A CALCULATION PROCEDURE FOR TWO-DIMENSIONAL ELLIPTIC SITUATIONS

Suhas V. Patankar
Department of Mechanical Engineering, University of Minnesota,
Minneapolis, Minnesota 55455

A calculation method based on the control-volume approach has been developed for solving two-dimensional elliptic problems involving fluid flow and heat and mass transfer. The main features of the method include a power-law formulation for the combined convection-diffusion influence, an equation-solving scheme that consists of a block-correction method coupled with a line-by-line procedure, and a new algorithm for handling the interlinkage between the momentum and continuity equations. Although the method is described for steady two-dimensional situations, its extension to unsteady flows and three-dimensional problems is very straightforward.

INTRODUCTION

In 1972, Patankar and Spalding [1] described a calculation procedure for three-dimensional parabolic flows. A parabolic flow, such as the flow in a duct, is calculated by marching in the main flow direction, while solving a two-dimensional elliptic problem for each cross section. Thus, a three-dimensional parabolic procedure contains within it a procedure for two-dimensional elliptic flows. This has already been recognized, and the resulting two-dimensional procedure has been employed in many applications. Further, since this type of two-dimensional procedure employed velocity components and pressure as the dependent variables, it was easy to extend it to three dimensions simply by adding the third coordinate and the third velocity component. Such a calculation procedure for three-dimensional elliptic flows has been described in [2] and [3]. This procedure, too, has been used quite extensively. Incidentally, the particular technique by which the velocity-pressure linkage is handled in all these procedures has been given the name SIMPLE (semi-implicit method for pressure-linked equations).

Because the publications [2] and [3] are somewhat less widely known, [1] has remained the major reference for the whole SIMPLE family of procedures. While describing an application of the two- or three-dimensional elliptic procedure, the author and his co-workers have often characterized the calculation method simply as an "appropriate adaptation" of the procedure of [1]. Such a description has perhaps been rather uninformative, and even confusing, to the general reader.

The primary objective of this paper is to remedy this shortcoming. The paper attempts to set forth a complete description of the calculation procedure for steady two-dimensional elliptic flows. The restriction to two dimensions helps to keep the equations and diagrams easy to follow, However, the reader is urged to appreciate that the extension to three-dimensional elliptic situations is truly straightforward.
NOMENCLATURE

\[ a \] discretization coefficient
\[ A \] area of a control-volume face; also convection-diffusion function [Eq. (10)]
\[ b \] constant term in discretization equation
\[ d \] pressure coefficient [Eq. (36)]
\[ D \] diffusion conductance [Eq. (8)]
\[ F \] flow rate
\[ J \] total (convection plus diffusion) flux [Eq. (3)]
\[ p \] pressure
\[ p^* \] estimated pressure
\[ p' \] pressure correction
\[ P \] Peclet number [Eq. (7)]
\[ S \] source term [Eq. (11)]
\[ S_C \] constant part of linearized source term [Eq. (11)]
\[ S_p \] coefficient of \( \phi_p \) in source linearization [Eq. (11)]
\[ \bar{S} \] average value of source term over control volume [Eq. (35)]
\[ u_i \] velocity component in \( i \) direction

\[ \alpha \] Cartesian coordinates
\[ \alpha_p \] under-relaxation factor for pressure [Eq. (45)]
\[ \beta(x) \] diffusion coefficient [Eq. (1)]
\[ \Delta V \] volume of control volume
\[ \Delta \] density
\[ \phi \] general dependent variable
\[ \phi^* \] current estimate of \( \phi \)
\[ \phi_i \] block correction along \( i \)th line

Subscripts
\[ e, n, s, w \] control volume faces (Fig. 2)
\[ E, N, P, S, W \] grid points (Fig. 2)
\[ nb \] neighbor grid point

The numerical scheme will be worked out for Cartesian coordinates; but the method can be employed, without any additional novelty, for any orthogonal coordinate system. Further, the procedure can also be adapted to unsteady situations; however, that extension will not be discussed here.

This paper also serves some other objectives. After the original development of the procedure of [1], several minor and major changes in the procedure have taken place. These changes are incorporated in the description presented here. Among the minor changes are a more convenient definition of the control volume and a new convection-diffusion scheme called the power-law scheme. A major change is the development of a variant called SIMPLER.

Although the SIMPLE procedure has been quite successful and has been applied in a variety of situations, its rate of convergence has not always been satisfactory. With the new version SIMPLER (which stands for SIMPLE-revised), the converged solution is attained in fewer iterations, resulting in a significant saving of computer time. In a number of test problems, a saving of 30-50% has been commonly achieved. On occasion, SIMPLER has required only one-tenth of the computer time needed by SIMPLE. Furthermore, SIMPLER has been found to converge with greater certainty than SIMPLE. Often, under-relaxation factors have to be adjusted correctly so as to produce a reasonable convergence of the SIMPLE procedure. SIMPLER, on the other hand, seems to converge without any special adjustment. Relevant experiences with SIMPLE and SIMPLER have been reported by Raithby and Schneider [4], where the SIMPLER procedure has been referred to as the PUP scheme (pressure update-Patankar scheme).

