The equations we need to solve, are then

\[
\begin{align*}
\frac{\partial}{\partial x} \rho U \frac{\partial U}{\partial x} + \frac{\partial}{\partial y} \rho V \frac{\partial U}{\partial y} & = -\frac{\partial P}{\partial x} + \frac{\partial}{\partial y} \left( \mu \frac{\partial U}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial U}{\partial y} \right) + S_u \\
\frac{\partial}{\partial x} \rho V \frac{\partial V}{\partial x} + \frac{\partial}{\partial y} \rho U \frac{\partial V}{\partial y} & = -\frac{\partial P}{\partial y} + \frac{\partial}{\partial x} \left( \mu \frac{\partial V}{\partial y} \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial V}{\partial y} \right) + S_v
\end{align*}
\]

(1a) and

\[
\frac{\partial}{\partial x} \rho \frac{\partial U}{\partial x} + \frac{\partial}{\partial y} \rho \frac{\partial V}{\partial y} = 0
\]

(1b)

where \( S_u \) and \( S_v \) represent any other source terms that may be present (from buoyancy, rotation, turbulent stresses, etc).

Before considering how to obtain the pressure field, we first examine alternative storage arrangements that can be used for the discretized flow variables, since these have a bearing on how source terms are represented, and thus on how the schemes to be described may be implemented.

### Pressure-Velocity Coupling

T. J. Craft

George Begg Building, C41

Contents:

- Introduction to CFD
- Numerical solution of equations
- Finite difference methods
- Finite volume methods
- Pressure-velocity coupling
- Solving sets of linear equations
- Unsteady problems
- Turbulence and other physical modelling
- Body-fitted coordinate systems

Reading:

J. Ferziger, M. Peric, *Computational Methods for Fluid Dynamics*


S.V. Patankar, *Numerical Heat Transfer and Fluid Flow*

Notes: [http://cfd.mace.manchester.ac.uk/tmcfd](http://cfd.mace.manchester.ac.uk/tmcfd)

Overview

Up to now, we have mainly considered discretization schemes, and how they are used for a single equation.

In practical flow solvers, we have to solve a set of coupled pde’s. In 3-D, this involves determining \( U, V, W \) and \( P \), as well as any additional quantities (temperature, enthalpy, turbulence energy, etc) that may be required in the problem.

We have already examined how to discretize the momentum equations and solve them to obtain \( U, V, W, P \). As explained, this can be done in an iterative fashion, employing under-relaxation if necessary, to account for the coupling between the equations.

However, we do not have a transport equation for the pressure \( P \). Instead, the 4th equation we have is the continuity equation – which does not even contain the pressure.

In this lecture we are thus going to consider a strategy for handling the coupling between the velocity and pressure fields in a flow computation.

For now, we restrict attention to 2-D flow, consider a steady-state flow, and work within a finite volume framework on a simple rectangular mesh. This is sufficient to demonstrate how schemes for coupling the pressure and velocity can be developed, and the same methods can readily be extended to 3-D and more complex problems.

Discretization

The \( U \) momentum equation can be integrated over the control volume shown below, resulting in a discretized equation of the form

\[
a_\text{up} a_{\text{up}} U_p = \sum a_i^x U_i + S_{\text{up}} + S_u
\]

(2)

where \( S_{\text{up}} \) are the source terms arising from integrating the pressure gradient over the cell, and \( S_u \) represents any other source terms.

One convenient arrangement for storing the discretized variables is to use a single set of control volumes, so all variables are stored at the same locations (colocated).

This has the advantage that all geometrical data only needs to be stored once.

However, as outlined below, this arrangement does also have some disadvantages.

If \( P \) is stored at the same position as \( U \), we need to interpolate between \( P_e \) and \( P_w \) to get \( P_e \), the pressure at the east face of the cell (and similarly for the west face). The source term \( S_{\text{up}} \) can then be approximated as

\[
S_{\text{up}} = \frac{(P_e - P_w)}{\Delta x} (\Delta x \Delta y)
\]

(3)
On a uniform grid, this interpolation gives $P_w - P_e = 0.5(P_W - P_P)$, so there is a relatively weak linkage between the velocity and local pressure field (as $P_E$ is not used).

This can lead to chequerboarding: a pressure field that oscillates from node to node on the grid can still appear as uniform to the discretized momentum equation. ($P_E$ and $P_W$ can be the same value, for example, whilst $P_P$ could be significantly higher or lower).

Methods are available to overcome the above problem. However, for now we consider an alternative arrangement that results in a stronger linkage between adjacent pressure and velocity values.

