Chapter 5

Finite Volume Methods for Non-Orthogonal Meshes

For most fluid mechanics problems of interest to Engineers the geometry of the problem can not be represented by a Cartesian mesh. Instead it is common for the boundaries to be curved in space. The Cartesian finite volume methods described in Chapters 2 and 4 must be extended to a non-orthogonal mesh to allow non-rectangular geometries to be accurately modelled.

A structured mesh is topologically rectangular, but may be deformed in space such that it is no longer Cartesian. Such a mesh may still be orthogonal, with the intersections of the mesh lines being at 90°, but such a mesh can be difficult to generate, and so a completely general mesh will be non-orthogonal with no restriction on the angles of the intersections of the mesh lines.

In this chapter the finite volume discretisation of the transport equation on a non-orthogonal mesh is discussed in five parts. First there is a description of the geometry of the mesh, which is followed by the discretisation of the Diffusion terms of the transport equation. The advective terms will then be discretised and the boundary conditions described. Finally the solution of the Navier–Stokes equations on a non-orthogonal mesh will be described, using the SIMPLE scheme discussed in Chapter 4. The method used to discretise the equations is similar to that described by Jones[72] and Perić[129, 43].

5.1 The Geometry of a Non-Orthogonal Mesh

The calculation of the geometric properties of a non-orthogonal mesh is non-trivial, and so it deserves discussion before discretising a PDE upon the mesh. Both two and three dimensional meshes are discussed, the simplifications arising from the use of a two-dimensional mesh warranting their separate discussion.

A non-orthogonal regular two-dimensional mesh is shown in Figure 5.1, with the mesh being shown in both computational and physical space. In computational space the coordinate space is represented by the $\xi, \eta, \zeta$ axes, whilst in physical space the $x, y, z$ axis system is used, with there being a one to one mapping between the two coordinate systems.

For a finite volume discretisation the mesh is defined by the cell vertex points. From these points we wish to find the cell centre, the centre of the cell faces, the area vectors of the cell faces, and the cell volume.

For the code used in this study the cell centre and the centre of the cell faces were approximated by
the average of the cell vertex values. Whilst this doesn’t give the geometric centroid of the cell it was considered a reasonable approximation.

For two dimensional cells the cell face area vectors can be calculated as the cross product of the cell face and the vector normal in the direction of a third (fictitious) axis. Thus for the east face for the cell in Figure 5.2 the cell face vector is

$$\Delta x_e = x_{ne} - x_{se},$$

and so the face area is

$$A_e = \Delta x_e \times k.$$  (5.1)

In component form this becomes

$$A_{ex} = \Delta y_e,$$
$$A_{ey} = -\Delta x_e.$$  (5.2)

The volume of the cell, $\Omega$, can be easily calculated as half the magnitude of the cross product of the cell diagonals, the volume of the cell in Figure 5.2 being

$$\Omega = \frac{1}{2} |(x_{ne} - x_{nw}) \times (x_{nw} - x_{se})|.$$  (5.3)

It is prudent to use the magnitude of the volume to prevent the inadvertent calculation of negative volumes.

For a three dimensional mesh the cell face areas are found as half the cross product of the face diagonals, so for the cell face in Figure 5.3 the face area is given by

$$A = \frac{1}{2} \left( a \times b \right).$$  (5.4)
5.2 The Discretisation of the Diffusion Terms on a Non-Orthogonal Mesh

To discretise the diffusion equation on a non-orthogonal mesh we need to find an approximation for the diffusive flux across a face of a mesh volume in a manner similar to that used in Section 2.2. For our example we will take the eastern face of a cell, but the process can be applied to any face of the volume. The geometry of the face in physical space is shown in Figure 5.4 along with the geometry in computational space.

From Fick’s law the diffusive flux is given by

\[ \Gamma \frac{\partial \phi}{\partial n}, \]  

where \( n \) is the direction normal to the surface. If \( A_n \) is the area vector of the face, and \( A_n = |A_n| \) is
the magnitude of the face area, then the total flux across the face is approximated by

\[ A_n \Gamma \frac{\partial \phi}{\partial n}. \tag{5.8} \]

The area \( A_n \) is easily calculated from the mesh geometry as was described in the previous section, and \( \Gamma \) is a given property, so all that remains is to approximate the derivative \( \frac{\partial \phi}{\partial n} \). We can readily approximate the derivatives in the computational axes \( \xi, \eta \) or \( \zeta \) using centred difference approximations. Unfortunately for a non-orthogonal mesh the face normal is not necessarily in the direction of any of the mesh axes \( \xi, \eta \) or \( \zeta \), but it can be approximated using the derivatives in the mesh axes by use of the chain rule

\[ \frac{\partial \phi}{\partial n} = \frac{\partial \phi}{\partial \xi} \frac{\partial \xi}{\partial n} + \frac{\partial \phi}{\partial \eta} \frac{\partial \eta}{\partial n} + \frac{\partial \phi}{\partial \zeta} \frac{\partial \zeta}{\partial n}. \tag{5.9} \]

As was mentioned, the derivatives of \( \phi \) in computational space can be approximated by centred difference approximations, which for the east face of the cell may be written as

\[
\begin{align*}
\frac{\partial \phi}{\partial \xi} &\approx \phi_E - \phi_P, \\
\frac{\partial \phi}{\partial \eta} &\approx \phi_{ne} - \phi_{se} \approx \frac{1}{4} (\phi_N + \phi_{NE} - \phi_S - \phi_{SE}), \\
\frac{\partial \phi}{\partial \zeta} &\approx \phi_{te} - \phi_{be} \approx \frac{1}{4} (\phi_T + \phi_{TE} - \phi_B - \phi_{BE}),
\end{align*}
\tag{5.10}
\]

noting that for a unit cell in computational space, \( \Delta \xi = \Delta \eta = \Delta \zeta = 1 \).

The mesh transformation terms can be calculated from a dot product of the cell face unit normal \( \mathbf{n} \) and the contravariant basis vectors

\[
\begin{align*}
e_1 &= \frac{\partial \xi}{\partial x}, \\
e_2 &= \frac{\partial \eta}{\partial y}, \\
e_3 &= \frac{\partial \zeta}{\partial z}.
\end{align*}
\tag{5.11}
\]
CHAPTER 5. NON-ORTHOGONAL MESHES

with

\[
\frac{\partial \xi}{\partial n} = \frac{\partial \xi}{\partial x}, \\
\frac{\partial \eta}{\partial n} = \frac{\partial \eta}{\partial x}, \\
\frac{\partial \zeta}{\partial n} = \frac{\partial \zeta}{\partial x},
\]

and with the unit normal being given by

\[
n = \frac{A_n}{|A_n|}.
\]

Figure 5.5: The covariant \( e_1, e_2 \) and contravariant \( e^1, e^2 \) basis vectors for the eastern face of a cell in a two-dimensional mesh.

Unfortunately the contravariant basis vectors \( e^1, e^2, e^3 \) (shown in Figure 5.5) which are normal to the \( \xi, \eta, \zeta \) mesh surfaces are unknown. However, an approximation to the covariant basis vectors, which are tangential to the \( \xi, \eta, \zeta \) surfaces

\[
e_1 = \frac{\partial x}{\partial \xi}, \\
e_2 = \frac{\partial x}{\partial \eta}, \\
e_3 = \frac{\partial x}{\partial \zeta},
\]

can be readily calculated from the mesh geometry. Thus we wish to estimate the contravariant basis vectors from their covariant counterparts.