In the description that follows, both techniques, SIMPLE and SIMPLER, are presented. Be able to app...
presented. Because of this arrangement, those already familiar with SIMPLE would be able to appreciate the additional features involved in SIMPLER. For the most part, the description in this paper focuses on what actual practices constitute the particular calculation procedure, rather than on why these practices were chosen. The reader may note that, although there are good reasons for adopting many of these practices, alternative approaches need not be ruled out.

MATHMATICAl FRAMEWORK

Governing Differential Equation

Practical situations involving fluid flow and heat and mass transfer are governed by the principles of conservation of mass, momentum, energy, chemical species, etc. These principles can be expressed in terms of partial differential equations. A major convenience for numerical formulation is provided by the recognition that these governing equations possess a common form. All that is necessary is to construct a numerical method for solving the differential equation of the common form.

If the dependent variable is denoted by \( \phi \), the corresponding steady-state form of the general differential equation is

\[
\frac{\partial}{\partial x_i} (\rho u_i \phi) = \frac{\partial}{\partial x_i} \left( \Gamma \frac{\partial \phi}{\partial x_i} \right) + S
\]

where the Cartesian-tensor notation has been used. The symbol \( u_i \) denotes the velocity component in the \( x_i \) direction, and \( \rho \) is the density. The term on the left-hand side of Eq. (1) is called the convection term, representing the flux of \( \phi \) convected by the mass flow rate \( \rho u_i \). The terms on the right-hand side are known as the diffusion term and the source term, respectively. The diffusion term is included in Eq. (1) because it is usually present, and often plays an important role, in any particular manifestation of \( \phi \). The source term \( S \) is primarily meant for representing the mechanisms for the generation (or destruction) of \( \phi \). But it can also be used as a general “dumping” ground; whatever cannot be conveniently expressed through the convection and diffusion terms can always be lumped into the source term. Because of this flexibility, the assumption that every dependent variable \( \phi \) is governed by Eq. (1) does not limit the physical processes or the types of the dependent variable that can be accommodated in the calculation procedure.

When the flow is turbulent, time-averaged forms of the governing equation are used. Then the quantities in Eq. (1) are interpreted as the time-mean values of the relevant variables. The additional terms arising in the time-averaging operation are expressed via appropriate redefinition of \( \Gamma \) and \( S \). For example, \( \Gamma \) may stand for the turbulent viscosity or eddy diffusivity.

Research into mathematical models of turbulence has shown that often it is desirable to compute, in addition to the time-mean properties of the flow variables, the time-averaged properties of the fluctuating motion itself. Thus, additional differential equations may have to be solved for turbulence parameters, such as the kinetic energy of turbulence and the length scale of turbulent eddies. Fortunately, these differential equations also possess the general form of Eq. (1), and the general solution technique continues to be applicable.
The dependent variable $\phi$ can stand for a variety of physical quantities. It can represent velocity components, enthalpy (or temperature), mass fractions of chemical species, turbulence parameters, radiation fluxes, and so on. Each meaning of $\phi$ is accompanied by appropriate expressions for $\Gamma$ and $S$ and appropriate boundary conditions. Once a portion of the computer program is written for the solution of Eq. (1), it can be repeatedly used with different meanings of $\phi$, the "DO loop" being executed the required number of times.

Complete expressions for $\Gamma$ and $S$ for different dependent variables will not be given here. These are available from standard textbooks. When one is faced with a new situation, it is assumed that at least a tentative mathematical model has been written down before beginning the computational task. However, some general comments about $\Gamma$ and $S$ will now be made.

**Diffusion Coefficient**

The diffusion coefficient $\Gamma$ is a general representation for the fluid properties such as viscosity or conductivity, which in conjunction with the gradient of the appropriate variable lead to the diffusion flux such as a viscous stress or a heat flux. For turbulent flow, many mathematical models replace the laminar (or molecular) values of $\Gamma$ by the corresponding turbulent (or effective) properties, for example, the turbulent viscosity or eddy diffusivity. If the diffusion flux for a given $\phi$ is not driven by the gradient of $\phi$, then $\Gamma$ may be set equal to zero and the diffusion flux expressed as a part of $S$. In general, the diffusion coefficient $\Gamma$ may be nonuniform; it may depend on position, as in a composite material, or on quantities such as the local velocity, temperature, density, etc.

**Source Term**

Internal heat generation in a fluid, production or destruction of a chemical species in a chemical reaction, the body force acting on a fluid element are common examples of the source term $S$ appearing in the equations for relevant variables. In a turbulence model, the equations for the turbulence parameters may contain production (and/or dissipation) terms. The source term $S$ in a momentum equation for the velocity component $u_i$ includes the pressure gradient $(-\partial p/\partial x_i)$, the body forces in the $i$ direction, and a contribution from the viscous stress terms that cannot be accommodated in the diffusion term.