A much stronger coupling between the velocity and pressure field nodal values is obtained by using a staggered grid.

In this arrangement the velocity components are stored at the centres of the faces of the pressure control volume.

The disadvantage of this is that there are separate control volumes for $U$, $V$ and $P$, so more geometrical information has to be stored.

Such overheads become particularly cumbersome in non-orthogonal and 3-D grid arrangements.

The pressure gradient term in the equation for $U_e$ is now simply

$$S_{up} = \frac{(P_P - P_E)}{\Delta x} (\Delta x \Delta y)$$

so there is a direct coupling between the velocity and the pressure values at adjacent nodes.

For now, we thus work within a staggered grid arrangement. In the CFD II course we do consider how to handle the case when all quantities are stored at the same location.

Pressure Correction Schemes

The problem in obtaining the pressure field arises because, although we have 3 equations for $U$, $V$ and $P$, the continuity equation does not explicitly contain $P$.

The momentum equations provide us with a set of discretized equations which can be solved for $U$ and $V$, if we know the pressure field. However, we cannot use the continuity equation directly to obtain $P$.

Instead, we consider how an iterative procedure can be used to adjust the pressure field in order to ensure that the resulting velocity field does satisfy continuity.

Such schemes are generally referred to as Pressure Correction schemes. At each iteration a ‘correction’ to the pressure distribution is calculated, designed to drive the local velocity field towards one that satisfies both momentum and continuity equations.

The SIMPLE Scheme

A widely-used scheme for coupling the pressure and velocity is the SIMPLE (Semi Implicit Method for Pressure Linked Equations) scheme of Patankar (1980), which is outlined in the following.

The discretized momentum equations for $U_e$ and $V_n$ are written as:

$$a^n_{e} U_e = \sum a^n_i U_i + S_{up} + S_u$$

$$a^n_{n} V_n = \sum a^n_i V_i + S_{vp} + S_v$$

Note that, because of the staggered grid, $U_e$ denotes a nodal value of the $U$ velocity, and $V_n$ a nodal value of $V$. The summation on the right hand sides represents the contributions from the surrounding nodal values, $S_{up}$ and $S_{vp}$ are the pressure-related source terms, and $S_u$ and $S_v$ any other source terms.

After dividing through by the diagonal coefficient, the discretized momentum equations can be written in the form

$$U_e = \sum \frac{a^n_i U_i}{a^n_{e}} + D_u (P_P - P_E) + S_u$$

$$V_n = \sum \frac{a^n_i V_i}{a^n_{n}} + D_v (P_P - P_N) + S_v$$

where $D_u = \Delta y / a^n_{e}$, $D_v = \Delta x / a^n_{n}$.
Starting off with some initial values for the pressure field, the above equations can be solved to obtain \( U \) and \( V \).

However, these will not, in general, satisfy the continuity equation.

Now suppose we add corrections \( U' \), \( V' \), \( P' \) to the velocities and pressure so that the corrected variables

\[
U^* = U + U' \quad V^* = V + V' \quad P^* = P + P'
\]

satisfy both the momentum and continuity equations.

By substituting these expressions into the discretized momentum and continuity equations, we now attempt to solve for the corrections \( U' \), \( V' \), and \( P' \).

Substituting these new values into the discretized momentum equations gives:

\[
(U_e + U'_e) = \frac{\alpha}{\alpha_n} (U_i + U'_i) + D_u (P_F - P_E) + D_m (P'_{p_1} - P'_{p_2}) + S_u \quad (8a)
\]

\[
(V_n + V'_n) = \frac{\alpha}{\alpha_n} (V_i + V'_i) + D_e (P_F - P_N) + D_m (P'_{p_1} - P'_{p_2}) + S_v \quad (8b)
\]

Subtracting equations (5) from these, gives

\[
U'_e = \sum \frac{\alpha}{\alpha_n} (U_i + U'_i) + D_u (P_F - P_E) \quad (9a)
\]

\[
V'_n = \sum \frac{\alpha}{\alpha_n} (V_i + V'_i) + D_e (P_F - P_N) \quad (9b)
\]

which provides relations between the corrections to the pressure and velocity that ensure the momentum equations are still satisfied.