By considering a parallelepiped formed at the east face of a cell by the covariant vectors (Figure 5.6), we can approximate the contravariant vectors from the ratio of the face areas to the volume of the parallelepiped. Thus

\[
e^1 = \frac{\partial \xi}{\partial x} \approx \frac{A^1}{\Omega}, \\
e^2 = \frac{\partial \eta}{\partial x} \approx \frac{A^2}{\Omega}, \\
e^3 = \frac{\partial \zeta}{\partial x} \approx \frac{A^3}{\Omega}.
\]

The face areas are easily calculated by taking the cross products of the covariant basis vectors, whilst the volume is calculated from the dot product of the normal area and \( e_1 \),

\[
A^1 = e_2 \times e_3, \\
A^2 = e_3 \times e_1, \\
A^3 = e_1 \times e_2, \\
\Omega = A^1.e_1.
\]
\( \mathbf{A}^{1} \) is equivalent to the cell face normal previously denoted as \( \mathbf{A}_{n} \).

Figure 5.6: The parallelepiped formed at the east face of a cell by the covariant vectors.

Substituting the approximations in Equation (5.15) into (5.12) gives

\[
\begin{align*}
\frac{\partial \xi}{\partial n} &= \frac{\mathbf{A}^{1} \cdot \mathbf{A}^{1}}{\Omega}, \\
\frac{\partial \eta}{\partial n} &= \frac{\mathbf{A}^{2} \cdot \mathbf{A}^{1}}{\Omega}, \\
\frac{\partial \zeta}{\partial n} &= \frac{\mathbf{A}^{3} \cdot \mathbf{A}^{1}}{\Omega},
\end{align*}
\]

(5.17)

where \( \mathbf{A}^{1} \) has been substituted for \( \mathbf{A}_{n} \). One final simplification is available from Equation (5.8) where it is seen that the derivatives multiplied by the face area \( |\mathbf{A}^{1}| \) are required. Substituting into Equation (5.17) gives the expressions for the geometric diffusion coefficients \( G^{1} \) for the east face (or face 1),

\[
\begin{align*}
G^{1} &= \frac{\mathbf{A}^{1} \cdot \mathbf{A}^{1}}{\Omega}, \\
G^{2} &= \frac{\mathbf{A}^{1} \cdot \mathbf{A}^{2}}{\Omega}, \\
G^{3} &= \frac{\mathbf{A}^{1} \cdot \mathbf{A}^{3}}{\Omega},
\end{align*}
\]

(5.18)

where

\[
|\mathbf{A}_{n}| \frac{\partial \phi}{\partial n} = G^{1} \frac{\partial \phi}{\partial \xi} + G^{2} \frac{\partial \phi}{\partial \eta} + G^{3} \frac{\partial \phi}{\partial \zeta}.
\]

(5.19)

Similar expressions can be derived for the north and top faces (faces 2 and 3) of the cell, and the derivatives normal to these faces can be approximated as

\[
\begin{align*}
|\mathbf{A}^{11}| \frac{\partial \phi}{\partial n^{1}} &= G^{11} \frac{\partial \phi}{\partial \xi} + G^{12} \frac{\partial \phi}{\partial \eta} + G^{13} \frac{\partial \phi}{\partial \zeta}, \\
|\mathbf{A}^{22}| \frac{\partial \phi}{\partial n^{2}} &= G^{21} \frac{\partial \phi}{\partial \xi} + G^{22} \frac{\partial \phi}{\partial \eta} + G^{23} \frac{\partial \phi}{\partial \zeta}, \\
|\mathbf{A}^{33}| \frac{\partial \phi}{\partial n^{3}} &= G^{31} \frac{\partial \phi}{\partial \xi} + G^{32} \frac{\partial \phi}{\partial \eta} + G^{33} \frac{\partial \phi}{\partial \zeta},
\end{align*}
\]

(5.20)

where \( n^{i} \) is the normal of the \( i^{th} \) face, \( \mathbf{A}^{ij} \) is the \( j^{th} \) area of the \( i^{th} \) face, the geometric diffusion coefficients \( G^{ij} \) are given by

\[
G^{ij} = \frac{\mathbf{A}^{ij} \cdot \mathbf{A}^{ij}}{\Omega^{i}},
\]

(5.21)
and the following approximations are used for the derivatives in computational space

\[
\begin{align*}
\frac{\partial \phi}{\partial \xi}^1 & \approx \phi_E - \phi_P, \\
\frac{\partial \phi}{\partial \eta}^1 & \approx \frac{1}{4} (\phi_N + \phi_{NE} - \phi_S - \phi_{SE}), \\
\frac{\partial \phi}{\partial \zeta}^1 & \approx \frac{1}{4} (\phi_T + \phi_{TE} - \phi_B - \phi_{BE}), \\
\frac{\partial \phi}{\partial \xi}^2 & \approx \frac{1}{4} (\phi_E + \phi_{NE} - \phi_W - \phi_{NW}), \\
\frac{\partial \phi}{\partial \eta}^2 & \approx \phi_N - \phi_P, \\
\frac{\partial \phi}{\partial \zeta}^2 & \approx \frac{1}{4} (\phi_T + \phi_{TN} - \phi_B - \phi_{BN}), \\
\frac{\partial \phi}{\partial \xi}^3 & \approx \frac{1}{4} (\phi_E + \phi_{TE} - \phi_W - \phi_{TW}), \\
\frac{\partial \phi}{\partial \eta}^3 & \approx \frac{1}{4} (\phi_N + \phi_{TN} - \phi_S - \phi_{TS}), \\
\frac{\partial \phi}{\partial \zeta}^3 & \approx \phi_T - \phi_P.
\end{align*}
\]

(5.22)

(5.23)

(5.24)

The areas and cell geometric diffusion coefficients are properties of the faces of the cell. Thus there are area vectors \(A_i^1\) located at both the east and west faces of the cell, with the areas in both cases being defined pointing out of the cell. The area terms \(A_i^1\) for the east face of the cell \(P\) are the negative of the \(A_i^1\) areas of the west face of the cell \(E\). The geometric diffusion coefficient is always positive, and is the same for the face regardless of whether the face is being considered the east face of cell \(P\) or the west face of cell \(E\).

For the implementation of the non-orthogonal mesh solver the face geometric properties for the high side faces of each cell (the east, north and top faces) are calculated and stored. These properties are denoted \(A_i^{2i}, A_i^{3i}, G_i^{1i}, G_i^{2i}\) and \(G_i^{3i}\). The values for the low faces \(A_i^{1i}, G_i^{2i}\) are then taken from the high faces of the neighbouring cells.

Using the approximations for the derivatives normal to the cell faces, the diffusive term in the transport equation (Equation (2.1))

\[
\nabla (\Gamma \nabla \phi),
\]

(5.25)

can be discretised and factorised into a linear system

\[
A \phi_i
\]

(5.26)

where the individual equations can be expanded as

\[
\begin{align*}
a_P \phi_P + a_E \phi_E + a_W \phi_W + a_N \phi_N + a_S \phi_S + a_T \phi_T + a_B \phi_B \\
+ a_{NE} \phi_{NE} + a_{NW} \phi_{NW} + a_{SE} \phi_{SE} + a_{SW} \phi_{SW} \\
+ a_{TN} \phi_{TN} + a_{TS} \phi_{TS} + a_{BN} \phi_{BN} + a_{BS} \phi_{BS} \\
+ a_{TE} \phi_{TE} + a_{TW} \phi_{TW} + a_{BE} \phi_{BE} + a_{BW} \phi_{BW},
\end{align*}
\]