**Continuity Equation**

Although the behavior of any dependent variable $\phi$ is governed by Eq. (1), the mass flow rate $\rho u_i$ appearing in it must satisfy an additional equation. This is the continuity equation for the flow field, which can be written for steady flows as

$$\frac{\partial}{\partial x_j} (\rho u_i) = 0$$

What is the additional degree of freedom that can be invoked to satisfy this extra equation? It is the pressure field, which enters the source terms of the momentum equations...
and which must adjust so that the resulting velocity field satisfies the continuity equation. This coupling between the velocity and pressure fields is used in constructing the calculation procedure for fluid flow.

**Boundary Conditions**

The boundaries of a calculation domain are usually of three kinds. They are no-flow, inflow, and outflow boundaries. At a no-flow boundary, an impermeable wall coincides with the boundary location. At such a boundary, either the value of the dependent variable $\phi$ is known or a relationship about the diffusion flux at the boundary is known. Thus, if $\phi$ stands for temperature, one may know the wall temperature, the wall heat flux, or an external convective heat transfer coefficient. At an inflow boundary, usually the value of $\phi$ in the incoming stream is known. At an outflow boundary, on the other hand, neither the value nor the flux of $\phi$ is known. However, if the numerical method is properly formulated, no information is indeed needed at an outflow boundary. This feature will later be explained with reference to the particular convection-diffusion formulation.

**NUMERICAL METHOD**

**Grid and Control Volumes**

In the numerical method to be described, the aim is to calculate the values of the relevant dependent variables at a set of chosen points called the grid points. The algebraic equations for these values (called the discretization equations) are derived by integrating the governing differential equation over a subdomain surrounding each grid point. These subdomains will be referred to as control volumes.

For a given grid, the control volumes can be defined in many ways. The practice chosen here is illustrated in Fig. 1. In this practice, it is convenient to decide the control-volume boundaries first and then place the grid points. As shown in Fig. 1, the calculation domain is divided into control volumes; the dashed lines denote the control-volume boundaries. Then, grid points are placed at the geometric centers of the control volumes. The solid lines in Fig. 1 are the grid lines, and the dots denote the grid points. For the two-dimensional situation shown, a given grid point communicates with four neighboring grid points through the four faces of the control volume. The situation with a near-boundary control volume is somewhat different; such a control volume is shown shaded.
in Fig. 1. Here, one face of the control volume coincides with the boundary of the calculation domain, and a boundary grid point is placed at the center of the control-volume face. (Another way of looking at this situation is to imagine a control volume of infinitesimal thickness located at the boundary.) This arrangement makes it rather easy to treat different boundary conditions; the shaded control volume can easily accept a given value of $\phi$ at the boundary or a given flux through the boundary surface.

Conservation Equation for the Control Volume

The general differential Eq. (1) can now be integrated over the control volume shown in Fig. 2. The control volume is constructed around the grid point $P$; the other grid points $E$, $W$, $N$, $S$ are the east, west, north, and south neighbors of $P$. The corresponding faces of the control volume are denoted by $e$, $w$, $n$, and $s$.

It is convenient to combine the convection and diffusion fluxes that appear in Eq. (1). Let $J_i$ denote the total (i.e., convection plus diffusion) flux in the $i$ direction. Then,

$$J_i = \rho u_i \phi - \Gamma \frac{\partial \phi}{\partial x_i}$$

(3)

The resulting differential equation is

$$\frac{\partial J_i}{\partial x_i} = S$$

(4)

The integration of this equation over the control volume shown in Fig. 2 leads to

$$J_w A_w - J_e A_e + J_s A_s - J_n A_n + S \Delta V = 0$$

(5)

Here the $J$'s represent appropriate total fluxes at the control-volume faces, $S$ is the average source term over the control volume, and $\Delta V$ is the volume of the control volume. Although for the Cartesian coordinates, the areas $A_w$ and $A_e$ are equal, separate symbols are used for generality.

Convection-plus-Diffusion Flux

Consider the region between grid points $P$ and $E$ in Fig. 2. If a one-dimensional convection-diffusion problem is solved between the points $P$ and $E$, an exponential

![Fig. 2. Typical control volume.](image-url)
PROCEDURE FOR ELLIPTIC SITUATIONS

The profile of $\phi$ is obtained as an exact solution. This is given by Spalding [5] and by Raithby and Torrance [6]. The exponential solution leads to

$$J_e = F_e \left( \phi_p + \frac{\phi_p - \phi_E}{\exp P_e - 1} \right)$$  \hspace{1cm} (6)

where the Peclet number $P_e$ is given by

$$P_e = \frac{F_e}{D_e}$$  \hspace{1cm} (7)

Here $F_e$ is the flow rate $(pu)_e A_e$ and $D_e$ is the diffusion conductance. If the diffusion coefficient $\Gamma$ is regarded as uniform over each control volume, the appropriate expression for $D_e$ is

$$D_e = A_e \left[ \frac{(\delta x)_e^-}{\Gamma_p} + \frac{(\delta x)_e^+}{\Gamma_E} \right]^{-1}$$  \hspace{1cm} (8)

where the distances $(\delta x)_e^-$ and $(\delta x)_e^+$ are shown in Fig. 2. The rationale for and the advantages of defining $D_e$ in this manner are discussed in [7].

