
5.1 The Momentum Equation

Each component of momentum satisfies its own scalar-transport equation. For one cell:

$$\underbrace{\frac{\mathrm{d}}{\mathrm{d}t}(\text{mass}\times\phi)}_{\text{rate of change}} + \sum_{\text{faces}}\Big(\underbrace{C\phi}_{\text{advection}} \underbrace{-\Gamma\frac{\partial\phi}{\partial n}A}_{\text{diffusion}} \Big) = \underbrace{S}_{\text{source}} \qquad (1)$$

where C is the mass flux through a cell face. For the momentum-equation components:

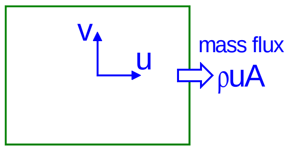
concentration, ϕ ← velocity component ($\phi = u, v, w$)
diffusivity, Γ ← viscosity, μ
source ← non-viscous forces

However, the equations for the momentum components differ from those for passive scalars (those not affecting the flow) because they are:

- non-linear;
- coupled;
- required also to be mass-consistent.

For non-linearity, note that the x-momentum flux through an x-directed face is

$$Cu = (\rho uA)u$$



The mass flux C is not constant but changes with u. The momentum equation is therefore *non-linear* and must be solved *iteratively*.

For coupling, note that the y-momentum flux through an x-directed face is

$$Cv = (\rho uA)v$$

The v equation depends on the solution of the u equation, and vice versa. Hence, the momentum equations are *coupled* and must be solved *together*.

Pressure also appears in the momentum equations. The need to determine it, together with the fact that velocity components u, v, w must satisfy mass conservation as well as momentum, further couples the equations. This distinguishes compressible and incompressible CFD.

- In *compressible* flow, continuity provides a transport equation for density ρ. Pressure is obtained by solving an energy equation to find temperature T and then using an equation of state (e.g. $p = \rho RT$).
- In *incompressible* flow, density variations are not determined by pressure. A pressure equation arises from the requirement that the solutions of the momentum equations are also mass-consistent; i.e. mass conservation actually leads to a pressure equation!

In most incompressible-flow solvers mass and momentum equations are solved sequentially and iteratively. This is called a *segregated* approach, as in the following pseudocode:

```
DO WHILE (not_converged)
   CALL SCALAR_TRANSPORT( u )
   CALL SCALAR_TRANSPORT( v )
   CALL SCALAR_TRANSPORT( w )

   CALL MASS_CONSERVATION
END DO
```

By contrast, in many compressible-flow codes the main fluid variables are assembled and solved as a vector $(\rho, \rho u, \rho v, \rho w, \rho e)$. This is called a *coupled* approach.

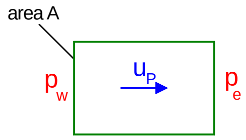
5.2 Pressure-Velocity Coupling

Question 1. How are velocity and pressure linked?

Question 2. How does a pressure equation arise?

Question 3. Should velocity and pressure be *co-located* (stored at the same locations)?

5.2.1 Link Between Pressure and Velocity



The discretised momentum equation for one cell is of the form

$$\underbrace{a_P u_P - \sum_F a_F u_F}_{\text{net momentum flux}} = \underbrace{A(p_w - p_e)}_{\text{pressure force}} + \text{other forces} \qquad (2)$$

Hence,

$$u_P = d_P(p_w - p_e) + \cdots \qquad \text{where} \qquad d_P = \frac{A}{a_P} \qquad (3)$$

<u>Answer 1</u>

(a) The force terms in the momentum equation provide a link between velocity and pressure.

(b) Velocity depends on the pressure gradient or, when discretised, on the difference between pressure values ½ cell either side. Symbolically:

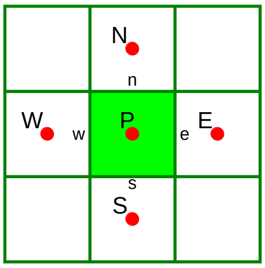
$$u = -d\Delta p + \cdots$$

where Δ indicates a centred difference ("right minus left").

Substituting for velocity in the continuity equation,

$$0 = (\rho u A)_e - (\rho u A)_w + \cdots$$
$$= (\rho A d)_e (p_P - p_E) - (\rho A d)_w (p_W - p_P) + \cdots$$
$$= -a_W p_W + a_p p_p - a_E p_E + \cdots$$

This has the same algebraic form as the scalar-transport equation.



Answer 2

The momentum equation provides a link between velocity and pressure which, when substituted into the continuity equation, gives an equation for pressure.

