Other staggering methods have been suggested, for example, the partially staggered ALE (Arbitrary Lagrangian-Eulerian) method (Hirt et al., 1974), in which both velocity components are stored at the corners of the pressure CVs, see Fig. 7.2. This variant has some advantages when the grid is non-orthogonal, an important one being that the pressure at the boundary need not be specified. However, it also has drawbacks, notably the possibility of producing oscillatory pressure or velocity fields.

Other arrangements have not gained wide popularity and will not be further discussed here.

7.3 Calculation of the Pressure

Solution of the Navier-Stokes equations is complicated by the lack of an independent equation for the pressure, whose gradient contributes to each of the three momentum equations. Furthermore, the continuity equation does not have a dominant variable in incompressible flows. Mass conservation is a kinematic constraint on the velocity field rather than a dynamic equation. One way out of this difficulty is to construct the pressure field so as to guarantee satisfaction of the continuity equation. This may seem a bit strange at first, but we shall show below that it is possible. Note that the absolute pressure is of no significance in an incompressible flow; only the gradient of the pressure (pressure difference) affects the flow.

In compressible flows the continuity equation can be used to determine the density and the pressure is calculated from an equation of state. This approach is not appropriate for incompressible or low Mach number flows.

Within this section we present the basic philosophy behind some of the most popular methods of pressure-velocity coupling. Section 7.5 presents a full set of discretized equations which form the basis for writing a computer code.

7.3.1 The Pressure Equation and its Solution

The momentum equations clearly determine the respective velocity components so their roles are clearly defined. This leaves the continuity equation, which does not contain the pressure, to determine the pressure. How can this be done? The most common method is based on combining the two equations.

The form of the continuity equation suggests that we take the divergence of the momentum equation (1.15). The continuity equation can be used to simplify the resulting equation, leaving a Poisson equation for the pressure:

$$\operatorname{div}(\operatorname{grad} p) = -\operatorname{div}\left[\operatorname{div}(\rho \mathbf{v}\mathbf{v} - \mathsf{S}) - \rho \mathbf{b} + \frac{\partial(\rho \mathbf{v})}{\partial t}\right] . \tag{7.14}$$

In Cartesian coordinates this equation reads:

$$\frac{\partial}{\partial x_i}\left(\frac{\partial p}{\partial x_i}\right) = -\frac{\partial}{\partial x_i}\left[\frac{\partial}{\partial x_j}\left(\rho u_i u_j - \tau_{ij}\right)\right] + \frac{\partial(\rho b_i)}{\partial x_i} + \frac{\partial^2 \rho}{\partial t^2} \,. \tag{7.15}$$

For the case of constant density and viscosity, this equation simplifies further; the viscous and unsteady terms disappear by virtue of the continuity equation leaving:

$$\frac{\partial}{\partial x_i}\left(\frac{\partial p}{\partial x_i}\right) = -\frac{\partial}{\partial x_i}\left[\frac{\partial(\rho u_i u_j)}{\partial x_j}\right] \,. \tag{7.16}$$

The pressure equation can be solved by one of the numerical methods for elliptic equations described in Chaps. 3 and 4. It is important to note that the right hand side of the pressure equation is a sum of derivatives of terms in the momentum equations; these must be approximated in a manner consistent with their treatment in the equations they are derived from.

It is also important to note that the Laplacian operator in the pressure equation is the product of the divergence operator originating from the continuity equation and the gradient operator that comes from the momentum equations. In a numerical approximation, it is essential that the consistency of these operators be maintained i.e. the approximation of the Poisson equations must be defined as the product of the divergence and gradient approximations used in the basic equations. Violation of this constraint leads to lack of satisfaction of the continuity equation. To emphasize the importance of this issue, the two derivatives of the pressure in the above equations were separated: the outer derivative stems from the continuity equation while the inner derivative arises from the momentum equations. The outer and inner derivatives may be discretized using different schemes – they have to be those used in the momentum and continuity equations.

A pressure equation of this kind is used to calculate the pressure in both explicit and implicit solution methods. To maintain consistency among the approximations used, it is best to derive the equation for the pressure from the discretized momentum and continuity equations rather than by approximating the Poisson equation. The pressure equation can also be used to calculate the pressure from a velocity field obtained by solving vorticity/streamfunction equations, see Sect. 7.4.2.