Simplification of the Flux Expression

Because the exponential function appearing in Eq. (6) is time-consuming to compute, approximations to the flux expression have been sought. One such approximation is the hybrid scheme proposed in [5]. A better approximation to Eq. (6) is given by the following expression:

$$J_e = F_e \phi_p + (D_e A_e (P_e) + [-F_e, 0]) (\phi_p - \phi_E)$$  \hspace{1cm} (9)

where

$$A_e(P_e) = [0, (1 - 0.1 |P_e|)^5]$$  \hspace{1cm} (10)

Here the symbol $[a, b]$ is used to denote the greater of $a$ and $b$. Because of the presence of the fifth power in Eq. (10), the discretization scheme resulting from the use of Eqs. (9) and (10) is called the power-law scheme. It can be shown that the function $A$ in Eq. (10) is much easier to compute than the exponential function and that Eqs. (9) and (10) provide an extremely good approximation to the exact expression given in Eq. (6).

Source Term Linearization

If the source term $\bar{S}$ in Eq. (5) is known, no difficulty arises. But often the source term depends on the variable $\phi$ itself. In order that the resulting discretization equation remains (at least nominally) linear, the source term $\bar{S}$ is expressed as a linear function of $\phi_p$. Thus, the linearization formula can be written as

$$J_e = \phi_p + (S_p + S_p \phi_p) (\phi_p - \phi_E)$$  \hspace{1cm} (6)

where

$$S_p = \frac{\bar{S}}{\phi_E - \phi_p}$$  \hspace{1cm} (11)
where $S_P$ is the coefficient of $\phi_P$, and $S_C$ is the part of $S$ that does not explicitly depend on $\phi_P$.

What the best expressions for $S_C$ and $S_P$ are depends on the actual formula for $S$. Many alternative formulations are possible. Here it may be sufficient to note two main points. First, the source-term linearization is often a very crucial operation; it is responsible for computational success in many complex situations. Second, it is desirable that the linearization results in a negative $S_P$; a positive $S_P$ can cause divergence.

**Final Discretization Equation**

Substitution of the relations derived in the last two subsections into Eq. (5) leads to the final discretization equation. It can be written as

$$a_p \phi_p = a_E \phi_E + a_W \phi_W + a_N \phi_N + a_S \phi_S + b$$  \hspace{1cm} (12)

where

$$a_E = D_e A((P_e) + [F_e, 0])$$  \hspace{1cm} (13)

$$a_W = D_w A((P_w) + [F_w, 0])$$  \hspace{1cm} (14)

$$a_N = D_n A((P_n) + [F_n, 0])$$  \hspace{1cm} (15)

$$a_S = D_S A((P_s) + [F_s, 0])$$  \hspace{1cm} (16)

$$b = S_C \Delta V$$  \hspace{1cm} (17)

$$a_p = a_E + a_W + a_N + a_S - S_P \Delta V$$  \hspace{1cm} (18)

Here $F_e$, $F_w$, $F_n$, and $F_s$ are the mass flow rates through the respective control-volume faces. The Peclet numbers $P_e$, $P_w$, $P_n$, and $P_s$ are defined along the lines of Eq. (7), and the corresponding diffusion conductances according to Eq. (8). The function $A$ is defined by Eq. (10). The expression for $a_p$ as given by Eq. (18) results from the assumption that the flow rates $F_e$, $F_w$, $F_n$, and $F_s$ satisfy the continuity equation for the control-volume.

At this stage, it is useful to write Eq. (12) in a generalized form

$$a_p \phi_p = \sum a_{nb} \phi_{nb} + b$$  \hspace{1cm} (19)

where the subscript $nb$ denotes the neighbor grid points of $P$; the summation is to be taken over all the neighbors. For the two-dimensional situation for which Eq. (12) was derived, there are four neighbors. Six neighbors must be considered for a three-dimensional problem.

**Boundary Conditions**

With reference to Fig. 1 it can be seen that, since there is a control volume around each internal grid point, there will be a corresponding discretization equation such as Eq. (12). In this system of equations, the values of $\phi$ at the boundary points will appear. If these values (such as $\phi_B$) are known, the solution of the equations is straightforward. If, instead, some information about the boundary flux is given, it is necessary to write an equation similar to Eq. (6) for the boundary flux in terms of $\phi_B$ and $\phi_T$. By the use of this equation for the boundary conditions give is large (i.e. the volume knowledge premise is large.)

Nor

Fro

Ur

Eq
of this equation, the unknown value $\phi_B$ can be eliminated from the control-volume equation for the grid point $I$.