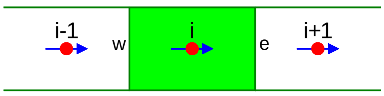
Hence, a pressure equation arises from the requirement that solutions of the momentum equation be mass-consistent.

5.2.2 Co-located Storage of Variables

Suppose pressure and velocity are *co-located* (stored at the same positions) and that advective velocities (cell-face velocities in the mass fluxes) are calculated by linear interpolation.

In the **momentum** equation the net pressure force involves

$$p_w - p_e = \frac{1}{2}(p_{i-1} + p_i) - \frac{1}{2}(p_i + p_{i+1})$$
$$= \frac{1}{2}(p_{i-1} - p_{i+1})$$

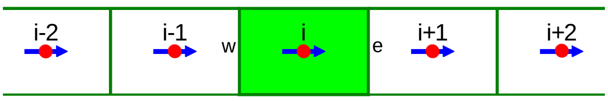


Hence, the discretised momentum equation has the form:

$$u_i = \frac{1}{2} d_i (p_{i-1} - p_{i+1}) + \cdots$$

In the **continuity** equation the net outward mass flux depends on

$$u_e - u_w = \frac{1}{2}(u_i + u_{i+1}) - \frac{1}{2}(u_{i-1} + u_i)$$
$$= \frac{1}{2}(u_{i+1} - u_{i-1})$$
$$= \frac{1}{4}[d_{i+1}(p_i - p_{i+2}) - d_{i-1}(p_{i-2} - p_i)] + \cdots$$

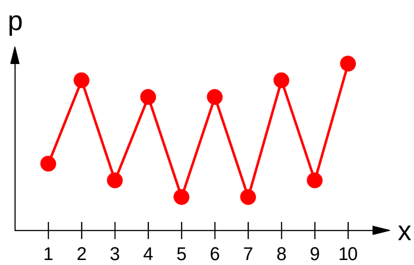


So, both mass and momentum equations only link pressures at alternate nodes.

Thus, the combination of:

- co-located u, p;
- linear interpolation for advective velocities;

leads to decoupling of odd nodal values $p_1, p_3, p_5, \ldots$ from even nodal values $p_2, p_4, p_6, \ldots$. This *odd-even decoupling* or *checkerboard* effect leads to oscillations in pressure.

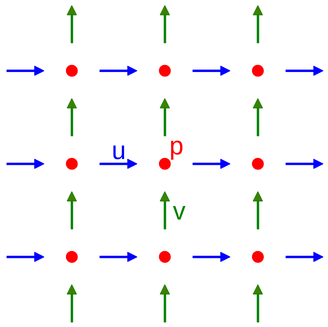


There are two common remedies:

(1) use a *staggered* grid (velocity and pressure stored at different locations); or

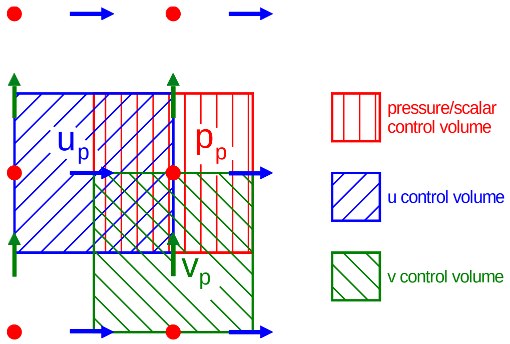
(2) use a *co-located* grid but *Rhie-Chow interpolation* for the advective velocities.

Both provide a link between adjacent pressure nodes, preventing odd-even decoupling.



5.2.3 Staggered Grid (Harlow and Welch, 1965)

In the *staggered-grid* arrangement, velocity components are stored half-way between the pressure nodes that drive them.

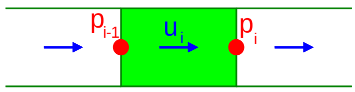


This leads to different sets of control volumes: one set for pressure (and other scalars) and others for the different velocity components.

One indexing convention gives each velocity node the same index (P or ijk) as the pressure node to which it points. Another convention refers to, e.g., the $u_{i-1/2}$ velocity.

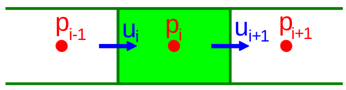
On a <u>cartesian</u> mesh …

- In the momentum equation, pressure is stored at precisely the points required to compute the pressure force.



$$u_i = d_i(p_{i-1} - p_i) + \cdots$$

- In the continuity equation velocity is stored at precisely the points required to compute mass fluxes. For net mass flux:



$$\begin{aligned} u_{i+1} - u_i + \cdots &= d_{i+1}(p_i - p_{i+1}) - d_i(p_{i-1} - p_i) + \cdots \\ &= -d_i p_{i-1} + (d_i + d_{i+1})p_i - d_{i+1}p_{i+1} \end{aligned}$$

No interpolation is required for cell-face values and there is a strong linkage between successive pressure nodes, avoiding odd-even decoupling.