7.3.2 A Simple Explicit Time Advance Scheme

Before considering commonly used methods for solving the steady state Navier-Stokes equations, let us look at a method for the unsteady equations that illustrates how the numerical Poisson equation for the pressure is constructed and the role it plays in enforcing continuity. The choice of the approximations to the spatial derivatives is not important here so the semi-discretized (discrete in space but not time) momentum equations are written symbolically as:

$$\frac{\partial(\rho u_i)}{\partial t} = -\frac{\delta(\rho u_i u_j)}{\delta x_j} - \frac{\delta p}{\delta x_i} + \frac{\delta \tau_{ij}}{\delta x_j} = H_i - \frac{\delta p}{\delta x_i}\,, \tag{7.17}$$

where $\delta/\delta x$ represents a discretized spatial derivative (which could represent a different approximation in each term) and H_i is shorthand notation for the advective and viscous terms whose treatment is of no importance here.

For simplicity, assume that we wish to solve Eq. (7.17) with the explicit Euler method for time advancement. We then have:

$$(\rho u_i)^{n+1} - (\rho u_i)^n = \Delta t \left(H_i^n - \frac{\delta p^n}{\delta x_i} \right) . \tag{7.18}$$

To apply this method, the velocity at time step n is used to compute H_i^n and, if the pressure is available, $\delta p^n/\delta x_i$ may also be computed. This gives an estimate of ρu_i at the new time step $n+1$. In general, this velocity field does not satisfy the continuity equation:

$$\frac{\delta(\rho u_i)^{n+1}}{\delta x_i} = 0 . \tag{7.19}$$

We have stated an interest in incompressible flows, but these include flows with variable density; this is emphasized by including the density. To see how continuity may be enforced, let us take the numerical divergence (using the numerical operators used to approximate the continuity equation) of Eq. (7.18). The result is:

$$\frac{\delta(\rho u_i)^{n+1}}{\delta x_i} - \frac{\delta(\rho u_i)^n}{\delta x_i} = \Delta t \left[\frac{\delta}{\delta x_i} \left(H_i^n - \frac{\delta p^n}{\delta x_i} \right) \right] . \tag{7.20}$$

The first term is the divergence of the new velocity field, which we want to be zero. The second term is zero if continuity was enforced at time step n; we shall assume that this is the case but, if it is not, this term should be left in the equation. Retaining this term is necessary when an iterative method is used to solve the Poisson equation for the pressure and the iterative process is not converged completely. Similarly, the divergence of the viscous component of H_i should be zero for constant ρ, but a non-zero value is easily accounted for. Taking all this into account, the result is the discrete Poisson equation for the pressure p^n:

$$\frac{\delta}{\delta x_i} \left(\frac{\delta p^n}{\delta x_i} \right) = \frac{\delta H_i^n}{\delta x_i} . \tag{7.21}$$

Note that the operator $\delta/\delta x_i$ outside the parentheses is the divergence operator inherited from the continuity equation, while $\delta p/\delta x_i$ is the pressure gradient from the momentum equations. If the pressure p^n satisfies this discrete Poisson equation, the velocity field at time step $n+1$ will be divergence free (in terms of the discrete divergence operator). Note that the time step to which this pressure belongs is arbitrary. If the pressure gradient term had been treated implicitly, we would have p^{n+1} in place of p^n but everything else would remain unchanged.

This provides the following algorithm for time-advancing the Navier-Stokes equations:

- Start with a velocity field u_i^n at time t_n which is assumed divergence free. (As noted, if it is not divergence free this can be corrected.)
- Compute the combination, H_i^n, of the advective and viscous terms and its divergence (both need to be retained for later use).
- Solve the Poisson equation for the pressure p^n.
- Compute the velocity field at the new time step. It will be divergence free.
- The stage is now set for the next time step.

Methods similar to this are commonly used to solve the Navier-Stokes equations when an accurate time history of the flow is required. The principal differences in practice are that time advancement methods more accurate than the first order Euler method are usually used and that some of the terms may be treated implicitly. Some of these methods will be described later.

We have shown how solving the Poisson equation for the pressure can assure that the velocity field satisfies the continuity equation i.e. that it is divergence free. This idea runs through many of the methods used to solve both the steady and unsteady Navier-Stokes equations. We shall now study some of the more commonly used methods for solving the steady Navier-Stokes equations.

