I. INTRODUCTION

The equations of fluid motion and accompanying boundary conditions are difficult to formulate and solve. For most practical problems, the equations and accompanying boundary conditions must be numerically solved. The types of numerical methods used by the majority of researchers and applications-oriented engineers to solve these partial differential equations fall into three categories: 1) finite difference (or finite volume) methods (FDM), 2) finite element methods (FEM), and 3) other approaches (boundary integrals, hybrids, analytical, spectral, etc.). The FDM has been used for a wide variety of problems; nearly all of the early numerical simulations dealing with heat transfer and fluid flow revolved around various solution strategies for the FDM.

As structural analyses using FEM became more routine, researchers began to apply the method to more difficult problem areas, particularly those fields dealing with fluid flow. Some of the earliest work in fluid simulation with finite elements can be traced to the mid-1960s; a comprehensive review is given in Zienkiewicz and Taylor [4.01]. Based on this early pioneering work, the numerical simulation of fluid flow with finite elements began to proliferate by the 1970s. Today, the FEM is a strong contender for simulating all modes of fluid flow processes, rivaling performance standards associated with FDM.

The two most often used ways to formulate the FEM are the Rayleigh-Ritz variational method and the Galerkin Method of Weighted Residuals (MWR). Both approaches use a combination of appropriate functions to approximate the solution. The unknown coefficients are determined using integral statements in such a way as to approximately satisfy the original differential equations. However, there is a major difference between the Rayleigh-Ritz method and the Galerkin method. The Rayleigh-Ritz method finds the unknown coefficients through an energy minimization process; this process requires a minimum principle. The Galerkin method is based on making the projection of the error in the approximating functions vanish in the finite dimensional space spanned by the functions. This approach allows the Galerkin method to be used in situations when minimum principles do not exist. Such cases occur when convection is the dominant transport mechanism in a fluid system. The Galerkin method is therefore the method of choice in problems involving fluid flow.

In general, the following steps are needed in any finite element approximation to the solution of a differential equation:

1) The equation (or system of equations) and its boundary and initial conditions must be defined to ensure that a well-posed problem is formulated,
2) An element type must be chosen to define the approximation functions to be used in the solution,
3) A mesh must be created that adequately refines regions where large changes in the solution are expected, and that allows the boundary conditions to be properly imposed,
4) The finite element algorithm must be formulated and used to solve the system of algebraic equations
5) The error in the approximation must be calculated to determine if the solution is converged or if a more refined solution is needed.

II. GOVERNING EQUATIONS

A. Mass Conservation

The most general form of the mass conservation or *continuity* equation, using indicial form, is

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_j}{\partial x_j} = 0 \qquad \text{Eq. (4-1)}$$

where ρ is the fluid density, u_j is velocity (here j = 1, 2, 3, denotes u, v, and w in the x-, y- and z-directions, respectively), and t denotes time. If the fluid is incompressible, Eq. (4-1) becomes

$$\frac{\partial u_j}{\partial x_j} = 0 \qquad \text{Eq. (4-2)}$$

which is also called the *dilatation* term and corresponds to the change of volume of a fluid particle.

B. Navier-Stokes

The Navier-Stokes equations for a Newtonian viscous fluid, also known as the momentum equations, can be written as

$$\rho\left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j}\right) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) + \lambda \frac{\partial u_k}{\partial x_k}\delta_{ij}\right] + \rho B_i \qquad \text{Eq. (4-3)}$$

Replacing λ by $-2/3\mu$ yields the usual form for compressible flow. If the flow is incompressible, Eq. (4-3) reduces to

$$\rho\left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j}\right) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\mu \frac{\partial u_i}{\partial x_j}\right) + \rho B_i \qquad \text{Eq. (4-4)}$$

For compressible flows the equation of state is required, given by the relation

$$p = \rho RT \qquad \text{Eq. (4-5)}$$

where R is the gas constant and T denotes temperature.