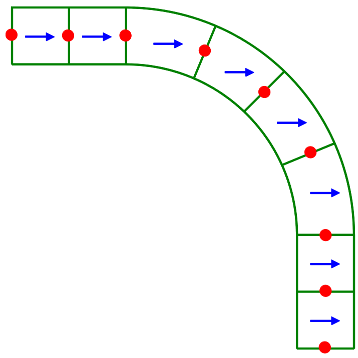
(In shallow-water codes, depth h replaces pressure p and can also be staggered from velocity.)

<u>Advantages</u>

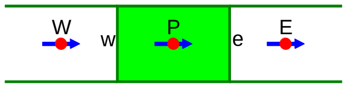
- No interpolation required; on cartesian meshes, variables are stored exactly where needed.
- No odd-even pressure decoupling.

<u>Disadvantages</u>

- Added geometrical complexity from multiple sets of nodes and control volumes.
- If the mesh is not cartesian then the velocity nodes may cease to lie between the pressure nodes that drive them (see right).
- Very difficult to implement on unstructured meshes.



5.2.4 Rhie-Chow Velocity Interpolation (Rhie and Chow, 1983)



The alternative approach uses co-located pressure and velocity but employs a different, pressure-dependent, interpolation for *advective velocities* (cell-face velocities in the mass fluxes).

The momentum equation connects velocity and pressure:

$$u_P = \frac{\sum a_F u_F}{a_P} - d_P(p_e - p_w) + \cdots \qquad (4)$$

Symbolically,

$$u = \hat{u} - d\Delta p \qquad (5)$$

$\hat{u}$ is called the *pseudovelocity*. It represents everything on the RHS except pressure. In the Rhie-Chow algorithm this *symbolic relation is applied at both nodes and faces*.

(i) Invert (5) to work out $\hat{u}$ at <u>nodes</u>:

$\hat{u} = u + d\Delta p_{\text{face}}$ with centred difference Δp from interpolated <u>face</u> values

(ii) Then linearly interpolate $\hat{u}$ and d to cell <u>faces</u>:

$u_{\text{face}} = \hat{u}_{\text{face}} - d_{\text{face}}\Delta p$ with centred difference Δp taken from adjacent <u>nodes</u>

This amounts to adding and subtracting centred pressure differences worked out at different places; e.g. on the east face, with an overbar denoting linear interpolation to that face,

$$u_e = (\overline{u + d\Delta p})_e - \overline{d}_e(p_E - p_P) \qquad (6)$$

Using this interpolative technique, mass conservation gives:

$$\begin{aligned} 0 &= (\rho A u)_e - (\rho A u)_w + \cdots \\ &= (\rho A \hat{u})_e - (\rho A \hat{u})_w \quad - \quad (\rho A d)_e(p_E - p_P) + (\rho A d)_w(p_P - p_W) + \cdots \qquad (7) \\ &= (\rho A \hat{u})_e - (\rho A \hat{u})_w \quad - \quad a_W p_W + a_P p_P - a_E p_E + \cdots \end{aligned}$$

Notes.

- The central pressure value does not cancel; there is no odd-even decoupling.
- This <u>is</u> a pressure equation. Moreover, it has the same algebraic form as a scalar-transport equation:

$$a_P p_P - \sum a_F p_F = b_P, \qquad \text{where } a_F \geq 0\ , \quad a_P = \sum a_F$$

- In practice it is common to solve iteratively for pressure *corrections* rather than pressure itself (Section 5.3). Equation (7) then becomes

$$0 = (\rho A u^*)_e - (\rho A u^*)_w \quad - \quad a_W p'_W + a_P p'_P - a_E p'_E + \cdots$$

where $*$ denotes "current value of". This can be rearranged as:

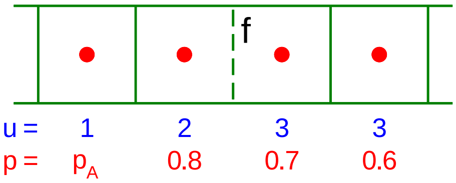
$$-a_W p'_W + a_P p'_P - a_E p'_E + \cdots = \ -\text{current mass outflow} \qquad (8)$$

Example

For the uniform cartesian mesh shown below the momentum equation gives a velocity/pressure relationship

$$u = -4\Delta p + \cdots$$

for each cell, where Δ denotes a centred difference.