7.3.3 A Simple Implicit Time Advance Method

To see what additional difficulties arise when an implicit method is used to solve the Navier-Stokes equations, let us construct such a method. Since we are interested in illuminating certain issues, let us use a scheme based on the the simplest implicit method, the backward or implicit Euler method. If we apply this method to Eq. (7.17), we have:

$$(\rho u_i)^{n+1} - (\rho u_i)^n = \Delta t \left(-\frac{\delta(\rho u_i u_j)^{n+1}}{\delta x_j} - \frac{\delta p^{n+1}}{\delta x_i} + \frac{\delta \tau_{ij}^{n+1}}{\delta x_j} \right) . \quad (7.22)$$

We see immediately that there are difficulties that were not present in the explicit method described in the preceding section. Let us consider these one at a time.

First, there is a problem with the pressure. The divergence of the velocity field at the new time step must be zero. This can be accomplished in much the same way as in the explicit method. We take the divergence of Eq. (7.22), assume that the velocity field at time step n is divergence free (this can be corrected for if necessary) and demand that the divergence at the new time step $n+1$ also be zero. This leads to the Poisson equation for the pressure:

$$\frac{\delta}{\delta x_i}\left(\frac{\delta p^{n+1}}{\delta x_i}\right) = \frac{\delta}{\delta x_i}\left(\frac{-\delta(\rho u_i u_j)^{n+1}}{\delta x_j}\right) . \tag{7.23}$$

The problem is that the term on the right hand side cannot be computed until the computation of the velocity field at time $n+1$ is completed and vice versa. As a result, the Poisson equation and the momentum equations have to be solved simultaneously. That can only be done with some type of iterative procedure.

Next, even if the pressure were known, Eqs. (7.22) are a large system of non-linear equations which must be solved for the velocity field. The structure of this system of equations is essentially the same as the structure of the matrix of the finite-differenced Laplace equation so solving them is far from a trivial matter. If one wishes to solve them accurately, the best procedure is to adopt the converged results from the preceding time step as the initial guess for the new velocity field and then converge to the solution at the new time step using the Newton–Raphson iteration method or a secant method designed for systems.

An alternative way of dealing with the non-linearity is constructed by linearizing the equations about the result at the preceding time step. If we write:

$$u_i^{n+1} = u_i^n + \Delta u_i , \tag{7.24}$$

then the non-linear term in Eqs. (7.22) can be expressed as:

$$u_i^{n+1} u_j^{n+1} = u_i^n u_j^n + u_i^n \Delta u_j + u_j^n \Delta u_i + \Delta u_i \Delta u_j . \tag{7.25}$$

We expect that, at least for small Δt, $\Delta u_i \sim \Delta t\, \partial u_i/\partial t$, so the last term in this equation is second order in Δt and is smaller in magnitude than the error made in the time discretization. It can therefore be neglected. If we use a second order method in time, such as the Crank-Nicolson scheme, this term would be of the same order as the spatial discretization error and we would still be justified in neglecting it.

We can then write Eq. (7.22) as:

$$\rho \Delta u_i = \Delta t \left(-\frac{\delta(\rho u_i u_j)^n}{\delta x_j} - \frac{\delta(\rho u_i^n \Delta u_j)}{\delta x_j} - \frac{\delta(\rho \Delta u_i u_j^n)}{\delta x_j} - \frac{\delta p^n}{\delta x_i} - \frac{\delta \Delta p}{\delta x_i} + \frac{\delta \tau_{ij}^n}{\delta x_j} + \frac{\delta \Delta \tau_{ij}}{\delta x_j} \right) . \tag{7.26}$$

This method takes advantage of the fact that the non-linearity is quadratic and removes most of the difficulty arising from it. However, we still need to solve a large system of linear equations with the structure discussed above. Direct solution of such a system is too expensive to consider so the solution needs to be found iteratively. An interesting possibility is to use the alternating direction implicit (ADI) method to split the equations into a series of one dimensional problems, each of which is block tridiagonal. The solution at the new time step can then be found with sufficient accuracy with just one iteration (one set of block tridiagonal solutions in each direction).

So a reasonable strategy is to use the local linearization based on Eq. (7.24) and update the equations by the ADI method using the old pressure gradient. We can then correct the velocity field using the following scheme.