The outflow boundary condition requires special mention. Usually no information is available at the locations where the fluid leaves the calculation domain. The difficulty is resolved by a closer examination of the total flux formulation. The coefficient expressions given by Eqs. (13)-(16) are such that, when the magnitude of the Peclet number is large (i.e., greater than 10), the coefficient of the downstream neighbor is zero. Thus, the value $\phi_B$ at an outflow boundary influences $\phi_I$ only through a zero coefficient. No knowledge of $\phi_B$ is, therefore, needed. (It is true that this reasoning is based on the premise that the Peclet number is large, but usually it is not a serious approximation to assume that the value of $\Gamma$ at the outflow boundary is small and hence the Peclet number large.)

Nonlinearity

From the discussion about the boundary conditions, it can be concluded that Eq. (12) represents a set of enough equations for the evaluation of the grid-point values of $\phi$. If these equations were truly linear, a straightforward solution would yield the final answer. However, it must be recognized at this stage that these equations are only nominally linear. The coefficients in Eq. (12) may themselves depend on the values of $\phi$. Further, since $\phi$ can stand for a number of physical quantities, the coefficients for one meaning of $\phi$ may be influenced by some of the other $\phi$'s. Thus, when $\phi$ stands for temperature, its discretization coefficients may depend on velocity, turbulence parameters, concentration, and so on.

Because of these interlinkages and nonlinearities, the final solution is to be obtained by iteration. At any given stage, the discretization coefficients can be calculated from the current estimates of all the $\phi$ values. The solution of equations such as (12) then gives an improved estimate. When, after many repetitions of this process, all the $\phi$ values cease to change, the final converged solution is reached. Thus, the solution to a set of nonlinear and interlinked equations can be obtained via many intermediate solutions of nominally linear and decoupled algebraic equations.

It does not always follow that successive iterations would lead to a converged solution. At times, the values of $\phi$ oscillate or drift away from what can be considered a reasonable solution. Such divergence of the iteration process must be avoided.

Various ingredients of the discretization equations have been developed so as to minimize the chances of divergence. Further, it is desirable to slow down the changes in the coefficients from iteration to iteration. This can be accomplished by controlling the changes in the $\phi$ values via a technique called under-relaxation, which is discussed next.

Under-relaxation

Equation (19) can be rewritten as

$$\phi_P = \phi_P^* + \left[ \frac{\Sigma a_{nb} \phi_{nb} + b}{a_P} - \phi_P^* \right]$$

(20)
where \( \phi_p^* \) stands for the value of \( \phi_p \) from the previous iteration. The contents of the square brackets in Eq. (20) can be interpreted as the change in \( \phi_p \) in the current iteration. To reduce this change, an under-relaxation factor \( \alpha \) (lying between 0 and 1) can be introduced so that

\[
\phi_p = \phi_p^* + \alpha \left[ \frac{\sum a_{nb} \phi_{nb} + b}{a_p} - \phi_p^* \right]
\]

(21)

It should be noted that, when convergence is reached (i.e., \( \phi_p = \phi_p^* \)), the satisfaction of Eq. (21) also guarantees the satisfaction of Eq. (19). Further rewriting of Eq. (21) leads to

\[
\frac{a_p}{\alpha} \phi_p = \sum a_{nb} \phi_{nb} + b + \frac{(1 - \alpha)a_p \phi_p^*}{\alpha}
\]

(22)

The use of Eq. (22) with an appropriate value of \( \alpha \) introduces the desired under-relaxation into the system. If \( \alpha \) is close to zero, the values of \( \phi \) change very slowly. When \( \alpha \) is unity, no under-relaxation is effected. If \( \alpha \) is chosen to be greater than unity, the values of \( \phi \) change more rapidly than implied by Eq. (19); this behavior is known as over-relaxation.

**Solution of the Algebraic Equations**

The solution procedure described so far has assumed that a set of (nominal) linear algebraic equations can be solved by a convenient method. If the discretization equation such as Eq. (12) is written for a one-dimensional situation, it leads to a system with a tridiagonal matrix of coefficients. There is a particularly efficient algorithm to solve this system; it is known as the Thomas algorithm or the tridiagonal matrix algorithm (TDMA). It is commonly used in computations of one-dimensional unsteady heat conduction and of two-dimensional boundary layers. The details of the algorithm are given in many textbooks, such as [8].

However, such a convenient direct method for solving the discretization equations for two- or three-dimensional problems does not exist. The direct methods that are available require excessive storage and computer time. Further, since the set of nominally linear equations must be solved every iteration, the expense of direct solution seems unacceptable. Therefore, an iterative method for solving the algebraic equations is employed. The proposed method consists of two parts: the line-by-line technique and the block-correction procedure.

**Line-by-line technique.** The method uses the TDMA as its basic unit. In a two-dimensional problem, the equations for all the \( \phi \) values along one grid line are considered simultaneously. These equations, of course, also contain the \( \phi \) values along the two neighboring grid lines. If these neighbor-line values of \( \phi \) are substituted from the best available estimate, then the equations along the chosen line have the same appearance as the one-dimensional equations and can be solved simultaneously by the TDMA. When all the lines are visited in this manner, one iteration of the line-by-line technique is complete. The process can be repeated by choosing lines in the other direction.