For the u and p values given, calculate the advective velocity on the cell face marked *f*:

(a) by linear interpolation;

(b) by Rhie-Chow interpolation, if $p_A = 0.6$ (local pressure maximum to the left of face);

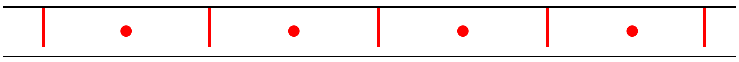
(c) by Rhie-Chow interpolation, if $p_A = 0.9$ (constant pressure gradient).

<u>Answer</u>

(a) By linear interpolation,

$$u_f = \frac{1}{2}(2 + 3) \quad = 2.5$$

(b) The working for Rhie-Chow may be set out as follows.



u	1		2		3		3	
p	0.6		0.8		0.7		0.6	
p_{face}		**0.7**		**0.75**		**0.65**		by interpolation of p
$\hat{u}$			**2.2**		**2.6**			$\hat{u} = u + 4(p_e - p_w)$
$\hat{u}_{\text{face}}$				**2.4**				by interpolation of $\hat{u}$
u_{face}				**2.8**				$u_{\text{face}} = \hat{u}_{\text{face}} - 4(p_E - p_W)$

Answer: $u_f = 2.8$

This is higher than that obtained from linear interpolation, showing the velocity field trying to alleviate the local pressure peak just to the left of the face.

(c) If $p_A = 0.9$ then Rhie-Chow interpolation produces $u_f = 2.5$, as for linear interpolation.

Analysis of Rhie-Chow Interpolation

Parts (a) and (c) of the example above show that Rhie-Chow interpolation gives the same result as linear interpolation if there is a constant pressure gradient, whereas if (as in part(b)) there is a local pressure peak to one side of the face then the advective velocity increases to try to reduce it.

For general values of u_i and p_i (and a uniform mesh and a constant coefficient d), Rhie-Chow interpolation gives (*exercise*):

$$u_{\text{face}} = \frac{1}{2}(u_2 + u_3) + \frac{d}{4}(-p_1 + 3p_2 - 3p_3 + p_4)$$

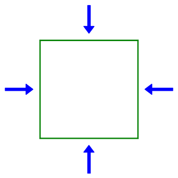
Thus, Rhie-Chow interpolation adds a "pressure-smoothing" term to the part obtained simply by interpolation.

Pressure-velocity coupling is the dominant feature of the Navier-Stokes equations. Staggered grids are an effective way of handling it on cartesian meshes. However, for non-cartesian meshes, co-located storage is the norm; the Rhie-Chow algorithm is employed in most general-purpose CFD codes.

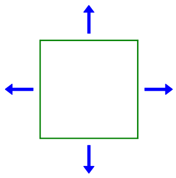
A generalised form of the Rhie-Chow algorithm (based on local pressure gradients) is used for unstructured meshes.

5.3 Pressure-Correction Methods

Consider how changing pressure could be used to enforce mass conservation.



Net mass flux in.
Increase cell pressure to drive mass out.



Net mass flux out.
Decrease cell pressure to suck mass in.

<u>What are *pressure-correction* methods?</u>

- Iterative numerical schemes for pressure-linked equations.
- Seek to produce velocity and pressure satisfying both mass and momentum equations.
- Consist of alternating updates of velocity and pressure:
  - solve the momentum equation for velocity with the current pressure;
  - use velocity-pressure link to rephrase continuity as a pressure-correction equation;
  - solve for pressure corrections to "nudge" velocity towards mass conservation.
- There are two common schemes: SIMPLE and PISO.

<u>Velocity and Pressure Corrections</u>

The momentum equation connects velocity and pressure:

$$u = d(p_{-1/2} - p_{+1/2}) + \cdots$$

where –½ and +½ indicate the relative locations as multiples of the grid spacing.

One must correct velocity to satisfy continuity:

$$u \to u^* + u'$$

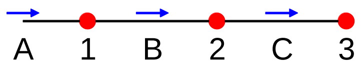
but simultaneously correct pressure so as to retain a solution of the momentum equation:

$$u' = d(p'_{-1/2} - p'_{+1/2}) + \cdots$$

The velocity-correction formula is, therefore,

$$\boxed{u \quad \to \quad u^* + d(p'_{-1/2} - p'_{+1/2}) + \cdots \qquad (9)}$$

Classroom Example 1 (Patankar, 1980)



In the steady, one-dimensional, constant-density situation shown, the pressure p is stored at locations 1, 2 and 3, whilst velocity u is stored at the staggered locations A, B and C. The velocity-correction formula is

$$u = u^* + u' \ , \quad \text{where} \quad u' = d(p'_{i-1} - p'_i)$$

where pressure nodes $i-1$ and i lie on either side of the location for u. The value of d is 2 everywhere. The inflow boundary condition is $u_A = 10$. If, at a given stage in the iteration process, the momentum equations give $u_B^* = 8$ and $u_C^* = 11$, calculate the values of p'_1, p'_2, p'_3 and the resulting corrected velocities.