(5.27)
and where the equation coefficients are given by

\[
\begin{align*}
a_E &= G_{e}^{11} + \frac{1}{4}(G_{n}^{21} - G_{s}^{21} + G_{l}^{21} - G_{b}^{21}), \\
a_W &= G_{w}^{11} - \frac{1}{4}(G_{n}^{21} - G_{s}^{21} + G_{l}^{21} - G_{b}^{21}), \\
a_N &= G_{n}^{22} + \frac{1}{4}(G_{e}^{12} - G_{w}^{12} + G_{l}^{32} - G_{b}^{32}), \\
a_S &= G_{s}^{22} - \frac{1}{4}(G_{e}^{12} - G_{w}^{12} + G_{l}^{32} - G_{b}^{32}), \\
a_T &= G_{l}^{33} + \frac{1}{4}(G_{w}^{13} - G_{n}^{23} + G_{b}^{23} - G_{s}^{33}), \\
a_B &= G_{b}^{33} - \frac{1}{4}(G_{w}^{13} - G_{n}^{23} + G_{l}^{33} - G_{s}^{33}), \\
a_P &= G_{e}^{11} + G_{w}^{11} + G_{n}^{22} + G_{s}^{22} + G_{l}^{33} + G_{b}^{33}, \\
a_{NE} &= -\frac{1}{4}(G_{e}^{12} + G_{n}^{21}), \\
a_{NW} &= -\frac{1}{4}(G_{w}^{12} + G_{n}^{21}), \\
a_{SE} &= -\frac{1}{4}(G_{e}^{12} + G_{s}^{21}), \\
a_{SW} &= -\frac{1}{4}(G_{w}^{12} + G_{s}^{21}), \\
a_{TN} &= -\frac{1}{4}(G_{n}^{23} + G_{l}^{32}), \\
a_{TS} &= -\frac{1}{4}(G_{n}^{23} + G_{l}^{32}), \\
a_{BN} &= -\frac{1}{4}(G_{n}^{23} + G_{b}^{32}), \\
a_{BS} &= -\frac{1}{4}(G_{n}^{23} + G_{b}^{32}), \\
a_{TE} &= -\frac{1}{4}(G_{e}^{13} + G_{l}^{31}), \\
a_{TW} &= -\frac{1}{4}(G_{w}^{13} + G_{l}^{31}), \\
a_{BE} &= -\frac{1}{4}(G_{e}^{13} + G_{b}^{31}), \\
a_{BW} &= -\frac{1}{4}(G_{w}^{13} + G_{b}^{31}).
\end{align*}
\]

For a two-dimensional system the slightly more manageable discretisation is

\[
\begin{align*}
&ap \phi_p + a_E \phi_E + a_W \phi_W + a_N \phi_N + a_S \phi_S \\
&+ a_{NE} \phi_{NE} + a_{NW} \phi_{NW} + a_{SE} \phi_{SE} + a_{SW} \phi_{SW},
\end{align*}
\]

with the equation coefficients

\[
\begin{align*}
&ap \phi_p + a_E \phi_E + a_W \phi_W + a_N \phi_N + a_S \phi_S \\
&+ a_{NE} \phi_{NE} + a_{NW} \phi_{NW} + a_{SE} \phi_{SE} + a_{SW} \phi_{SW},
\end{align*}
\]

\[
\begin{align*}
a_E &= G_{e}^{11} + \frac{1}{4}(G_{n}^{21} - G_{s}^{21}), \\
a_W &= G_{w}^{11} - \frac{1}{4}(G_{n}^{21} - G_{s}^{21}), \\
a_N &= G_{n}^{22} + \frac{1}{4}(G_{e}^{12} - G_{w}^{12}), \\
a_S &= G_{s}^{22} - \frac{1}{4}(G_{e}^{12} - G_{w}^{12}), \\
a_P &= G_{e}^{11} + G_{w}^{11} + G_{n}^{22}, \\
a_{NE} &= \frac{1}{4}(G_{e}^{12} + G_{n}^{21}), \\
a_{NW} &= -\frac{1}{4}(G_{w}^{12} + G_{n}^{21}), \\
a_{SE} &= -\frac{1}{4}(G_{e}^{12} + G_{s}^{21}), \\
a_{SW} &= \frac{1}{4}(G_{w}^{12} + G_{s}^{21}).
\end{align*}
\]

The computational molecule for the Cartesian equations given in Section 2.2 contained 5 points for
the two-dimensional mesh, and 7 points in three-dimensions. With the addition of the extra terms for
the non-orthogonal diffusion these molecules have expanded to 9 and 19 members in two- and three-
dimensions respectively. To allow the use of the linear solvers developed for Cartesian meshes, the
system can be calculated using a deferred correction scheme, with the extra diffusive terms from the
$G^{ij, i\neq j}$ terms being absorbed into the source term of the equation. Thus the system in Equation (5.26)
can be rewritten as

\[
A\phi + B\phi,
\]

with $A$ containing the orthogonal components, and $B$ the non-orthogonal components. For a three
dimensional system the equation coefficients are

\[
\begin{align*}
a_E & = G_{e}^{11}, \\
a_W & = G_{w}^{11}, \\
a_N & = G_{n}^{22}, \\
a_S & = G_{s}^{22}, \\
a_T & = G_{t}^{33}, \\
a_B & = G_{b}^{33}, \\
ap & = G_{e}^{11} + G_{w}^{11} + G_{n}^{22} + G_{s}^{22} + G_{t}^{33} + G_{b}^{33}, \\
b_E & = \frac{1}{4}(G_{n}^{21} - G_{s}^{21} + G_{t}^{31} - G_{b}^{31}), \\
b_W & = \frac{1}{4}(G_{n}^{21} - G_{s}^{21} + G_{t}^{31} - G_{b}^{31}), \\
b_N & = \frac{1}{4}(G_{e}^{12} - G_{w}^{12} + G_{t}^{22} - G_{b}^{22}), \\
b_S & = \frac{1}{4}(G_{e}^{12} - G_{w}^{12} + G_{t}^{22} - G_{b}^{22}), \\
b_T & = \frac{1}{4}(G_{e}^{13} - G_{w}^{13} + G_{n}^{23} - G_{s}^{23}), \\
b_B & = \frac{1}{4}(G_{e}^{13} - G_{w}^{13} + G_{n}^{23} - G_{s}^{23}), \\
b_P & = 0, \\
b_{NE} & = \frac{1}{4}(G_{e}^{12} + G_{n}^{21}), \\
b_{NW} & = \frac{1}{4}(G_{w}^{12} + G_{n}^{21}), \\
b_{SE} & = \frac{1}{4}(G_{e}^{12} + G_{s}^{21}), \\
b_{SW} & = \frac{1}{4}(G_{w}^{12} + G_{s}^{21}), \\
b_{TN} & = \frac{1}{4}(G_{n}^{23} + G_{t}^{32}), \\
b_{TS} & = \frac{1}{4}(G_{n}^{23} + G_{t}^{32}), \\
b_{BN} & = \frac{1}{4}(G_{n}^{23} + G_{b}^{32}), \\
b_{BS} & = \frac{1}{4}(G_{n}^{23} + G_{b}^{32}), \\
b_{TE} & = \frac{1}{4}(G_{e}^{13} + G_{t}^{31}), \\
b_{TW} & = \frac{1}{4}(G_{e}^{13} + G_{t}^{31}), \\
b_{BE} & = \frac{1}{4}(G_{e}^{13} + G_{b}^{31}), \\
b_{BW} & = \frac{1}{4}(G_{e}^{13} + G_{b}^{31}),
\end{align*}
\]

