{1}{r} \right] / \left[ -\frac{1}{4} (r_o^2 - r_i^2)^2 + r_i^2 r_o^2 \ln \left( \frac{r_o}{r_i} \right)^2 \right]
\]

where \( u \) is the bulk tangential velocity. The flow solution is independent of the Reynolds number. Calculations were performed for outer to inner radius ratios of 3 and 6 using the coarse and fine grids shown in Fig. 8, which have cell sizes of 0.1 \( r_o \) and 0.05 \( r_o \), respectively. Velocity vector plots of the results are shown in Fig. 9 and the tangential velocities determined for all mesh points are plotted in Fig. 10. The latter figure shows a slight dependence on angular position just for the coarse grid and radius ratio of 6 close to the inner radius. This is due to the poor grid definition of the inner radius for this case which is apparent in Fig. 8. There is also a small difference between the calculated values and the analytical solution for the coarse grid and the radius ratio of 3, but there is no indication of any dependence on angular position. The results on the fine grid for both radius ratios are within 1% of the analytical solution.

In order to check that the calculations were of second order accuracy, a mesh convergence test was performed for the radius ratio of 6 for mesh sizes from 0.01 \( r_o \) to 0.1 \( r_o \). The error, computed as the average absolute value of the difference between the analytical and numerical solutions, for both the tangential and radial velocity components is shown in Fig. 11. The error values correctly follow the trend line for a second order scheme. A further test was performed using the wall shear stress in order to confirm that second order accuracy is achieved in the cut cells alone. The average absolute error in the wall shear stress is plotted in Fig. 11. The wall
Fig. 8. Coarse and fine grids for outer to inner radius ratios of 3 and 6.

Fig. 9. Velocity vectors on coarse and fine grids for radius ratios of 3 and 6.
shear stress depends on the velocity gradient and hence for second order accuracy in velocity, the wall shear stress error should follow the first order trend line. This is the case for all but the coarsest grid, where the poor definition of the inner wall shown in Fig. 8 leads to an error on the inner wall which is more than four times greater than that on the outer wall. In comparison, for the finer meshes the error on the inner wall is only about 50% greater. These results therefore demonstrate that the cut cell methodology presented in this paper achieves second order solution accuracy for highly curved boundaries.

3.2. Test case 2

The flow around an isolated cylinder of diameter $D$ is chosen as the second test case as there is a large quantity of experimental and numerical data available for validation. The uniform computational grid shown in Fig. 12 uses a mesh size of $0.1D$ and extends to $10D$ upstream, above and below the cylinder and to $20D$ downstream. A uniform flow boundary condition was imposed upstream with constant pressure boundary conditions on the other three outer boundaries. Calculations were performed at Reynolds numbers of 20, 40, 50, 80 and 300. Velocity vector plots for the region close to the cylinder and for the near wake are shown in Fig. 13. Experimental observations Wieselsberger [20] and
Williamson [21] have shown that for $Re < 45$, a steady wake develops, but at higher $Re$ alternate vortices are shed from the cylinder. In the current results for $Re = 20$ and 40 the wake flow remains steady, but becomes unsteady for $Re = 50$, 80 and 300. The time history of the lift and drag coefficients at $Re = 80$ is shown in Fig. 14 and is very similar to that obtained by Ye et al. [22]. Ye used an non-uniform Cartesian grid with 16 cells across the diameter of his cylinder compared with just 10 cells for the current work.

Williamson [21] have shown that for $Re < 45$, a steady wake develops, but at higher $Re$ alternate vortices are shed from the cylinder. In the current results for $Re = 20$ and 40 the wake flow remains steady, but becomes unsteady for $Re = 50$, 80 and 300. The time history of the lift and drag coefficients at $Re = 80$ is shown in Fig. 14 and is very similar to that obtained by Ye et al. [22]. Ye used an non-uniform Cartesian grid with 16 cells across the diameter of his cylinder compared with just 10 cells for the current work.

**Table 1**

<table>
<thead>
<tr>
<th>Reynolds number</th>
<th>20</th>
<th>40</th>
<th>80</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$C_D$</td>
<td>$L/d$</td>
<td>$C_D$</td>
<td>$L/d$</td>
</tr>
<tr>
<td>Wieselsberger</td>
<td>2.05</td>
<td>1.70</td>
<td>1.45</td>
<td>1.22</td>
</tr>
<tr>
<td>[20]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Williamson [21]</td>
<td>2.03</td>
<td>0.92</td>
<td>1.52</td>
<td>2.27</td>
</tr>
<tr>
<td>Ye et al. [22]</td>
<td>2.17</td>
<td>0.93</td>
<td>1.65</td>
<td>2.08</td>
</tr>
<tr>
<td>Current</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Fig. 13.** Wake downstream of isolated cylinder for $Re = 20$, 40, 80 and 300.