C. Energy Conservation

The energy equation follows from the first law of thermodynamics. In terms of internal energy per unit mass, conservation of energy is written as

$$\rho\left(\frac{\partial e}{\partial t} + u_j \frac{\partial e}{\partial x_j}\right) = -\frac{\partial q}{\partial x_j} + \sigma_{ij} \frac{\partial u_i}{\partial x_j} + f \qquad \text{Eq. (4-6)}$$

The heat transfer q is given by Fourier's law, $q = -k\nabla T$,

where k is the material's conductivity and f is a volumetric heat source or sink. For a calorically perfect fluid, we can relate the internal energy to temperature through the thermodynamic equation

$$de = c_V \, dT \qquad \text{Eq. (4-7)}$$

where c_V is the specific heat at constant volume. With this assumption, Eq. (4-7) becomes

$$\rho c_v \left(\frac{\partial T}{\partial t} + u_j \frac{\partial T}{\partial x_j} \right) = \frac{\partial}{\partial x_j} (k \frac{\partial T}{\partial x_j}) + \sigma_{ij} \frac{\partial u_i}{\partial x_j} + f \qquad \text{Eq. (4-8)}$$

For an incompressible fluid, Eq. (4-8) is written in the form

$$\rho c_v \left(\frac{\partial T}{\partial t} + u_j \frac{\partial T}{\partial x_j} \right) = \frac{\partial}{\partial x_j} (k \frac{\partial T}{\partial x_j}) + \phi + f \qquad \text{Eq. (4-9)}$$

where ϕ is the viscous heat dissipation function and is defined as

$$\phi = \left[\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \lambda \frac{\partial u_k}{\partial x_k} \delta_{ij} \right] \frac{\partial u_i}{\partial x_j} \qquad \text{Eq. (4-10)}$$

In many applications involving incompressible fluids, the dissipation term is small and is neglected.

D. Mass Transport

The equation for mass transport is written as

$$\frac{\partial c}{\partial t} + u_i \frac{\partial c}{\partial x_i} = \frac{\partial}{\partial x_i} \left(D \frac{\partial c}{\partial x_i} \right) + S \qquad \text{Eq. (4-11)}$$

where c is the species component, D is the mass diffusion coefficient, and S represents sources and/or sinks. In general, one equation of the form of Eq. (4-11) is needed for each component in the fluid. The coupling between the equations can be quite complicated, particularly in the case of flows with chemical reactions.

E. Boundary Conditions

The boundary conditions must be physically realistic and are dependent on the particular geometry, the materials involved, and the values of pertinent parameters. At a solid boundary, the velocity is zero, i.e., a *no-slip* boundary condition. However, the no-slip boundary condition is only valid when the continuum hypothesis is justified; it is not a realistic boundary condition at a solid boundary for a gas with a large mean free path. In this case there is a slip velocity of the gas relative to the solid boundary.

In general, the pressure is not required to satisfy boundary conditions, but one must prescribe a reference value for it to be determined uniquely. In some flow problems the known pressure at a free surface provides the equilibrium condition along that interface, or the prescribed pressure at an open fluid boundary is the driving force in the system. In these cases, the pressure is prescribed along these boundaries.

III. THE FINITE ELEMENT METHOD

To illustrate finite element methodology, assume a linear operator (L) exists in two-dimensions such that

$$L \equiv -\nabla \cdot k \nabla = -\frac{\partial}{\partial x}\left(k \frac{\partial}{\partial x} \right) - \frac{\partial}{\partial y}\left(k \frac{\partial}{\partial y} \right) \qquad \text{Eq. (4-12)}$$

One must now reformulate the basic problem in a way appropriate for the application of the FEM. Let an equation exist of the form

$$Lu(\mathbf{x}) - f(\mathbf{x}) = 0$$

where u and f are functions of (x). Now define the *residual* function as

$$R(u, \mathbf{x}) \equiv Lu(\mathbf{x}) - f(\mathbf{x}) \qquad \text{Eq. (4-13)}$$

It then follows that if u* is the solution to the differential equation, then R(u*,**x**) ≡ 0. However, if u is only an approximation to the solution, the residual provides a measure of the error in the satisfaction of the equation.