(5.32)

whilst for a two dimensional system the terms for the third axis are dropped and the equations are

\[
\begin{align*}
a_E & = G_{e}^{11}, \\
a_W & = G_{w}^{11}, \\
a_N & = G_{n}^{22}, \\
a_S & = G_{s}^{22}, \\
ap & = G_{e}^{11} + G_{w}^{11} + G_{n}^{22} + G_{s}^{22}, \\
b_E & = \frac{1}{4}(G_{n}^{21} - G_{s}^{21}), \\
b_W & = -\frac{1}{4}(G_{n}^{21} - G_{s}^{21}), \\
b_N & = \frac{1}{4}(G_{e}^{12} - G_{w}^{12}), \\
b_S & = -\frac{1}{4}(G_{e}^{12} - G_{w}^{12}), \\
b_P & = 0, \\
b_{NE} & = \frac{1}{4}(G_{e}^{12} + G_{n}^{21}), \\
b_{NW} & = -\frac{1}{4}(G_{w}^{12} + G_{n}^{21}), \\
b_{SE} & = \frac{1}{4}(G_{e}^{12} + G_{s}^{21}), \\
b_{SW} & = \frac{1}{4}(G_{w}^{12} + G_{s}^{21}),
\end{align*}
\]

(5.33)

For an orthogonal mesh the dot product \( \mathbf{A}^{ij} \mathbf{A}^{ij} \) is only non-zero when \( i = j \). Thus for such a mesh
the geometric diffusion coefficients $G^{ij}$ are only non-zero for $i = j$, and Equations (5.20) to (5.24) reduce to the rather more manageable system of

$$\begin{align*}
|A^{11}| \frac{\partial \phi}{\partial n^1} &= G^{11} \frac{\partial \phi}{\partial \xi}, \\
|A^{22}| \frac{\partial \phi}{\partial n^2} &= G^{22} \frac{\partial \phi}{\partial \eta}, \\
|A^{33}| \frac{\partial \phi}{\partial n^3} &= G^{33} \frac{\partial \phi}{\partial \zeta},
\end{align*}$$

(5.34)

with

$$\begin{align*}
\frac{\partial \phi}{\partial \xi}^1 &\approx \phi_E - \phi_P, \\
\frac{\partial \phi}{\partial \eta}^2 &\approx \phi_N - \phi_P, \\
\frac{\partial \phi}{\partial \zeta}^3 &\approx \phi_T - \phi_P.
\end{align*}$$

(5.35) (5.36) (5.37)

For such a system $B = 0$, and the discretised equation can be written as,

$$A \phi,$$

(5.38)

where the equation coefficients for a two dimensional discretisation are given by

$$\begin{align*}
a_E &= G^{11}, \\
a_W &= G^{11}, \\
a_N &= G^{22}, \\
a_S &= G^{22}, \\
a_P &= G^{11} + G^{11}_w + G^{22}_n + G^{22}_s,
\end{align*}$$

(5.39)

whilst for a three dimensional discretisation they become,

$$\begin{align*}
a_E &= G^{11}, \\
a_W &= G^{11}, \\
a_N &= G^{22}, \\
a_S &= G^{22}, \\
a_T &= G^{33}, \\
a_B &= G^{33}, \\
a_P &= G^{11} + G^{11}_w + G^{22}_n + G^{22}_s + G^{33}_t + G^{33}_b.
\end{align*}$$

(5.40)

For an orthogonal mesh that is Cartesian, the geometric diffusion coefficients $G^{ii}$ for the above system are

$$\begin{align*}
G^{11} &= \frac{\Delta y \Delta z}{\delta x_e}, \\
G^{22} &= \frac{\Delta x \Delta z}{\delta x_n}, \\
G^{33} &= \frac{\Delta x \Delta y}{\delta x_t},
\end{align*}$$

(5.41)

and the equations reduce to those given in Section 2.2.
5.3 The Discretisation of the Advective and Source Terms on a Non-Orthogonal Grid

It now only remains to discretise the advection and source terms to complete the discretisation of the transport equation on a non-orthogonal mesh. The source terms are treated in an identical manner to that used for a Cartesian mesh, with the cell centre value being taken as the average for the cell as a whole, and so the integral for the source over the cell is approximated as

\[
\int_{\Omega} S d\Omega \approx S_{f}\Omega. \tag{5.42}
\]

The advective fluxes in the transport equation

\[
\nabla (\rho u \phi), \tag{5.43}
\]

are approximated in the same manner as was used with the Cartesian discretisations given in Section 2.2, with the face value of \( \phi \) being interpolated on the regular mesh in computational space. The only difference between the Cartesian and non-orthogonal geometry discretisation is the calculation of the face mass flux. For the Cartesian mesh the mass flux is calculated as \( \rho A u \) where \( A \) is the area and \( u \) is the velocity normal to the face. For the non-orthogonal mesh the mass flux is instead calculated from the dot product of the velocity and the face area normal, and so all velocity components must be interpolated to the faces. The mass flux for east, north and top faces are then given by,

\[
\begin{align*}
    m_e &= \rho \langle u_e A_e^1 \rangle, \\
    m_n &= \rho \langle u_n A_n^2 \rangle, \\
    m_t &= \rho \langle u_t A_t^3 \rangle.
\end{align*} \tag{5.44}
\]

For a Cartesian mesh these expressions reduce to

\[
\begin{align*}
    m_e &= \rho u_e A_e, \\
    m_n &= \rho u_n A_n, \\
    m_t &= \rho u_t A_t,
\end{align*} \tag{5.45}
\]

which were the expressions given in Equation (2.17).

The full discretisation of the transport equation on a non-orthogonal mesh can now be given. By approximating the advective fluxes with the central difference scheme, which was described for a Cartesian mesh in Section 2.2.3, the steady transport equation can be discretised as

\[
A \phi + B \phi = c. \tag{5.46}
\]

For a discretisation of the two-dimensional transport equation the individual coefficients of the \( A, B \)}
and c arrays are,

\[
\begin{align*}
 a_E &= -G_{e}^{11} + \frac{1}{2}m_e, \\
 a_W &= -G_{w}^{11} - \frac{1}{2}m_w, \\
 a_N &= -G_{n}^{22} + \frac{1}{2}m_n, \\
 a_S &= -G_{s}^{22} - \frac{1}{2}m_s, \\
 a_T &= -G_{t}^{33} + \frac{1}{2}m_t, \\
 a_B &= -G_{b}^{33} - \frac{1}{2}m_b, \\
 a_P &= -(a_E + a_W + a_N + a_S) + (m_e - m_w + m_n - m_s), \\
 c &= S_P \Omega, \\
 b_E &= \frac{1}{4}(G_{n}^{21} - G_{s}^{21}), \\
 b_W &= -\frac{1}{4}(G_{n}^{21} - G_{s}^{21}), \\
 b_N &= \frac{1}{4}(G_{e}^{12} - G_{w}^{12}), \\
 b_S &= -\frac{1}{4}(G_{e}^{12} - G_{w}^{12}), \\
 b_P &= 0, \\
 b_{NE} &= \frac{1}{4}(G_{e}^{12} + G_{n}^{21}), \\
 b_{NW} &= -\frac{1}{4}(G_{e}^{12} + G_{n}^{21}), \\
 b_{SE} &= -\frac{1}{4}(G_{e}^{12} + G_{s}^{21}), \\
 b_{SW} &= \frac{1}{4}(G_{e}^{12} + G_{s}^{21}).
\end{align*}
\]