Classroom Example 2

In the 2-d staggered-grid arrangement shown below, u and v (the x and y components of velocity), are stored at nodes indicated by arrows, whilst pressure p is stored at the intermediate nodes A – D. The grid spacing is uniform and the same in both directions. The velocity is fixed on the boundaries as shown. The velocity components at the interior nodes (u_B, u_D, v_C and v_D) are to be found.

At an intermediate stage of calculation the internal velocity values are found to be

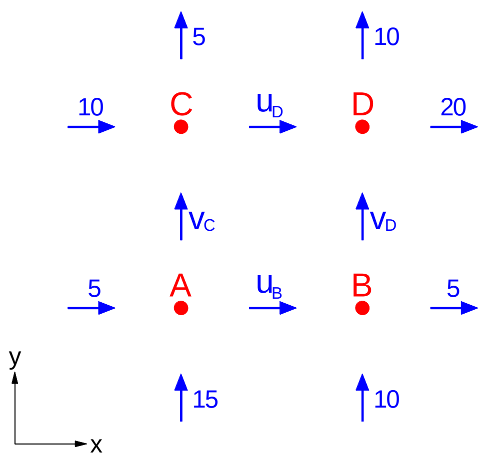
$$u_B = 11, \quad u_D = 14, \quad v_C = 8, \quad v_D = 5$$

whilst correction formulae derived from the momentum equation are

$$u' = 2(p'_w - p'_e) \ , \qquad\qquad v' = 3(p'_s - p'_n)$$

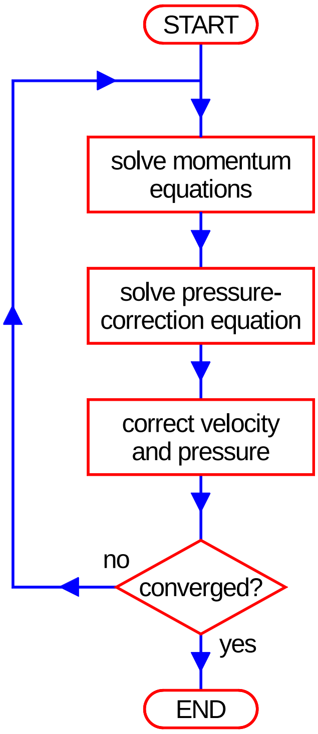
with geographical (*w*,*e*,*s*,*n*) notation indicating the relative location of pressure nodes.

(a) Show that applying mass conservation to control volumes centred on pressure nodes leads to simultaneous equations for the pressure corrections. Solve for the pressure corrections and use them to generate a mass-consistent flow field.



(b) Explain why, in practice, it is necessary to solve for the pressure correction and not just the velocity corrections in order to satisfy mass conservation.

5.3.1 SIMPLE: Semi-Implicit Method for Pressure-Linked Equations
(Patankar and Spalding, 1972)



Stage 1. Solve the momentum equation with current pressure.

$$a_P u_p - \sum a_F u_F = \underbrace{A(p_w^* - p_e^*)}_{\text{pressure force}} + \underbrace{b_P}_{\text{other forces}}$$

The resulting velocity generally won't be mass-consistent.

Stage 2. Formulate the pressure-correction equation.

(i) Relate changes in u to changes in p:

$$u'_P = \frac{\sum a_F u'_F}{a_P} + d_P(p'_w - p'_e), \qquad d_P = \frac{A}{a_P}$$

(ii) Make the SIMPLE approximation: neglect $\sum a_F u'_F$.

$$u'_P \approx d_P(p'_w - p'_e)$$

(Legitimate, since corrections vanish in the final solution.)

(iii) Apply mass conservation to control volumes centred on the pressure nodes. The net mass flux results from current ($\mathbf{u}^*$) plus correction ($\mathbf{u}'$) velocity fields:

$$\text{net mass flow out} = \sum_{\text{faces}} \rho u_n^* A + \sum_{\text{faces}} \rho u'_n A = 0$$

Hence,

$$(\rho u' A)_e - (\rho u' A)_w + \cdots = -\dot{m}^* \qquad \text{(minus the current net mass flux)}$$

or, writing in terms of the pressure correction (staggered or non-staggered mesh):

$$(\rho A d)_e (p'_P - p'_E) - (\rho A d)_w (p'_W - p'_P) + \cdots = -\dot{m}^*$$

This results in a pressure-correction equation of the form:

$$a_P p'_P - \textstyle\sum_F a_F p'_F = -\dot{m}^* \qquad (10)$$

Stage 3. Solve the pressure-correction equation

The discretised pressure-correction equation (10) has the same form as the discretised scalar-transport equation, and hence the same solver may be used.