**Fig. 14.** Lift and drag coefficients for isolated cylinder flow at $Re = 80$.

**Fig. 15.** Plate geometry. Plate overall length 1268 mm.
The unsteadiness is much weaker at $Re = 50$ with peak to peak variations in $CL$ of $10^{-4}$ at a Strouhal number of 0.083. Table 1 shows the wake extent $L$, measured to the detached stagnation point, for the steady wakes and the Strouhal numbers $St$ for higher $Re$ values for the current work, the experimental results Wieselsberger [20] and Williamson [21] and the numerical work of Ye et al. [22]. The current values are very similar to these published results hence confirming that the numerical approach provides accurate results for this test case.

3.3. Test case 3

The third test case is the development of a laminar boundary layer on the flat plate shown in Fig. 15. The plate is 1268 mm in length with a thickness of 6 mm and has a semi elliptic 6:1 leading edge. Five grids were used each consisting of approximately 140,000 $1 \text{mm} \times 1 \text{mm}$ cells where the gridlines were at angles of 0°, 10°, 20°, 30° or 40° to the plate surface. These five angles cover the full range of possible grids formed when the gridlines are rotated in increments of 10° through a full 360°. Detail of the leading edge region for each of the grids is shown in Fig. 16. A wide range of cut cell and parent/child merged cells are present. A uniform flow inlet condition was applied on the boundary upstream of the plate with constant pressure outlet conditions applied on the boundaries above, below and downstream of the plate. The flow geometry and boundary conditions were identical for the five grids such that the effect on the flow solution of rotating the gridlines could be assessed. The flow was computed on each of the five grids for plate Reynolds numbers of 25,000, 62,500 and 125,000 (three different viscosities). Boundary layer profiles were extracted at ten streamwise stations along the plate for each flow. For each profile the velocity values were determined using the spline functions at points where the measurement plane intersected each gridline. The boundary layer integral parameters were then obtained through numerical integration of the profiles. The integral parameters are highly sensitive to the value of freestream velocity used in their calculation and so an accurate algorithm is required. The velocity was averaged starting from the outermost freestream...
velocity and this average value was compared with the local value and the local velocity gradient once the maximum velocity in the profile had been passed as shown in Fig. 17. The edge of the boundary layer was defined as the point where the local velocity was less than the integrated average velocity and the velocity gradient was greater than 5% of the gradient obtained by dividing the local velocity by the distance from the wall. The integrated average velocity at this point is then used as the freestream velocity in evaluating the boundary layer integral parameters. This definition ensures only the flow inside the boundary layer contributes to the displacement thickness.

The predicted development of the boundary layer along the plate for all 15 cases is shown in Fig. 18 together with the theoretical Blasius prediction. It can be seen that there is virtually no
difference between the results for the different grids at the lowest Reynolds number, but as the Reynolds number is increased some grid dependence is apparent. The discrepancy appears to be largely due to incorrect prediction of the boundary layer development close to the leading edge as if the points on each grid were plotted using a different origin for \( Re \), then they would lie much closer to the theoretical Blasius curve for the higher plate Reynolds numbers. It should be noted though that the largest errors are not associated with the larger grid angles and it is actually the 0° results which lie furthest from the theoretical curve. The boundary layer shape factors presented in Fig. 19 show a similar trend with increasing grid dependence as the plate Reynolds number is increased particularly close to the leading edge. The full boundary layer profiles in Fig. 20 show these discrepancies more clearly. The differences in the u velocity profiles for the profiles close to the leading edge can be seen, but larger errors occur in the \( v \) velocity profile which, when plotted in non-dimensional form, is subject to errors in the determination of the momentum thickness. One possible reason for these discrepancies is the small changes in the leading edge geometry which are apparent in the grids shown.