- Call the velocity field computed by updating the momentum equations with the old pressure gradient u_i^*. It does not satisfy the continuity equation.
- Solve a Poisson equation for the pressure correction:

$$\frac{\delta}{\delta x_i}\left(\frac{\delta \Delta p}{\delta x_i}\right) = \frac{1}{\Delta t}\,\frac{\delta(\rho u_i^*)}{\delta x_i}\,. \tag{7.27}$$

- Update the velocity:

$$u_i^{n+1} = u_i^* - \frac{\Delta t}{\rho}\,\frac{\delta \Delta p}{\delta x_i}\,, \tag{7.28}$$

  which does satisfy continuity.

With these tricks, the method suggested here is about twice as expensive as the explicit method per time step.

The method described above is designed to produce an accurate solution of an unsteady problem. In problems of that kind, the required accuracy in time usually sets the time step, which will be rather small. Because they allow large time steps to be used without instability, implicit methods are often used to solve steady state problems. The idea is to compute in time until a steady solution is obtained. In this type of calculation, the error made in linearizing the problem is no longer negligible and the type of method described here may not be the best choice. Methods designed for solving steady state problems are given in the next section. They introduce other means of getting around the problems we encountered here.

7.3.4 Implicit Pressure-Correction Methods

As noted in Chap. 6, many methods for steady problems can be regarded as solving an unsteady problem until a steady state is reached. The principal difference is that, when solving an unsteady problem, the time step is chosen so that an accurate history is obtained while, when a steady solution is sought, large time steps are used to try to reach the steady state quickly. Implicit methods are preferred for steady and slow-transient flows, because they have less stringent time step restrictions than explicit schemes (they may not have any).

Many solution methods for steady incompressible flows are of the latter type; some of the most popular ones can be regarded as variations on the method of the preceding section. They use a pressure (or pressure-correction) equation to enforce mass conservation at each time step or, in the language preferred for steady solvers, each outer iteration. We now look at some of these methods.

If an implicit method is used to advance the momentum equations in time, the discretized equations for the velocities at the new time step are non-linear. If the pressure gradient term is not included in the source term, these may be written:

$$A_{\mathrm{P}}^{u_i} u_{i,\mathrm{P}}^{n+1} + \sum_l A_l^{u_i} u_{i,l}^{n+1} = Q_{u_i}^{n+1} - \left(\frac{\delta p^{n+1}}{\delta x_i} \right)_{\mathrm{P}} . \tag{7.29}$$

As always, P is the index of an arbitrary velocity node, and index l denotes the neighbor points that appear in the discretized momentum equation. The source term Q contains all of the terms that may be explicitly computed in terms of u_i^n as well as any body force or other linearized terms that may depend on the u_i^{n+1} or other variables at the new time level (like temperature) – hence the superscript $n+1$. The pressure term is written in symbolic difference form to emphasize the independence of the solution method from the discretization approximation for the spatial derivatives. The discretizations of the spatial derivatives may be of any order or any type described in Chap. 3.

Due to the non-linearity and coupling of the underlying differential equations, Eqs. (7.29) cannot be solved directly as the coefficients A and, possibly, the source term, depend on the unknown solution u_i^{n+1}. Iterative solution is the only choice; some approaches were described in Chap. 5. If we are computing an unsteady flow and time accuracy is required, iteration must be continued within each time step until the entire system of non-linear equations is satisfied to within a narrow tolerance. For steady flows, the tolerance can be much more generous; one can then either take an infinite time step and iterate until the steady non-linear equations are satisfied, or march in time without requiring full satisfaction of the non-linear equations at each time step.

The iterations within one time step, in which the coefficient and source matrices are updated, are called *outer iterations* to distinguish them from the *inner iterations* performed on linear systems with fixed coefficients. On each outer iteration, the equations solved are:

$$A_{\mathrm{P}}^{u_i} u_{i,\mathrm{P}}^{m*} + \sum_l A_l^{u_i} u_{i,l}^{m*} = Q_{u_i}^{m-1} - \left(\frac{\delta p^{m-1}}{\delta x_i} \right)_{\mathrm{P}} . \tag{7.30}$$

We dropped the time step index $n+1$ and introduced an outer iteration counter m; u_i^m thus represents the current estimate of the solution u_i^{n+1}. At the beginning of each outer iteration, the terms on the right hand side of Eq. (7.30) are evaluated using the variables at the preceding outer iteration.