For a discretisation of the three dimensional transport equation the equation coefficients are given by
\[ a_E = -G_e^{11} + \epsilon m_e, \]
\[ a_W = -G_w^{11} - \epsilon m_w, \]
\[ a_N = -G_n^{22} + \epsilon m_n, \]
\[ a_S = -G_s^{22} - \epsilon m_s, \]
\[ a_T = -G_t^{33} + \epsilon m_t, \]
\[ a_B = -G_b^{33} - \epsilon m_b, \]
\[ a_P = -(a_E + a_W + a_N + a_S + a_T + a_B) \]
\[ + (m_e - m_w + m_n - m_s + m_t - m_b), \]
\[ c = S_p \Omega, \]
\[ b_E = -\frac{1}{4} (G_n^{21} - G_s^{21} + G_t^{31} - G_b^{31}), \]
\[ b_W = -\frac{1}{4} (G_n^{21} - G_s^{21} + G_t^{31} - G_b^{31}), \]
\[ b_N = -\frac{1}{4} (G_e^{12} - G_w^{12} + G_t^{32} - G_b^{32}), \]
\[ b_S = -\frac{1}{4} (G_e^{12} - G_w^{12} + G_t^{32} - G_b^{32}), \]
\[ b_T = -\frac{1}{4} (G_e^{13} - G_w^{13} + G_t^{33} - G_b^{33}), \]
\[ b_B = -\frac{1}{4} (G_e^{13} - G_w^{13} + G_t^{33} - G_b^{33}), \]
\[ b_P = 0, \]
\[ b_{NE} = -\frac{1}{4} (G_e^{12} + G_s^{21}), \]
\[ b_{NW} = -\frac{1}{4} (G_e^{12} + G_s^{21}), \]
\[ b_{SE} = -\frac{1}{4} (G_e^{12} + G_s^{21}), \]
\[ b_{SW} = +\frac{1}{4} (G_e^{12} + G_s^{21}), \]
\[ b_{TN} = +\frac{1}{4} (G_n^{23} + G_t^{32}), \]
\[ b_{TS} = -\frac{1}{4} (G_n^{23} + G_t^{32}), \]
\[ b_{BN} = +\frac{1}{4} (G_n^{23} + G_b^{32}), \]
\[ b_{BS} = -\frac{1}{4} (G_n^{23} + G_b^{32}), \]
\[ b_{TE} = +\frac{1}{4} (G_e^{13} + G_t^{31}), \]
\[ b_{TW} = -\frac{1}{4} (G_e^{13} + G_t^{31}), \]
\[ b_{BE} = -\frac{1}{4} (G_e^{13} + G_b^{31}), \]
\[ b_{BW} = +\frac{1}{4} (G_e^{13} + G_b^{31}). \]

These systems may be compared to the equivalent systems given in Equations (2.25) and (2.26) for a discretisation upon a Cartesian mesh. The system is solved using a deferred correction scheme, with the \( B \) array being multiplied by the previous iterations value of the \( \phi \) field and absorbed into the right hand side, so that at any given iteration system being solved by the linear solver is

\[ \mathbf{A} \phi = \mathbf{c} - B \phi^{old}, \]

where \( \phi^{old} \) is the previous iterations solution.

For an orthogonal mesh \( B = 0 \), and so the two and three dimensional transport equations reduce to

\[ \mathbf{A} \phi = \mathbf{c}, \]
where the coefficients for a two-dimensional discretisation are
\[
\begin{align*}
    a_E &= -G_{e}^{11} + \frac{1}{2}m_e, \\
    a_W &= -G_{w}^{11} - \frac{1}{2}m_w, \\
    a_N &= -G_{n}^{22} + \frac{1}{2}m_n, \\
    a_S &= -G_{s}^{22} - \frac{1}{2}m_s, \\
    a_T &= -G_{t}^{33} + \frac{1}{2}m_t, \\
    a_B &= -G_{b}^{33} - \frac{1}{2}m_b, \\
    a_P &= -(a_E + a_W + a_N + a_S + a_T + a_B) + (m_e - m_w + m_n - m_s), \\
    c &= S_{p}\Omega,
\end{align*}
\] (5.51)

and for a three-dimensional discretisation, the equation coefficients are given by
\[
\begin{align*}
    a_E &= -G_{e}^{11} + \frac{1}{2}m_e, \\
    a_W &= -G_{w}^{11} - \frac{1}{2}m_w, \\
    a_N &= -G_{n}^{22} + \frac{1}{2}m_n, \\
    a_S &= -G_{s}^{22} - \frac{1}{2}m_s, \\
    a_T &= -G_{t}^{33} + \frac{1}{2}m_t, \\
    a_B &= -G_{b}^{33} - \frac{1}{2}m_b, \\
    a_P &= -(a_E + a_W + a_N + a_S + a_T + a_B) + (m_e - m_w + m_n - m_s + m_t - m_b), \\
    c &= S_{p}\Omega.
\end{align*}
\] (5.52)

### 5.4 Boundary Conditions for a Non-Orthogonal Mesh

The boundary conditions for a discretisation on a non-orthogonal mesh are similar to those for a Cartesian mesh, except for Neumann boundaries which have extra cross derivative components in the calculation of the gradient normal to the boundary.

Considering the case for an eastern boundary, as illustrated in Figure 5.7, a Dirichlet boundary condition where the value of the scalar is specified at the boundary as
\[
\phi = \phi_b, \tag{5.53}
\]
results in an equation for the boundary node of the form
\[
\phi_E = 2\phi_b - \phi_P, \tag{5.54}
\]
which can be substituted into the equation for point \(P\) given in Equations (5.27) and (5.29).

For Neumann boundary conditions the gradient normal to the face is specified,
\[
\frac{\partial \phi}{\partial m} = \phi'_b, \tag{5.55}
\]
The normal gradient is given in Equation (5.20) above, and this is substituted into the system at the boundary nodes. The non-orthogonal terms involving cross-derivatives at the faces are normally solved using the deferred correction scheme as was given in Equation (5.31), and so at any iteration the boundary gradient may not be as specified by Equation (5.55), but for a fully converged solution the boundary condition will apply.

For discontinuities in the boundary conditions, such as a step change in the specified boundary value for Dirichlet boundaries along a face, a change from Neumann to Dirichlet boundaries on a face, or a
corner of the mesh, the central difference approximation for the cross derivatives in the mesh (such as those given in the second and third equations in Equation (5.10)) will be in error, and instead of using central differences, one sided differences should be used as appropriate.

5.5 The Solution of the Navier–Stokes Equations on a Non-Orthogonal Mesh

For solving the Navier–Stokes equations on a non-orthogonal mesh there are three more issues to consider. The first concerns the discretisation of the pressure term in the momentum equations, the other terms in the equation being treated in the same manner as their counterparts in the generic transport equation. The second issue is the pressure-velocity coupling and the generation of an equation for the pressure correction. And finally the interpolation scheme used to calculated the velocity at the cell faces needs to be formulated.

5.5.1 Evaluation of the Pressure Gradient Term

The pressure force on a cell can be approximated by the integral of the pressure over the faces of the cell, the pressure force on each face acting normal to that face. The pressure at the face can be approximated by an interpolation of the values at the centre of the cells on either side of the face. Thus the value at the eastern face can be approximated as,

\[ p_e \approx \frac{p_P + p_E}{2}. \] (5.56)

The area of the eastern face is given above as \( A_{e11} \), and so the pressure force on the eastern face is approximated by

\[ p_e = -A_{e11} \left( \frac{p_P + p_E}{2} \right), \] (5.57)

the negative sign being due to the pressure force acting inwards on the cell, whilst the face area is defined pointing outwards. Integrating over the six faces of the cell gives the pressure force on the cell as

\[ p = -\frac{1}{2} \left( A_{e11} (p_P + p_E) + A_{w11} (p_P + p_W) \right. \\
+ A_{n22} (p_P + p_N) + A_{e22} (p_P + p_S) \\
+ A_{e33} (p_P + p_T) + A_{w33} (p_P + p_B) \right). \] (5.58)
This can be factorised to
\[ p = -\frac{1}{2} \left( p_E A_e^{11} + p_W A_w^{11} + p_N A_n^{22} + p_S A_s^{22} + p_T A_t^{33} + p_B A_b^{33} \right) \]
\[ - \frac{1}{2} \rho_p \left( A_e^{11} + A_w^{11} + A_n^{22} + A_s^{22} + A_t^{33} + A_b^{33} \right). \]  

(5.59)

The sum of the areas on the second line of Equation (5.59) will be zero for a closed volume, and the expression reduces to
\[ p = -\frac{1}{2} \left( p_E A_e^{11} + p_W A_w^{11} + p_N A_n^{22} + p_S A_s^{22} + p_T A_t^{33} + p_B A_b^{33} \right). \]

(5.60)

The individual Cartesian components of this force can then be added to the source term for each component of the momentum equation.