Stage 4. Correct pressure and velocity:

$$\begin{aligned} p_P &\rightarrow p_P^* + p'_P \\ u_P &\rightarrow u_P^* + d_P(p'_w - p'_e) \end{aligned} \qquad (11)$$

Iterate.

Repeat stages 1 to 4 until mass and momentum equations are simultaneously satisfied.

Notes.

- The source term for the pressure-correction equation is minus the current net mass flux ($-\dot{m}^*$), as expected. If at any stage there is net mass flow into a control volume then the pressure there must rise in order to "push" mass back out.

- In practice, substantial under-relaxation of the pressure update is needed to prevent divergence. In the correction step the pressure (but not the velocity) update is relaxed:

  $$p \rightarrow p^* + \alpha_p p'$$

  Typical values of α_p are $0.1 - 0.3$. Velocity is under-relaxed in the momentum transport equations, but the under-relaxation is generally less severe: $\alpha_u \approx 0.6 - 0.8$.

- As they involve the velocities being computed, matrix elements change at each iteration. There is little to be gained by solving matrix equations exactly at each stage, but only doing enough iterations of the matrix solver to reduce the residuals by a sufficient amount. Alternative strategies at each SIMPLE iteration are:
  (1) m iterations of each u, v, w equation, followed by n iterations of the p' equation (typically $m = 1, n = 4$); or
  (2) do enough iterations of each equation to reduce the residual error to a small fraction of the original (say 10%).

5.3.2 Variants of SIMPLE

The SIMPLE scheme can be inefficient and requires considerable pressure under-relaxation. This is because the corrected fields are good for updating velocity (since a mass-consistent flow field is produced) but not pressure (because of the inaccuracy of the approximation connecting velocity and pressure corrections). To remedy this, a number of variants of SIMPLE have been produced.

<u>SIMPLER</u> (Patankar, 1980)
This variant acknowledges that the correction equation is good for updating velocity but not pressure, and precedes the momentum and pressure-correction equations with the equation for the pressure itself (equation (7)).

<u>SIMPLEC</u> (Van Doormaal and Raithby, 1984)
This scheme seeks a more accurate relationship between velocity and pressure changes. From the momentum equations, the velocity and pressure equations are related by

$$u'_P = \underbrace{\frac{1}{a_P}\sum a_F u'_F}_{(*)} + d_P(p'_w - p'_e) \tag{12}$$

The SIMPLE approximation is to neglect the term (*). However, this is actually of comparable size to the LHS. In the SIMPLEC scheme, (12) is rewritten by subtracting $(1/a_P)\sum a_F\, u'_P$ from both sides:

$$(1 - \frac{1}{a_P}\sum a_F)u'_P = \underbrace{\frac{1}{a_P}\sum a_F(u'_F - u'_P)}_{(**)} + d_P(p'_w - p'_e)$$

Assuming that $|u'_F - u'_P| \ll |u'_P|$, it is more accurate to neglect the term (**), thus producing an alternative formula connecting velocity and pressure changes:

$$u'_P \approx \frac{d_P}{1 - \sum a_F/a_P}(p'_w - p'_e) \tag{13}$$

Compare:

$$\begin{aligned} u'_P &\approx d_P(p'_w - p'_e) && \text{(SIMPLE)} \\ u'_P &\approx \frac{d_P}{1 - \sum a_F/a_P}(p'_w - p'_e) && \text{(SIMPLEC)} \end{aligned} \tag{14}$$

Although conceptually appealing, there is a difficulty. The "sum-of-the-neighbouring-coefficients" constraint on a_P means that, in steady-state calculations, $\sum a_F/a_P = 1$, and hence the denominator of (13) vanishes. This problem doesn't arise in time-varying calculations, where a time-dependent part is added to a_P, removing the singularity.

<u>SIMPLEX</u>

This scheme assumes that velocity and pressure corrections are linked by some general relationship

$$u'_P \approx \delta_P(p'_w - p'_e) \tag{15}$$

This includes both SIMPLE and SIMPLEC as special cases (see equation (14)). However, the SIMPLEX scheme attempts to improve on this by actually solving equations for the δ_P. These equations are derived from (12) but we shall not go into the details here.

This author's experience is that SIMPLER and SIMPLEX offer substantial performance improvements over SIMPLE on staggered grids, but that the "advanced" schemes are difficult (impossible?) to formulate on co-located grids and offer little advantage there.