The momentum equations are usually solved sequentially i.e. the set of algebraic equations for each component of the momentum is solved in turn, treating the grid point values of its dominant velocity component as the sole set of unknowns. Since the pressure used in these iterations was obtained from the previous outer iteration or time step, the velocities computed from Eqs. (7.30) do not normally satisfy the discretized continuity equation. To enforce the continuity condition, the velocities need to be corrected; this requires modification of the pressure field; the manner of doing this is described next.

The velocity at node P, obtained by solving the linearized momentum equations (7.30), can be formally expressed as:

$$u_{i,\mathrm{P}}^{m*} = \frac{Q_{u_i}^{m-1} - \sum_l A_l^{u_i} u_{i,l}^{m*}}{A_\mathrm{P}^{u_i}} - \frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p^{m-1}}{\delta x_i} \right)_\mathrm{P} . \tag{7.31}$$

As already stated, these velocities do not satisfy the continuity equation, so $u_{i\mathrm{P}}^{m*}$ is not the final value of the velocity for iteration m; it is a predicted value, which is why it carries an asterisk (*). The corrected final values should satisfy the continuity equation. For convenience, the first term on the right hand side of the above equations is called $\tilde{u}_{i,\mathrm{P}}^{m*}$:

$$u_{i,\mathrm{P}}^{m*} = \tilde{u}_{i,\mathrm{P}}^{m*} - \frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p^{m-1}}{\delta x_i} \right)_\mathrm{P} . \tag{7.32}$$

The velocity field $\tilde{u}_i^{m*}$ can be thought of as one from which the contribution of the pressure gradient has been removed. Because the method is implicit, this is not the velocity that would be obtained by dropping the pressure gradient entirely from Eq. (7.30).

The next task is to correct the velocities so that they satisfy the continuity equation:

$$\frac{\delta(\rho u_i^m)}{\delta x_i} = 0 , \tag{7.33}$$

which can be achieved by correcting the pressure field. The corrected velocities and pressure are linked by (see Eq. (7.32)):

$$u_{i,\mathrm{P}}^{m} = \tilde{u}_{i,\mathrm{P}}^{m*} - \frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p^{m}}{\delta x_i} \right)_\mathrm{P} . \tag{7.34}$$

Continuity is now enforced by inserting this expression for u_i^m into the continuity equation (7.33), to yield a discrete Poisson equation for the pressure:

$$\frac{\delta}{\delta x_i} \left[\frac{\rho}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p^m}{\delta x_i} \right) \right]_\mathrm{P} = \left[\frac{\delta(\rho \tilde{u}_i^{m*})}{\delta x_i} \right]_\mathrm{P} . \tag{7.35}$$

As noted earlier, the derivatives of the pressure inside the brackets must be discretized in the same way they are discretized in the momentum equations; the outer derivatives, which come from the continuity equation, must be approximated in the way they are discretized in the continuity equation.

After solving the Poisson equation for the pressure, (7.35), the final velocity field at the new iteration, u_i^m, is calculated from Eq. (7.34). At this point, we have a velocity field which satisfies the continuity condition, but the velocity and pressure fields do not satisfy the momentum equations (7.30). We begin another outer iteration and the process is continued until a velocity field which satisfies both the momentum and continuity equations is obtained.

This method is essentially a variation on the one presented in the preceding section. Methods of this kind, which first construct velocity field that does not satisfy the continuity equation and then correct it by subtracting something (usually a pressure gradient) are known as *projection methods.* The name is derived from the concept that the divergence-producing part of the field is projected out.

In one of the most common methods of this type, a pressure-correction is used instead of the actual pressure. The velocities computed from the linearized momentum equations and the pressure p^{m-1} are taken as provisional values to which a small correction must be added:

$$u_i^m = u_i^{m*} + u' \quad \text{and} \quad p^m = p^{m-1} + p' \tag{7.36}$$

If these are substituted into the momentum equations (7.30), we obtain the relation between the velocity and pressure corrections:

$$u'_{i,\mathrm{P}} = \tilde{u}'_{i,\mathrm{P}} - \frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p'}{\delta x_i} \right)_\mathrm{P} , \tag{7.37}$$

where $\tilde{u}'_i$ is defined by (see Eq. (7.31)):

$$\tilde{u}'_{i,\mathrm{P}} = - \frac{\sum_l A_l^{u_i} u'_{i,l}}{A_\mathrm{P}^{u_i}} . \tag{7.38}$$

Application of the discretized continuity equation (7.33) to corrected velocities and use of expression (7.37) produces the following pressure-correction equation:

$$\frac{\delta}{\delta x_i} \left[\frac{\rho}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p'}{\delta x_i} \right) \right]_\mathrm{P} = \left[\frac{\delta (\rho u_i^{m*})}{\delta x_i} \right]_\mathrm{P} + \left[\frac{\delta (\rho \tilde{u}'_i)}{\delta x_i} \right]_\mathrm{P} . \tag{7.39}$$

The velocity corrections $\tilde{u}'_i$ are unknown at this point, so it is common practice to neglect them. This is hard to justify and is probably the major reason why the resulting method does not converge very rapidly.