### 5.5.2 Interpolation of the Face Velocities

The velocity interpolation at the faces is performed in an identical manner to the method used for Cartesian meshes, described in Section 4.2.2. The method, attributed to Rhie and Chow[139], interpolates the sum of the velocities and pressure forces, rather than interpolating the velocities alone.

The equation for the conservation of momentum at the point \( P \) can be written in vector form as,
\[ \mathbf{u}_P - \frac{1}{\alpha_P} \sum_f A_f \cdot \mathbf{p}_f = \frac{\mathbf{S}}{\alpha_P} - \frac{1}{\alpha_P} \sum_{n=nb} a_n \mathbf{u}_n, \]

(5.61)

where \( \mathbf{u}_P \) is the velocity vector at the point \( P \), \( \sum_f A_f \cdot \mathbf{p}_f \) is the sum of pressure forces on the faces of the cell centred at \( P \), and \( \alpha_P \) is the diagonal element of the system of linear equations \( \sum_{n=nb} a_n \mathbf{u}_n \). Such an equation exists for each cell in the domain, and with Rhie–Chow interpolation it is assumed that a similar equation can be approximated at the faces of a cell. Thus for the eastern face of a cell there is an equation of the form,
\[ \mathbf{u}_e - \left( \frac{1}{\alpha_P} \right) \left( \sum_f A_f \cdot \mathbf{p}_f \right)_e = \left( \frac{\mathbf{S}}{\alpha_P} \right)_e - \left( \frac{1}{\alpha_P} \sum_{n=nb} a_n \mathbf{u}_n \right)_e, \]

(5.62)

with similar equations being able to be written for the north and top faces of the cell. The left hand side of equation (5.62) at the face of the cell is assumed to be equal to a linear interpolation of the left hand sides of the equations for the \( E \) and \( P \) cells on either side of the face,
\[ \mathbf{u}_e - \left( \frac{1}{\alpha_P} \right) \left( \sum_f A_f \cdot \mathbf{p}_f \right)_e = \overline{\mathbf{u}_e} - \left( \frac{1}{\alpha_P} \sum_f A_f \cdot \mathbf{p}_f \right)_e, \]

(5.63)

where an overbar \( \overline{\mathbf{u}_e} \) signifies a linear interpolation of the values at the \( P \) and \( E \) cell centres. The value of \( \frac{1}{\alpha_P} \) at the eastern face in the left hand side of Equation (5.63) can be approximated by a linear interpolation, and so an expression for the velocity at the eastern face is
\[ \mathbf{u}_e = \overline{\mathbf{u}_e} + \left( \frac{1}{\alpha_P} \right) \left( \sum_f A_f \cdot \mathbf{p}_f \right)_e - \left( \frac{1}{\alpha_P} \sum_f A_f \cdot \mathbf{p}_f \right)_e. \]

(5.64)

The sum of the pressure forces at the eastern face is approximated by
\[ \left( \sum_f A_f \cdot \mathbf{p}_f \right)_e = \left( A_e^{11} - A_e^{11} \right) + \frac{1}{4} \left( A_e^{22} + A_{ne}^{n} + A_{ne}^{n} - A_s^{22} + A_{ne}^{n} - A_s^{22} \right) \]
\[ + \frac{1}{4} \left( A_t^{33} + A_{ne}^{n} + A_{ne}^{n} - A_b^{33} + A_{ne}^{n} - A_b^{33} \right). \]

(5.65)
where $A_{eE}^{22}$ is the face area $A_{e}^{22}$ for the cell $E$. The pressure forces on the cells $P$ and $E$ are given by Equation (5.60), and so all the terms in Equation (5.64) are defined and the face velocity can be calculated. The velocities for the north and top faces can be interpolated using similar expressions, and so the face mass fluxes can be calculated using Equation (5.44),

$$
\begin{align*}
m_e &= \rho \left( A_{eE}^{11} u_e \right), \\
m_n &= \rho \left( A_{eE}^{22} u_n \right), \\
m_t &= \rho \left( A_{eE}^{33} u_t \right).
\end{align*}
$$

### 5.5.3 The Non-Orthogonal Pressure Correction Equation

The coupling of the velocity and pressure equations on a non-orthogonal mesh is performed in an identical manner to that described in Section 4.2.1 for Cartesian systems. The discretisation of the Navier–Stokes has been described above, as has the method used for interpolating the velocity to the cell faces. The only remaining task is the derivation of the pressure and velocity corrections.

As was described in Section 4.2.1 the SIMPLE coupling scheme involves the calculation of an initial estimate of the velocity field, $u^*$, from which the face velocities are interpolated. The generated field will not generally satisfy the continuity equation, and so a pressure correction $p'$ is then calculated, which when added to the momentum equations results in a velocity correction $u'$ which which satisfies the continuity equation.

The initial calculation of the velocity field is with an equation of the form of

$$
a_P u_P^* + \sum_n a_n u_n^* = S + \sum_f A_f p_f. \tag{5.67}
$$

where $n$ signifies a summation of the neighbouring cells, whilst $f$ is a summation over the cell faces. The solution to this equation will not generally satisfy the continuity equation. We therefore wish to add the velocity correction $u'$ generated from a pressure correction $p'$, such that the resulting velocity field $u^* + u'$ satisfies the continuity equation.

Substituting $p + p'$ and $u^* + u'$ into the momentum Equation (5.67) gives

$$
a_P (u_P^* + u_P') + \sum_n a_n (u_n^* + u_n') = S + \sum_f A_f (p_f + p_f'). \tag{5.68}
$$

Approximating the off-diagonal terms in the momentum equation by Equation (5.67) as

$$
\sum_n a_n (u_n^* + u_n') \approx \sum_n a_n u_n^*, \tag{5.69}
$$

and then subtracting the momentum Equation (5.67) gives

$$
a_P u_P' = \sum_f A_f p_f', \tag{5.70}
$$

or

$$
u_P' = \frac{1}{a_P} \sum_f A_f p_f'. \tag{5.71}
$$

The correction velocities for the east, north and top faces of the cell are then found by interpolation as

$$
\begin{align*}
u_e' &= \left( \frac{1}{a_P} \right)_e \left( \sum_f A_f p_f' \right)_e, \\
u_n' &= \left( \frac{1}{a_P} \right)_n \left( \sum_f A_f p_f' \right)_n, \\
u_t' &= \left( \frac{1}{a_P} \right)_t \left( \sum_f A_f p_f' \right)_t. \tag{5.72}
\end{align*}
$$
Following the derivation of Rhie and Chow [139] the cross derivative terms at the faces are ignored, since for near-orthogonal meshes they are small, and the \( p' \) field vanishes for a converged solution. Therefore the expressions for the face velocity corrections can be written as

\[
\mathbf{u}'_e = \left( \frac{1}{a_{Pe}} \right) A_{e11} p_E' - p_P',
\]

\[
\mathbf{u}'_n = \left( \frac{1}{a_{Pn}} \right) A_{n22} p_N' - p_P',
\]

\[
\mathbf{u}'_t = \left( \frac{1}{a_{Pt}} \right) A_{t33} p_T' - p_P'.
\]