5.3.3 PISO

PISO – <u>P</u>ressure <u>I</u>mplicit with <u>S</u>plitting of <u>O</u>perators (Issa, 1986).

This was originally proposed as a time-dependent, non-iterative pressure-correction method. Each timestep ($t^{\text{old}} \to t^{\text{new}}$) consists of a sequence of three stages:

(i) Solution of the time-dependent momentum equation with the t^{old} pressure.

(ii) A pressure-correction equation and pressure/velocity update à la SIMPLE to produce a mass-consistent flow field.

(iii) A second corrector step to produce a second mass-consistent flow field but with time-advanced pressure.

Apart from time-dependence, steps (i) and (ii) are essentially the same as SIMPLE. However, step (iii) is designed to eliminate the need for *outer* iteration at each time step as would be the case with SIMPLE. This sounds great … however, the pressure-correction equations (ii) and (iii) must be solved to a much finer tolerance on each pass, so requiring many more *inner* iterations of the matrix equations.

Evidence suggests that PISO can be more efficient in some time-dependent calculations, but SIMPLE and its variants are better in direct iteration to steady state.

Summary

- Each component of momentum satisfies its own scalar-transport equation:
  concentration, ϕ ← velocity component (u, v, w)
  diffusivity, Γ ← viscosity, μ
  source, S ← non-viscous forces

- However, the momentum equations are:
  non-linear;
  coupled;
  required also to be mass-consistent.
  and, as a consequence, have to be solved:
  iteratively;
  together;
  in conjunction with the mass equation.

- For incompressible flow, the requirement that solutions of the momentum equation be mass-consistent generates a pressure equation.

- Pressure-gradient source terms lead to odd-even decoupling when all variables are co-located (stored at the same nodes). This may be remedied by using either:
  – a staggered velocity grid;
  – a non-staggered grid, but Rhie-Chow interpolation for advective velocities.

- Pressure-correction methods make small corrections to pressure in order to "nudge" the velocity field towards mass conservation whilst still preserving a solution of the momentum equation.

- Widely-used pressure-correction algorithms are SIMPLE (and its variants) and PISO. The first is an iterative scheme; the second is a non-iterative, time-dependent scheme.

References

Harlow, F.H. and Welch, J.E., 1965, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, Physics of Fluids, 8, 2182-2189.

Issa, R.I., 1986, Solution of the implicitly discretised fluid flow equations by operator splitting, J. Comput. Phys., 62, 40-65.

Patankar, S.V., 1980, *Numerical Heat Transfer and Fluid Flow*, McGraw-Hill.

Patankar, S.V., 1988, Elliptic systems: finite difference method I, Chapter 6 in *Handbook of Numerical Heat Transfer*, Minkowycz, W.J., Sparrow, E.M., Schneider, G.E. and Pletcher, R.H. (eds.), Wiley.

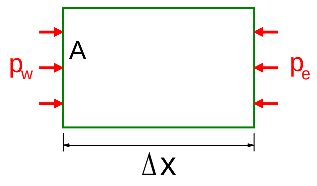
Raithby, G.D. and Schneider, G.E., 1988, Elliptic systems: finite difference method II, Chapter 7 in *Handbook of Numerical Heat Transfer*, Minkowycz, W.J., Sparrow, E.M., Schneider, G.E. and Pletcher, R.H. (eds.), Wiley.

Rhie, C.M. and Chow, W.L., 1983, A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation, AIAA Journal, 21, 1525-1532.

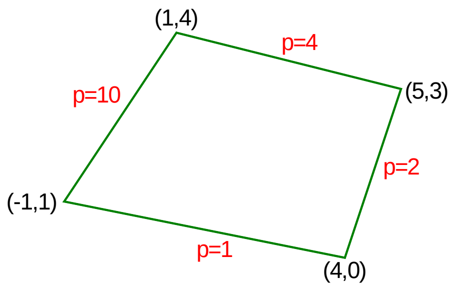
Van Doormaal, J.P. and Raithby, G.D., 1984, Enhancements of the SIMPLE method for predicting incompressible fluid flows, Numerical Heat Transfer, 7, 147-163.

Examples

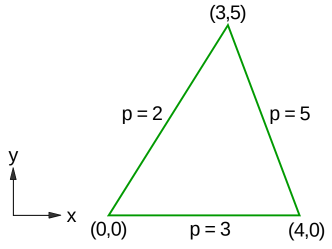
Q1.
For the rectangular control volume with surface pressures shown, what is:
(a) the net force in the x direction?
(b) the net force in the x direction, per unit volume?
(c) the average pressure gradient in the x direction?