Alternative methods that are less brutal to the velocity correction will be described below. In the present method, once the pressure correction has been solved for, the velocities are updated using Eqs. (7.36) and (7.37). This is known as the SIMPLE algorithm (Caretto et al., 1972), an acronym whose origin will not be detailed. We shall discuss its properties below.

Simple C A more gentle way of treating the last term in the pressure-correction equation (7.39) is to approximate it rather than neglecting it. One could approximate the velocity correction u_i' at any node by a weighted mean of the neighbor values, for example,

$$u_{i,\mathrm{P}}' \approx \frac{\sum_l A_l^{u_i} u_{i,l}'}{\sum_l A_l^{u_i}} . \tag{7.40}$$

This allows us to approximate $\tilde{u}_{i,\mathrm{P}}'$ from Eq. (7.38) as

$$\tilde{u}_{i,\mathrm{P}}' \approx -u_{i,\mathrm{P}}' \frac{\sum_l A_l^{u_i}}{A_\mathrm{P}^{u_i}} , \tag{7.41}$$

which, when inserted in Eq. (7.37), leads to the following approximate relation between u_i' and p':

$$u_{i,\mathrm{P}}' = -\frac{1}{A_\mathrm{P}^{u_i} + \sum_l A_l^{u_i}} \left(\frac{\delta p'}{\delta x_i}\right)_\mathrm{P} . \tag{7.42}$$

With this approximation the coefficient $A_\mathrm{P}^{u_i}$ in Eq. (7.39) is replaced by $A_\mathrm{P}^{u_i} + \sum_l A_l^{u_i}$ and the last term disappears. This is known as the SIMPLEC algorithm (van Doormal and Raithby, 1984).

PISO Still another method of this general type is derived by neglecting $\tilde{u}_i'$ in the first correction step as in the SIMPLE method but following the correction with another corrector step. The second correction to the velocity u'' is defined by (see Eq. (7.37)):

$$u_{i,\mathrm{P}}'' = \tilde{u}_{i,\mathrm{P}}' - \frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p''}{\delta x_i}\right)_\mathrm{P} ; \tag{7.43}$$

where $\tilde{u}_i'$ is calculated from Eq. (7.38) after u_i' has been calculated from Eq. (7.37) with $\tilde{u}_i'$ neglected. Application of the discretized continuity equation (7.33) to corrected velocities leads to the second pressure-correction equation:

$$\frac{\delta}{\delta x_i}\left[\frac{\rho}{A_\mathrm{P}^{u_i}}\left(\frac{\delta p''}{\delta x_i}\right)\right]_\mathrm{P} = \left[\frac{\delta(\rho \tilde{u}_i')}{\delta x_i}\right]_\mathrm{P} . \tag{7.44}$$

Note that the coefficients on the left hand side are the same as in Eq. (7.39), which can be exploited (a factorization of the matrix may be stored and reused). Still further corrector steps can be constructed in the same way, but this is seldom done. This procedure is essentially an iterative method for solving Eq. (7.39) with the last term treated explicitly; it is known as the PISO algorithm (Issa, 1986).

$$\alpha_p = 1 + \frac{\sum_l A_l^{u_i}}{A_P^{u_i}} . \tag{7.49}$$

We may now recall that all conservative schemes, in the absence of any contribution to A_P from source terms, lead to $A_P = -\sum_l A_l + A_P^t$, where A_P^t is the contribution from the unsteady term. If a steady solution is sought through iterating for an infinite time step, $A_P^t = 0$, but we have to use under-relaxation, as explained in Sect. 5.4.2. In that case, $A_P = -\sum_l A_l/\alpha_u$, where α_u is the under-relaxation factor for velocities (usually the same for all components, but it need not be so). We then obtain:

$$\alpha_p = 1 - \alpha_u , \tag{7.50}$$

which has been found to be nearly optimum and yields almost the same convergence rate for outer iterations as the SIMPLEC method.