Fig. 20. Boundary layer profiles at a plate Reynolds number of 125,000 for the 0°, 10°, 20°, 30° and 40° grids. Black symbols – upper and grey symbols – lower boundary layer.
in Fig. 16. However, this does not have a significant effect at lower Reynolds numbers. The velocity vector diagrams for $Re_{plate} = 125,000$ in Fig. 21 indicate that the boundary layer is only a few cells thick in the leading edge region and is only about six cells thick at $Re_x = 10,000$ where the first profile is taken. The resolution of the boundary layer is therefore significantly less than the $Re_{plate} = 25,000$ case where the boundary layer extends to about 14 cells at this location. The calculations were therefore repeated for $Re_{plate} = 125,000$ using a grid with 0.5 mm × 0.5 mm cells. The boundary layer results are presented in Fig. 22 and velocity vector plots of the leading edge flow in Fig. 23. The results in Fig. 22 show that the improved resolution of the leading edge reduces the variation with grid angle substantially and the increased number of cells within the boundary layer also leads to results closer to
the theoretical Blasius values. Some of the shape factors close to
the leading edge region are still more than 10% greater than the
Blasius value, but over the majority of the plate are within 3%.
There is a streamwise favourable pressure gradient from the stag-
nation point over the leading edge followed by a weak adverse
pressure gradient as the leading edge curvature relaxes which will
result in a slightly raised shape factor not accounted for in the Bla-
sius theory. Velocity results for the complete flow domain are
shown in Fig. 24 for $\alpha = 30^\circ$. This figure shows how the vortex
shedding from the blunt trailing edge of the plate is captured by
the present unsteady calculation.

The most important observation though is the lack of any differ-
ence between the $\alpha = 0$ case and the other cases where the grid-
lines are at an angle to the plate confirming that the accuracy
achieved in rectangular cells ($\alpha = 0$) has also been achieved in a full
range of skewed cells ($\alpha \neq 0$).

4. Conclusions

(1) A new method for representation of flow variables through-
out a Cartesian cell using spline functions has been de-
veloped. This method provides a convenient way to evaluate
the fluxes in mass and momentum through the faces of cut
cells on the flow domain boundaries. Results from a mesh
convergence test confirm that the numerical scheme
achieves 2nd order accuracy for both the cut and uncut cells.
(2) The method is tested for three flow cases to produce accu-
rate results for cells which are cut at any angle even for
boundaries with high curvature where this angle changes
rapidly from cell to cell. The accuracy of the current method
is shown to be at least as good as other published Cartesian
methods.

5. Further work

Although not reported here, the method described in this paper
has been used successfully to calculate 2-d flow for several other
geometries including separated flow from an aerofoil and flow over a backward facing step. The accuracy of the results is similar to that for calculations on a body fitted grid of similar density but with a reduction in both memory and cpu time requirements.

There are no restrictions to extending the current method to 3-d flows. The algorithms presented in this paper, although more complex to implement in 3-d, remain valid. However there are only three types of 2-d cut cell (2 × 4 node and 1 × 5 node) whereas there are five types (1 × 8 node and 4 × 10 node) in 3-d and this together with the increased number of nodes per cell and the use of splines in three directions increases complexity. This complexity will not add significantly to the cpu overhead as the template coefficients do not need to be evaluated at each time step.

Another necessary avenue for development of the current method is the adoption of quadtree (octree in 3-d) methods in order to provide high cell densities only in areas of high shear rate. This will lead to a significant reduction in the total number of computation cells required for any particular calculation. The cell faces between cells of differing size (quadtree/octree level) will need to provide high cell densities only in areas of high shear rate. There are no restrictions to extending the current method to 3-d geometries including separated flow from an aerofoil and flow over a backward facing step. The accuracy of the results is similar to that for calculations on a body fitted grid of similar density but with a reduction in both memory and cpu time requirements.

There are no restrictions to extending the current method to 3-d flows. The algorithms presented in this paper, although more complex to implement in 3-d, remain valid. However there are only three types of 2-d cut cell (2 × 4 node and 1 × 5 node) whereas there are five types (1 × 8 node and 4 × 10 node) in 3-d and this together with the increased number of nodes per cell and the use of splines in three directions increases complexity. This complexity will not add significantly to the cpu overhead as the template coefficients do not need to be evaluated at each time step.

Another necessary avenue for development of the current method is the adoption of quadtree (octree in 3-d) methods in order to provide high cell densities only in areas of high shear rate. This will lead to a significant reduction in the total number of computation cells required for any particular calculation. The cell faces between cells of differing size (quadtree/octree level) will need special treatment, but this is obtainable without compromising the continuity in the flow quantity first derivatives achieved in the present paper.

References