The discretised continuity equation can be written as

\[
m_e - m_w + m_n - m_s + m_t - m_b = 0.
\]

Substituting in \( m_f = m_f^* + m_f' \), where \( m_f^* \) is the interpolated velocity from the \( \mathbf{u}^* \) velocity field, and \( m_f' \) is the mass flux correction defined as \( m_f' = \rho A_f \mathbf{u}'_f \), and rearranging gives

\[
\begin{align*}
\rho A_{e11} \cdot A_{e11} \frac{p_E' - p_P'}{a_{Pe}} & - \rho A_{w11} \cdot A_{w11} \frac{p_W' - p_P'}{a_{Pw}} + \rho A_{n22} \cdot A_{n22} \frac{p_N' - p_P'}{a_{Pn}} - \rho A_{s22} \cdot A_{s22} \frac{p_S' - p_P'}{a_{Ps}} \\
+ \rho A_{t33} \cdot A_{t33} \frac{p_T' - p_P'}{a_{Pt}} & - \rho A_{b33} \cdot A_{b33} \frac{p_B' - p_P'}{a_{Pb}} = m_w - m_e + m_s - m_n + m_b - m_t.
\end{align*}
\]

This can be factorised as

\[
b_E p_E' + b_E p_E' + b_W p_W' + b_N p_N' + b_S p_S' + b_T p_T' + b_B p_B' = c,
\]

where

\[
\begin{align*}
b_E & = \frac{A_{e11} \cdot A_{e11}}{a_{Pe}}, \\
b_W & = \frac{A_{w11} \cdot A_{w11}}{a_{Pw}}, \\
b_N & = \frac{A_{n22} \cdot A_{n22}}{a_{Pn}}, \\
b_S & = \frac{A_{s22} \cdot A_{s22}}{a_{Ps}}, \\
b_T & = \frac{A_{t33} \cdot A_{t33}}{a_{Pt}}, \\
b_B & = \frac{A_{b33} \cdot A_{b33}}{a_{Pb}}, \\
b_P & = -(b_E + b_W + b_N + b_S + b_T + b_B), \\
c & = (m_w - m_e + m_s - m_n + m_b - m_t).
\end{align*}
\]

This system can be solved for the pressure correction \( p' \), which is then used to correct the velocities to satisfy the continuity equation, the velocity and pressure fields being updated by

\[
\begin{align*}
\mathbf{u}^{\text{new}} & = \mathbf{u}^* + \mathbf{u}', \\
p^{\text{new}} & = p + p'.
\end{align*}
\]

Aside from the use of velocity vectors at cell faces, rather than single Cartesian components of velocity, and the use of area vectors rather than the magnitudes of the cell face areas, the system is identical to Equation (4.23) derived for a Cartesian system in Section 4.2.1.
5.6 Examples of Non-Orthogonal Problems

To test the implementation of the non-orthogonal differencing schemes the code was used to solve three test problems. The first models two-dimensional conduction in a skew domain. For the geometry chosen a Cartesian finite difference solution was easily calculated that could be used as a comparison for the solution calculated using the non-orthogonal finite volume discretisation.

The second problem modelled the two-dimensional driven cavity flow described in Section 4.4.1. The flow was solved for a square cavity, with one solution being generated on a square mesh, and the other on a non-orthogonal mesh that fitted the cavity. As the mesh is refined the solutions calculated on the Cartesian and non-orthogonal meshes should both converge towards the benchmark solution provided by Ghia et al[50].

The final test case was the driven cavity flow provided as a benchmark test case by Demirdžić, Lilek and Perić[36]. This problem models a driven flow in a skew cavity, similar in shape to the domain used in the conduction test.

5.6.1 Conduction in a Skew Domain

To test the discretisation of the diffusion terms of the transport equation, and the implementation of Dirichlet and Neumann boundary conditions, the initial problem modelled was one of steady state conduction in a skew domain. The geometry of the problem is given in Figure 5.8 along with the two sets of boundary conditions used.

![Figure 5.8: The steady state conduction in a skew domain problem, showing the problem geometry and the two boundary conditions modelled. The cavity is of unit length and height, with the side walls angled at 45°.](image-url)
The first problem applies Dirichlet conditions on all boundaries, with the lid of the domain being set to \( \phi = 1 \), whilst the other three boundaries are set to \( \phi = 0 \). The second test imposed Dirichlet conditions on the top and bottom boundaries of the domain, with values of \( \phi = 1 \) and \( \phi = 0 \) respectively, whilst the two side boundaries had a zero flux Neumann boundary condition imposed, \( \frac{\partial \phi}{\partial n} = 0 \), with \( n \) being the direction normal to the surface.

To provide a benchmark solution, the problem was also modelled using a finite difference discretisation on a Cartesian mesh. The two types of meshes that were used are shown in Figure 5.9, with the Cartesian mesh used by the finite difference solver being on the left, whilst the boundary fitted non-orthogonal mesh used by the finite volume solver is shown on the right.

The solutions calculated by the two methods are shown in Figures 5.10 and 5.11, the Cartesian finite difference solutions being calculated on a \( 131^2 \) mesh, whilst the non-orthogonal finite volume solutions are calculated on a \( 51^2 \) mesh. The contours of the two solutions look similar for both the Neumann and Dirichlet problems.
To better compare the two methods, the temperature profiles for each problem, calculated on $131^2$ meshes, are given in Figures 5.12 and 5.13. Figure 5.12 shows the temperature profile across the domain along the horizontal centreline (i.e. midway up the domain) calculated for the Dirichlet boundary condition problem. The finite volume solution is shown as a solid line, with that calculated using the finite difference scheme being given by the crosses. As can be seen the two solutions agree quite well. Similarly Figure 5.13 shows the solution for the Neumann boundary conditions along the centreline running up the centre of the cavity. Again the two methods are in excellent agreement with each other.

Figure 5.12: The calculated temperature profile along the horizontal centreline of the Dirichlet skew conduction problem.

Finally, the convergence of the solutions calculated by the two methods as the mesh is refined are shown in Figures 5.14 and 5.15. For the Dirichlet problem, the temperature at the centre of the domain is plotted as a function of the cell size, whilst for the Neumann problem the temperature at the mid-height of the left wall is plotted. For both problems the two solutions converge as the mesh is refined. No ranking of the relative accuracy of the two schemes can be made, since for the Neumann problem the finite volume solution is the more accurate, whilst for the Dirichlet problem the reverse is true.

To summarise, for the problems tested the non-orthogonal finite volume scheme gives a solution that is comparable in accuracy to that generated by a Cartesian finite difference method. As the mesh is refined the solutions given by both schemes converge to a common solution.
Figure 5.13: The calculated temperature profile along the vertical centreline of the Neumann skew conduction problem.

Figure 5.14: The convergence of the solutions calculated with the finite difference and finite volume solvers as the mesh is refined. The temperature in the centre of the domain for the Dirichlet problem is plotted as a function of the grid size $\Delta x$. 
Figure 5.15: The convergence of the solutions calculated with the finite difference and finite volume solvers as the mesh is refined. The wall temperature at the mid-height of the domain for the Neumann problem is plotted as a function of the grid size $\Delta x$. 
5.6.2 Driven Cavity Flow with a Distorted Mesh

The second problem that was modelled was the Driven Cavity flow for a square domain, previously encountered in Section 4.4.1. For this flow there exist a number of benchmark solutions, and a number of solutions for the flow were calculated in the test described in Chapter 4.