**Block-correction procedure.** The speed of convergence of the line-by-line technique can be further increased by supplementing it with an additive-correction method, which is called the methods are d

Let the

where the sul

respectively. I

Let us assume

eliminated fr

substituted ar

is that

The ce

\( \phi_{ij} \) obtained

lines of const

The correctic

blocks define

is obtained b

of \( j \). The resu

where

The summat

written for i

(24) implies

needed.

It is i the \( \phi^* \) field.

A sim

constant \( j \).

block corr.

two directio
PROCEDURE FOR ELLIPTIC SITUATIONS

is called the block-correction procedure here. The general ideas of additive-correction methods are described by Settari and Aziz [9]. The details relevant to the present problem will be given here.

Let the discretization Eq. (12) be expressed as

$$a_{ij}\phi_{ij} = b_{ij}\phi_{i+1,j} + c_{ij}\phi_{i-1,j} + d_{ij}\phi_{i,j+1} + e_{ij}\phi_{i,j-1} + f_{ij}$$

(23)

where the subscripts i and j denote the grid-point positions in the x and y directions, respectively. Equations such as (23) can be written for all internal grid points for which

$$i = 2, M \quad j = 2, N$$

Let us assume that the boundary values \(\phi_{1,j}, \phi_{M+1,j}, \phi_{i,1}, \) and \(\phi_{i,N+1}\) have been either eliminated from Eq. (23) by use of the flux boundary condition or, if they are known, substituted and absorbed into \(f_{ij}\) for the near-boundary control volume. The consequence is that

$$b_{M,j} = 0 \quad c_{1,j} = 0 \quad d_{i,N} = 0 \quad e_{i,1} = 0$$

(24)

The central idea of the block-correction procedure is that an uncorrected field \(\phi^*_{ij}\) obtained from prior iterations is corrected by adding uniform corrections \(\delta_i\) along lines of constant \(i\). Thus

$$\phi_{ij} = \phi^*_{ij} + \delta_i$$

(25)

The corrections \(\delta_i\) are chosen such that the integral conservation over the control-volume blocks defined by each constant-\(i\) line is exactly satisfied. The equation determining \(\delta_i\) is obtained by substituting Eq. (25) into Eq. (23) and adding such equations for all values of \(j\). The result is

$$A_1\delta_i = B_1\delta_i + C_1\delta_i + D_i$$

(26)

where

$$A_1 = \Sigma (a_{ij} - d_{ij} - e_{ij})$$

(27)

$$B_1 = \Sigma b_{ij}$$

(28)

$$C_1 = \Sigma c_{ij}$$

(29)

$$D_i = \Sigma (b_{ij}\phi^*_{i+1,j} + c_{ij}\phi^*_{i-1,j} + d_{ij}\phi^*_{i,j+1} + e_{ij}\phi^*_{i,j-1} + f_{ij} - a_{ij}\phi^*_{ij})$$

(30)

The summations in these expressions are to be taken over \(j = 2, N\). Equation (26) can be written for \(i = 2, M\) and then the set can be conveniently solved by the TDMA. Since Eq. (24) implies that \(B_M\) and \(C_1\) will be zero, the boundary values \(\delta_1\) and \(\delta_{M+1}\) are not needed.

It is interesting to note that \(D_i\) is the integral residual in the \(i\)th block implied by the \(\phi^*\) field. The corrections \(\delta_i\) would reduce these integral residuals to zero.

A similar procedure can be worked out for the block correction along lines of constant \(j\). The complete procedure for solving the discretization equations includes the block corrections in the \(i\) and \(j\) directions and the line-by-line TDMA traverses in the two directions.
Considerations for Fluid Flow

The general differential Eq. (1) has been cast into the discretization form with the understanding that the flow field $\rho u_i$ as well as $\Gamma$ and $S$ is known. The density $\rho$ is algebraically related to the other variables such as concentration and temperature. The velocity components $u_i$ are governed by the momentum equations, which are special cases of the general differential Eq. (1). Thus, the fields of $u_i$ are to be obtained simply by interpreting the general dependent variable $\phi$ as the velocity $u_i$. In this sense, a calculation procedure for the flow field has already been described. That the discretization coefficients depend on the flow field is simply a case of nonlinearity, which can be handled by iteration. Is there any reason for giving special attention to the flow field calculation?

Main difficulty. A closer look at the governing equations reveals that it is not strictly true that the velocity components $u_i$ can be obtained by solving the momentum equations. These equations contain the pressure gradient $(-\partial p/\partial x_i)$ as an important source term, which is not expressible in terms of $u_i$ or other $\phi$’s. What indirectly determines the pressure field is the continuity equation, which was stated as Eq. (2) earlier. This indirect specification is not convenient as a computational procedure; a direct method of determining the pressure field must be found.

There are some other related difficulties in handling the momentum and continuity equations. It can be shown that, if the velocity components and the pressure are calculated for the same grid points, some physically unrealistic fields arise as solutions. A remedy for this ailment is the staggered grid, which is described next.