To simplify the analysis, we now split the velocity and pressure corrections into two parts: \( U' = U'_1 + U'_2 \) etc. Then we can write

\[
U'_{e1} + U'_{e2} = \sum \frac{\alpha}{\alpha_n} (U_i + U'_i) + D_u (P_{p1} - P_{p_2}) + D_m (P'_{p1} - P'_{p_2}) + S_u \quad (10a)
\]

\[
V'_{n1} + V'_{n2} = \sum \frac{\alpha}{\alpha_n} (V_i + V'_i) + D_e (P_{p1} - P_{p_2}) + D_m (P'_{p1} - P'_{p_2}) + S_v \quad (10b)
\]

As a first approximation, we now assume that the second part of the corrections \( U'_2 \), \( V'_2 \), \( P'_2 \) may be neglected. Substituting equations (11) into the discretized continuity equation then results in

\[
\Delta y \left[ [\rho D_u]_e (P_{p1} - P_{p_2}) - [\rho D_u]_w (P_{p1} - P_{p_2}) \right] + \Delta x \left[ [\rho D_e]_w (P_{p1} - P_{p_2}) - [\rho D_e]_e (P_{p1} - P_{p_2}) \right] = -S_m \quad (14)
\]

The above equation can then simply be written in the form

\[
a_p P'_{p1} = a_e P'_{e1} + a_w P'_{w1} + a_n P'_{n1} + a_s P'_{s1} + S_u \quad (15)
\]

where

\[
a_e = \Delta y [\rho D_u]_e \quad a_w = \Delta y [\rho D_u]_w \quad a_n = \Delta x [\rho D_e]_w \quad a_s = \Delta x [\rho D_e]_e
\]

\[
a_p = a_e + a_w + a_n + a_s \quad S_u = -S_m
\]

This set of linear equations for the pressure correction \( P'_e \) can be solved, using methods considered in earlier lectures.

The corresponding corrections to the velocities, \( U'_i \) and \( V'_i \), are then obtained from equations (11), and the pressure and velocities are thus all updated.
These expressions are then substituted into the discretized continuity equation, which now becomes:

\[
(\rho w'_{x2} - \rho w'_{w2}) \Delta y + (\rho n'_{w2} - \rho n'_{x2}) \Delta x = 0 \tag{17}
\]

We thus get:

\[
\Delta y \left[ [\rho D_w]_w (P'_{x2} - P'_{w2}) - [\rho D_w]_w (P'_{w2} - P'_{x2}) \right] + \Delta x \left[ [\rho D_v]_n (P'_{x2} - P'_{n2}) - [\rho D_v]_n (P'_{n2} - P'_{x2}) \right] = \\
\Delta y \left[ \rho \sum a_p U'_{i1} \frac{a_{U_i}}{a_p} \right]_w - \Delta y \left[ \rho \sum a_p V'_{i1} \frac{a_{V_i}}{a_p} \right]_e \\
+ \Delta x \left[ \rho \sum a_p U'_{i1} \frac{a_{U_i}}{a_p} \right]_n - \Delta x \left[ \rho \sum a_p V'_{i1} \frac{a_{V_i}}{a_p} \right]_s \tag{18}
\]

This can again be written in the generic form:

\[
a_p P'_{x2} = a_w P'_{w2} + a_w P'_{n2} + a_v P'_{n2} + a_v P'_{x2} + S_u \tag{19}
\]

where the \( a_i \) coefficients are the same as those in the equation for the 1\( P_i \) correction, and the source term \( S_u \) is now given by the right hand side of equation (18) above.

In the PISO scheme, therefore, between steps 6 and 7 of the SIMPLE algorithm, the source terms \( S_u \) of equation (19) are calculated, the equation for \( P'_i \) is solved and the result used as a second correction to the pressure.

---

The PISO Scheme

One problem with the SIMPLE scheme as described is that it can be rather slow to converge.

One reason for this is the neglect of the corrections \( U'_2 \), \( V'_2 \) and \( P'_2 \).

There are a number of variants on the SIMPLE scheme, designed to improve convergence speeds.

One such variant, which is very similar in structure, and may sometimes lead to better convergence, is the PISO (Pressure Implicit solution by Split Operator method) scheme proposed by Issa (1982).

In the PISO scheme, the same decomposition of corrections to the velocity and pressure is made as in the SIMPLE scheme, and \( U'_1 \), \( V'_1 \) and \( P'_1 \) are computed as described earlier.

However, a second corrector stage is now added, in an attempt to account for the neglected \( U'_2 \), \( V'_2 \) and \( P'_2 \) in the first stage.

In this second corrector stage the second parts of equations (11) are approximated by:

\[
U'_{x2} = \sum a_p U'_{i1} \frac{a_{U_i}}{a_p} + D_u (P'_{x2} - P'_{w2}) \tag{16a}
\]

\[
V'_{n2} = \sum a_p V'_{i1} \frac{a_{V_i}}{a_p} + D_v (P'_{n2} - P'_{n2}) \tag{16b}
\]