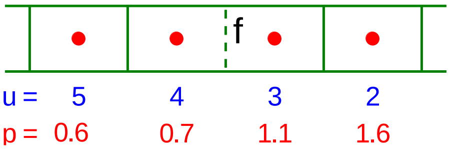
Q2.
The figure right shows a quadrilateral cell in a 2-d mesh, together with the coordinates of its vertices and the average pressures on the cell faces. Calculate the x and y components of the net pressure force on the cell (per unit depth).



Q3.
The figure shows a triangular cell in a 2-d mesh for an inviscid CFD calculation. The vertex coordinates and the average pressures on the cell edges are shown in the figure. The density ρ = 1.0 everywhere. Find the x and y components of:
(i) the net pressure force on the cell;
(ii) the fluid acceleration.



Q4.
The figure shows part of a cartesian mesh with the velocity u and pressure p at the centre of 4 control volumes. If the momentum equation leads to a pressure-velocity linkage of the form



$$u = -3\Delta p + \cdots,$$

(where Δ represents a centred difference) use the Rhie-Chow procedure to find the advective velocity on the cell face marked f.

Q5. (Exam 2016)

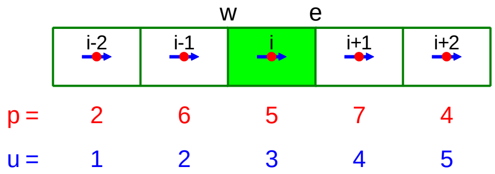
(a) By considering an infinitesimal cuboid cell with edges aligned with the coordinate axes, show that the pressure force per unit volume in the x direction is $-\partial p/\partial x$.

(b) For what purpose is the Rhie-Chow algorithm used in pressure-based finite-volume simulations of incompressible fluid flow?

Part of a uniform structured mesh is shown in the figure below, along with the cell-centred values of the x-velocity component u and the pressure p in consistent units. From the discretised momentum equation, velocities and pressures are found to be linked by

$$u = -\frac{1}{2}\Delta p + \cdots$$

where Δ here denotes a centred difference.



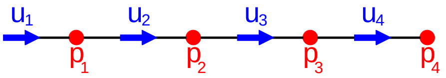
(c) Find the velocity on the 'w' and 'e' faces of cell i (positions $i - ½$ and $i + ½$) using:
  (i) linear interpolation;
  (ii) Rhie-Chow interpolation.

Comment briefly on your results and the effect of the pressure field on advective velocities.

(d) Assuming flow only in the x direction, and using the face velocities found in part (c)(ii) as the current velocity values, set up – but do not solve – the pressure-correction equation for cell i.

Q6.

The figure defines the relative position of velocity (→) and pressure (•) nodes in a 1-d, *staggered-grid* arrangement.



Velocity $u_1 = 4$ is fixed as a boundary condition. After solving the momentum equation the velocities at the other nodes are found to be

$$u_2 = 3, \quad u_3 = 5, \quad u_4 = 6.$$

The relationship between velocity and pressure is found (from the discretised momentum equation) to be of the form

$$u = -4\Delta p + \cdots$$

at each node, where Δ denotes a space-centred difference.

Apply mass conservation to cells centred on scalar nodes, calculate the pressure corrections necessary to enforce continuity, and confirm that a mass-consistent velocity field is obtained.

Q7. (Exam 2020)

(a) Outline (without mathematical details) the main steps of the SIMPLE pressure-correction algorithm.

(b) What particular problem can arise from storing pressure and velocity at the same nodal points in a finite-volume CFD calculation? State two widely-used remedies for this.

In the 2-d staggered-grid arrangement shown below, u and v (the x and y components of velocity) are stored at nodes indicated by arrows, whilst pressure p is stored at intermediate nodes A, B, C, D. Grid spacing is uniform and the same in both directions. Velocity is fixed at inflow. Upper and lower boundaries are impermeable.

At an intermediate stage of calculation internal velocity components are

$$u_C = 4, \quad u_D = -4, \quad v_A = -2, \quad v_C = 1$$

whilst the outflow velocities are

$$u_E = 6, \quad u_F = 6,$$

Velocity-pressure correction formulae are

$$u' = 2(p'_w - p'_e)\,, \qquad\qquad v' = 3(p'_s - p'_n)$$

with geographical (w,e,s,n) notation indicating the relative location of pressure nodes.

(c) Apply a uniform scale factor to the outflow velocities to enforce global mass conservation, stating the scale factor and outflow velocities after its application.

(d) Show that applying mass conservation to control volumes centred on pressure nodes leads to simultaneous equations for the pressure corrections. Solve for the pressure corrections and use them to generate a mass-consistent flow field.