Staggered grid. Figure 3 shows a portion of a two-dimensional grid. The locations for which the velocity components are calculated are shown by short arrows. In the staggered grid, the velocity components are computed for the points lying on the control-volume faces. All other variables including pressure are calculated at the grid points shown by dots. A consequence of this arrangement is that the normal velocity components are directly available at the control-volume faces, where they are needed for the calculation of mass flow rates. Further, the pressure difference between two grid points can be used to “drive” the velocity component located between them.

Discretization Equations for Fluid Flow

Momentum equations. The staggered locations for the velocity components determine corresponding control volumes such that their faces normal to the direction of the component now pass through the grid points. The appropriate control volumes for $u$ and $v$ are shown in Fig. 4. Thus the two faces of the control volume around velocity $u$ pass through ten as

\[
\text{where the area are ident control} \ \ D\ \ \text{th}\
\]

\[
\text{It denote t like} \ u^* \ w
\]

\[
\text{Pro} \ \text{equation pressure}
\]

\[
\text{and}
\]

\[
\text{Equation formula}
\]

\[
\text{where}
\]

\[
\text{Th}
\]

![Fig. 3 Staggered grid.](image)

The diagram illustrates the staggered grid structure with control volumes aligning with the velocity components' directions.
th the y p is e. The special simply calculation can be v field is not entum ortant deter-eial, direct sion In the ontrol-points om-po or the points deter-of the u and e pass through the grid points P and E. The corresponding momentum equation can be written as

\[ a_e u_e = \sum a_{nb} u_{nb} + b + A_e (p_P - p_E) \]  

(31)

where the term \( b \) includes the source terms other than the pressure gradient, and \( A_e \) is the area over which the pressure force acts. The coefficient expressions for \( a_{nb}, a_e, \) and \( b \) are identical to those given in Eqs. (13)-(18), except that the staggered geometry of the control volume must be taken into account in determining the diffusion conductances \( D \) and the flow rates \( F \).

It is possible to solve the momentum equations for a given pressure field. Let \( u^* \) denote the velocity field based on an estimated pressure field \( p^* \). (In general, velocities like \( u^* \) will not satisfy the continuity equation.) This implies

\[ a_e u_e^* = \sum a_{nb} u_{nb} + b + A_e (p_P^* - p_E^*) \]  

(32)

Pressure-correction equation. In order that the velocity field satisfy the continuity equation, the "starred" velocities \( (u^* \) and \( v^* ) \) must be corrected as a consequence of a pressure correction \( p' \) applied to the estimated pressure \( p^* \). Thus,

\[ p = p^* + p' \]  

(33)

\[ u = u^* + u' \]  

(34)

Equations (31)-(34) can be manipulated to obtain a velocity-correction formula. This formula can be simplified if the term \( \Sigma a_{nb} u'_{nb} \) is considered negligible. The final result is

\[ u_e = u_e^* + d_e (p_P - p_E) \]  

(35)

where

\[ d_e = \frac{A_e}{a_e} \]  

(36)

The continuity equation for the control volume shown in Fig. 2 can be written as

\[ (\rho u A)_o - (\rho u A)_e + (\rho u A)_s - (\rho u A)_n = 0 \]  

(37)
where \( u \) and \( v \) denote the velocity components in the \( x \) and \( y \) directions, respectively. If \( u_e, u_w, v_n, \) and \( v_s \) are expressed via equations like (35), a discretization equation for the pressure correction can be obtained and cast into the form

\[
ap_{p_{e}}' = a_{E}p'_{E} + a_{W}p'_{W} + a_{N}p'_{N} + a_{S}p'_{S} + b
\]

where

\[
\begin{align*}
a_{E} &= \left( \rho dA \right)_{e} \\
a_{W} &= \left( \rho dA \right)_{w} \\
a_{N} &= \left( \rho dA \right)_{n} \\
a_{S} &= \left( \rho dA \right)_{s} \\
a_{F} &= a_{E} + a_{W} + a_{N} + a_{S} \\
b &= (\rho u^* A)_{w} - (\rho u^* A)_{e} + (\rho u^* A)_{s} - (\rho u^* A)_{n}
\end{align*}
\]

It may be noted that the expression for \( b \) implies that \( b \) is the “mass source” present in the starred velocity field. The task of the pressure correction is to annihilate this mass source.

**SIMPLE Procedure**

Now that the ingredients required for the flow field calculation have been assembled, the procedure can be described by listing the calculation steps.

1. Guess the pressure field \( p^* \).
2. Solve the momentum equations such as Eq. (32) to get \( u^* \) and \( v^* \).
3. Hence obtain the mass source \( b \) from Eq. (44) and solve the pressure-correction Eq. (38).
4. Correct the pressure field and the velocity field by the use of equations like (33) and (34).
5. Solve the discretization equations for other \( \phi \)'s if they influence the fluid flow through density, viscosity, or source terms.
6. Regarding the corrected pressure field \( p \) as a new guess \( p^* \), return to step 2, and repeat the procedure until convergence.