To test the non-orthogonal discretisations used in the flow solver, the problem was modelled on a distorted mesh, as shown in Figure 5.17. Solutions were also calculated on a Cartesian mesh for comparison, and the benchmark solution of Ghia, Ghia and Shin[50] was also used as a test of the solution. Flow was modelled on a range of meshes, for a flow with a Reynolds number of 1000. Solutions were calculated using the MSOU and FOU differencing schemes.

Figure 5.16: The driven cavity problem.

Figure 5.17: The meshes used for the driven cavity problem. On the left the Cartesian mesh, on the right the distorted non-orthogonal mesh.

The solution for a flow calculated on an $81^2$ non-orthogonal mesh using the MSOU differencing scheme is shown in Figure 5.18, with contour plots of the streamfunction $\psi$, the magnitude of velocity $|\mathbf{u}|$, and the $u$ and $v$ components of velocity being given. Contour levels are identical to those used in Section 4.4.1 and the paper of Ghia et al.
For the bulk of the flow the agreement between the Cartesian and non-orthogonal solutions is extremely good, with the contours in Figure 5.18 overlaying those given in Figure 4.5. However the recirculation bubbles in the lower left and right corners of the cavity are incorrectly resolved, with the reattachment points being incorrectly located. This problem occurs with solutions calculated using the FOU, MSOU and QUICK differencing schemes, and the Author is uncertain if this is due to the approximations used in the non-orthogonal discretisation, or is a mesh dependent error that would disappear with mesh refinement.

The velocities on the vertical and horizontal centrelines are shown in Figures 5.19 and 5.20, calculated using the non-orthogonal meshes using the MSOU differencing scheme. Solutions are given for a range of mesh densities, with the solutions being compared with the benchmark velocities of Ghia et al. As with the Cartesian case the calculated velocities converge to the benchmark solution, giving an excellent agreement with the finer mesh solutions.

Finally the velocity fields calculated using the Cartesian and non-orthogonal meshes are compared in Figure 5.21. For each mesh size used, the Cartesian velocity fields were interpolated onto the distorted mesh using bicubic interpolation, with the RMS error between the Cartesian and non-orthogonal solutions being calculated. The error between the two solutions velocity fields is plotted as a function of the mean cell size $\Delta x$ for solutions calculated using the FOU and MSOU differencing schemes. As the mesh is refined the RMS error decreases and the two solutions converge. The error between the MSOU solutions seems larger than those between the FOU solutions. However, as the mesh is refined the MSOU error reduces at a faster rate than that of the FOU solutions.
Figure 5.18: The solution field for the driven cavity calculated on a $81^2$ cell distorted mesh, for a Reynolds number of 1000 using MSOU differencing. From left to right, top to bottom, the fields shown are the stream function, velocity magnitude $|u|$, $u$ velocity and $v$ velocity. Streamfunction contours are at 0.01 intervals for $-0.11 < \psi < 0$, with further contours at $\pm 10^{-3}, \pm 10^{-4}, \pm 10^{-5}, \pm 10^{-6}$ and $\pm 10^{-7}$. Velocity contours are at 0.05 intervals between $\pm 1$. The solution should be compared with those given in Figures 4.4 and 4.5.
Figure 5.19: The profile of the $u$ velocity along the vertical centreline calculated on the distorted mesh at a Reynolds number of 1000 using MSOU differencing. Calculated data compared with the benchmark of Ghia, Ghia and Shin[50].

Figure 5.20: The profile of the $v$ velocity along the horizontal centreline calculated on the distorted mesh at a Reynolds number of 1000 using MSOU differencing. Calculated data compared with the benchmark of Ghia, Ghia and Shin[50].
Figure 5.21: The RMS error between the solutions calculated on a Cartesian and distorted mesh, as a function of average mesh size $\Delta x$. Errors given for $u$ and $v$ velocity fields, for solutions calculated with FOU and MSOU differencing schemes.
5.6.3 Driven Cavity Flow in a Skew Cavity

As a final test, the non-orthogonal code was compared to the benchmark solution given by Demirdžić, Lilek and Perić[36] for a driven cavity flow in a skew cavity. The geometry of the problem is given in Figure 5.22, and solutions were obtained for a cavity with unit length walls ($\ell = h = 1$), a wall angle of $45^\circ$, at a Reynolds number of 1000, using the QUICK differencing scheme.

![Figure 5.22: The skew driven cavity problem.](image)

The solution field calculated using a mesh of $131 \times 131$ points is shown in Figure 5.24, with contour plots of streamfunction, velocity magnitude, and the $u$ and $v$ components of velocity being given. The contour levels used are identical to those used by Demirdžić et al in their paper, and the agreement between the published and calculated solutions is excellent.

![Figure 5.23: The mesh used for the calculation of the skew driven cavity flow field.](image)

The velocities calculated on the vertical and horizontal centrelines are plotted in Figure 5.26, with the solutions being given for a number of meshes. They are compared with the benchmark values of Demirdžić et al, and again the agreement is excellent on the finer meshes. In addition the errors in the coarser mesh solutions show the same trends as the coarse mesh solutions of Demirdžić et al.
Figure 5.24: The solution field for the skew driven cavity, calculated at a Reynolds number of 1000 using QUICK differencing. From left to right, top to bottom, the fields shown are the stream function, velocity magnitude $|u|$, $u$ velocity and $v$ velocity. Streamfunction contours are at 0.0059375 intervals between $-0.0475$ and 0, and at 0.002 intervals between 0 and 0.008. Velocity contours are at 0.05 intervals between $\pm 1$.

Figure 5.25: The profile of the $u$ velocity along the centreline for the skew driven cavity, calculated for a Reynolds number of 1000 using QUICK differencing. Calculated data compared with the benchmark of Demirdžić et al[36].
Figure 5.26: The profile of the $v$ velocity along the centreline for the skew driven cavity, calculated for a Reynolds number of 1000 using QUICK differencing. Calculated data compared with the benchmark of Demirdžić et al[36].
5.7 Conclusions

The finite volume discretisations given in Chapter 2, and the SIMPLE pressure–velocity coupling scheme described in Chapter 4 have been extended from Cartesian meshes to structured non-orthogonal meshes. The discretisation of the diffusion terms in the transport equation has been derived, whilst the discretisation of the advective terms is handled in an identical manner to the discretisations on Cartesian meshes described in Chapter 2. The pressure–velocity coupling scheme was then adapted to non-orthogonal meshes, with the pressure correction equation being re-derived.

The switch from a Cartesian to a non-orthogonal mesh complicates the calculation of the geometric properties of the mesh, such as face areas and cell volumes, and so the techniques used for calculating these values is described in Section 5.1. In addition, the switch from an orthogonal to a non-orthogonal mesh increases the number of terms in the linear equations generated by the finite volume discretisation, with the number of terms increasing from 5 to 9 for two-dimensional discretisations, and from 7 to 19 in discretisations on three-dimensional meshes. A deferred correction scheme that allows the use of the linear solvers developed in Chapter 3 is described and used.

Finally, to test the code it was used to solve three benchmark problems. The first, a two-dimensional conduction problem, tested the discretisation and implementation of the diffusion terms in the transport equation, and the implementation of the Neumann and Dirichlet boundary conditions. The second and third problems modelled two-dimensional driven cavity flows, with the results being compared against published benchmarks. In all cases the agreement between the non-orthogonal code and the benchmarks is excellent, with convergence being demonstrated to occur with refinement of the mesh.