The solution algorithm for this class of methods can be summarized as follows:

1. Start calculation of the fields at the new time t_{n+1} using the latest solution u_i^n and p^n as starting estimates for u_i^{n+1} and p^{n+1}.
2. Assemble and solve the linearized algebraic equation systems for the velocity components (momentum equations) to obtain u_i^{m*}.
3. Assemble and solve the pressure-correction equation to obtain p'.
4. Correct the velocities and pressure to obtain the velocity field u_i^m, which satisfies the continuity equation, and the new pressure p^m.
   For the PISO algorithm, solve the second pressure-correction equation and correct both velocities and pressure again.
   For SIMPLER, solve the pressure equation for p^m after u_i^m is obtained above.
5. Return to step 2 and repeat, using u_i^m and p^m as improved estimates for u_i^{n+1} and p^{n+1}, until all corrections are negligibly small.
6. Advance to the next time step.

Methods of this kind are fairly efficient for solving steady state problems; their convergence can be improved by the multigrid strategy, as will be demonstrated in Chap. 11. There are many derivatives of the above methods which are named differently, but they all have roots in the ideas described above and will not be listed here. We shall show below that the artificial compressibility method can also be interpreted in a similar way.

7.4 Other Methods

7.4.1 Fractional Step Methods

In the methods of the preceding section, the pressure is used to enforce continuity. It is also used in computing the velocity field in the first step of the

Another similar method of this kind was proposed by Patankar (1980) and is called SIMPLER. In it, the pressure-correction equation (7.39) is solved first with the last term neglected as in SIMPLE. The pressure correction so obtained is used only to correct the velocity field so that it satisfies continuity i.e. to obtain u_i^m. The new pressure field is calculated from the pressure equation (7.35) using $\tilde{u}_i^m$ instead of $\tilde{u}_i^{m*}$. This is possible because u_i^m is now available.

SIMPLER

As already noted, due to the neglect of $\tilde{u}_i'$ in Eq. (7.39) (which is equivalent to neglecting it in Eq. (7.37)), the SIMPLE algorithm does not converge rapidly. Its performance depends greatly on the size of time step, or – for steady flows – on the value of the under-relaxation parameter used in the momentum equations. It has been found by trial and error that convergence can be improved if one adds only a portion of p' to p^{m-1}, i.e. if one takes

$$p^m = p^{m-1} + \alpha_p p' \tag{7.45}$$

after the pressure-correction equation is solved, where $0 \le \alpha_p \le 1$. SIMPLEC, SIMPLER and PISO do not need under-relaxation of the pressure correction.

One can derive an optimum relation between the under-relaxation factors for velocities and pressure by the following argument[1].

The velocities in the SIMPLE method are corrected by

$$u'_{i,\mathrm{P}} = -\frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p'}{\delta x_i} \right)_\mathrm{P} , \tag{7.46}$$

i.e., $\tilde{u}'_{i,\mathrm{P}}$ is neglected. To make up for this crudeness, we may now go back to the momentum equations (7.31) and look for pressure which will satisfy these equations when u_i^{m*} is replaced by corrected velocities u_i^m, which now satisfy the continuity equation (this is the path that leads to the pressure equation in SIMPLER). By assuming that the final pressure correction is $\alpha_p p'$, we arrive at the following equation:

$$u'_{i,\mathrm{P}} = \tilde{u}'_{i,\mathrm{P}} - \alpha_p \frac{1}{A_\mathrm{P}^{u_i}} \left(\frac{\delta p'}{\delta x_i} \right)_\mathrm{P} . \tag{7.47}$$

By making use of Eq. (7.46), we arrive at the following expression for α_p:

$$\alpha_p = 1 - \frac{\tilde{u}'_{i,\mathrm{P}}}{u'_{i,\mathrm{P}}} . \tag{7.48}$$

We can calculate $\tilde{u}'_{i,\mathrm{P}}$ using Eq. (7.38) but, in multi-dimensional problems, we would have more than one equation from which α_p can be calculated. However, if instead of calculating $\tilde{u}'_{i,\mathrm{P}}$, we use the approximation (7.41) used in SIMPLEC, then the above equation reduces to:

[1] Raithby and Schneider (1979) found this relation following a different route.