**Discussion of SIMPLE**

At this time, it is appropriate to consider the implications of the omission of the term \( \Sigma a_{n_{e}}u_{n_{e}} \) in the derivation of the pressure-correction equation. In the iterative calculation procedure, this omission does not influence the correctness of the final solution, because when the mass sources \( b \) are all zero, the pressure correction \( p'_f \) becomes zero. Thus, the converged solution correctly satisfies the momentum equations and also the continuity equation (as the mass sources are zero).

The rate of convergence to the final solution is, however, influenced by the omission. The approximate \( p'_f \) equation tends to overestimate the values of \( p'_f \), although the resulting velocity corrections are reasonable. Indeed, the SIMPLE procedure is prone to divergence unless some under-relaxation is used. A common practice is to employ an
under-relaxation factor $\alpha$ around 0.5 while solving the momentum equations, and to replace Eq. (33) by

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

(45)

with $\alpha_p$ around 0.8. Of course, it is to be understood that these relaxation factors are not necessarily the optimum ones and different problems may require a different set of values.

As mentioned earlier, although the SIMPLE procedure has been successfully used for a large number of problems, its rate of convergence has not always been satisfactory. A revised version, SIMPLER, has been formulated. It takes advantage of the property that, although the values of $p'$ may be overestimated, the associated velocity corrections are of the right magnitude. In SIMPLER, therefore, the $p'$ equation is used for the purpose of obtaining only the velocity corrections, while a separate equation is used for evaluating pressure.

**Pressure Equation**

Equation (31) can be rewritten as

$$u_e = \tilde{u}_e + d_e (p_p - p_E)$$

(46)

where the pseudovelocity $\tilde{u}_e$ is given by

$$\tilde{u}_e = \frac{\Sigma a_{nb} u_{nb} + b}{a_e}$$

(47)

It should be noted that $\tilde{u}_e$ does not depend on the pressure field but is defined in terms of the neighbor velocities $u_{nb}$. The similarity between Eqs. (35) and (46) is obvious. If formulas such as Eq. (46) for $u$ and $\tilde{u}$ are substituted into Eq. (37), an equation for the pressure $p$ can be derived. It is very similar to the pressure-correction equation and can be written as

$$a_p p_p = a_{EE} p_E + a_{WW} p_W + a_{NN} p_N + a_{SS} p_S + b$$

(48)

where the coefficients $a_{EE}, a_{WW}, a_{NN}, a_{SS}$, and $a_p$ are given by Eqs. (39)-(43), and $b$ is defined by

$$b = (\rho \tilde{u} A)_w - (\rho \tilde{u} A)_e + (\rho \tilde{u} A)_N - (\rho \tilde{u} A)_n$$

(49)

Again, the term $b$ here can be regarded as a "mass source" implied by the pseudovelocity field $\tilde{u}, \tilde{v}$.

**SIMPLER Procedure**

The revised procedure will now be outlined in terms of the calculation sequence.

1. Guess a velocity field.
2. Calculate the coefficients in the momentum equations and hence obtain $\tilde{u}$ and $\tilde{v}$.
from equations like (47) by substituting the values of the neighbor velocities. (Note that this is simply a substitution; no solution of simultaneous equations is involved.)

3. Evaluate the mass source \( b \) from Eq. (49) and hence solve Eq. (48) to obtain the pressure field \( p \).

4. Regarding this pressure field as \( p^* \), solve the momentum equations such as (32) to obtain \( u^* \) and \( v^* \).

5. Calculate the mass source \( b \) from Eq. (44) and hence solve the \( p' \) equation.

6. Using the \( p' \) field, correct the starred velocities via equations such as (35). Do not correct pressure.

7. Solve the discretization equations for other \( \phi \)'s if necessary.

8. Return to step 2 with the corrected velocity field and repeat the procedure until convergence.

Discussion of SIMPLER

Although the \( p' \) and \( p \) equations have identical appearance and very similar derivations, there is one important difference. The term \( \sum a_{nb} u_{nb} \) was dropped while deriving the \( p' \) equation, while no such approximation was used in obtaining the \( p \) equation. Thus, the pressure field obtained from the \( p \) equation is likely to be more correct than the pressure field constructed from the \( p' \) values. Indeed, if the guessed velocity field happened to be the correct velocity field, the \( p \) equation would yield the correct pressure field. It is from this characteristic of the pressure equation that the SIMPLER procedure derives its effectiveness.

An initial guessed pressure field plays a major role in SIMPLE; on the other hand, SIMPLER does not require guessed pressures but internally generates a pressure field that is compatible with the velocity field. In most problems, it is easier to guess a reasonable velocity field than to propose a good pressure field.

Because the pressures calculated in SIMPLER are closer to the correct solution, the need for under-relaxation is greatly reduced. The experience so far indicates that, for the momentum equations, a relaxation factor \( \alpha \) around 0.75 is sufficient, while the pressure is not under-relaxed at all.

It is true that one iteration of SIMPLER requires about 30% more computational effort than one iteration of SIMPLE. However, the extra effort per iteration is amply compensated by a drastic reduction in the number of iterations required for convergence.

REFERENCES


*Received September 29, 1981
Accepted October 19, 1981*