If we now multiply Eq. (4-13) by a weighting function w defined over domain (space) Ω, integrate over Ω, and set the integral equal to zero, we obtain the *weighted residuals* form

$$\int_\Omega w(x) R(u, x) \, d\Omega = \int_\Omega w(Lu - f) \, d\Omega = 0 \qquad \text{Eq. (4-14)}$$

Hence,

$$-\int_\Omega \left[w\left(\frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y}\left(k \frac{\partial T}{\partial y} \right) \right) + w f \right] d\Omega = 0$$

Eq. (4-15)

We now apply Green's theorem to the second integral terms, i.e.,

$$\int_\Omega \left[\frac{\partial}{\partial x}\left(k w \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y}\left(k w \frac{\partial T}{\partial y} \right) \right] d\Omega = \int_\Gamma \left(k w \frac{\partial T}{\partial x} dy - k w \frac{\partial T}{\partial y} dx \right)$$

Eq. (4-16)

Using the fact that the components of the unit outward normal to Γ are $n_x = dy/d\Gamma$ and $n_y = -dx/d\Gamma$, the line integrals in Eq. (4-16) become

$$\int_\Gamma \left(k w \frac{\partial T}{\partial x} n_x + k w \frac{\partial T}{\partial y} n_y \right) d\Gamma = \int_\Gamma k w \frac{\partial T}{\partial n} d\Gamma$$

Eq. (4-17)

where n denotes the normal to the surface. Hence, Eq. (4-16) can be rewritten as

$$\int_\Omega \left[k \frac{\partial w}{\partial x} \frac{\partial T}{\partial x} + k \frac{\partial w}{\partial y} \frac{\partial T}{\partial y} - w f \right] d\Omega + \int_\Gamma w \left(-k \frac{\partial T}{\partial n} \right) d\Gamma = 0$$

Eq. (4-18)

Weak weighted residual formulations for any second-order linear differential operator can be obtained in the manner described above. Nonlinear problems must be treated on a case-by-case basis, but we can always generate a weak form.

A. Error in Finite Element Approximation

The mathematical properties of finite element approximation have been thoroughly studied, and there are many excellent books that contain detailed information [4.02, 4.03, 4.04]. The amount of work that has been done in the area of stability and error analysis is exhaustive.

To obtain useful error estimates, one can make use of the relation

$$\left\| u^* - u^h \right\|_o \le C_1 h^2 \left\| u^* \right\|_2 \qquad \text{Eq. (4-19)}$$

where h, defined as $h = \max(|x_i - x_j|, |y_i - y_j|)$, is the maximum mesh size in either the x- or y-direction, and the norm $\|\bullet\|_2$, is defined as

$$\| u \|_2 = \left(\int_\Omega \left[u^2 + \left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 + \left(\frac{\partial^2 u}{\partial x^2} \right)^2 + \left(\frac{\partial^2 u}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 u}{\partial y^2} \right) \right] d\Omega \right)^{\frac{1}{2}}$$

Eq. (4-20)

B. One-dimensional Elements

1. Linear Element

Consider a piecewise polynomial approximation of the domain $0 < x < 1$. Divide the domain into two equal intervals, and seek a solution that is linear over each of the subintervals. A linear function u between two nodal points x_i and x_{i+1} can be written as

$$u(x) = \left[\frac{x_{i+1} - x}{x_{i+1} - x_i} \right] u_i + \left[\frac{x - x_i}{x_{i+1} - x_i} \right] u_{i+1} \qquad \text{Eq. (4-21)}$$

Thus,

$$u(x) = \sum N_i(x) u_i = N_1 u_1 + N_2 u_2 + N_3 u_3 \qquad \text{Eq. (4-22)}$$

where

$$N_1(x) = \begin{cases} 1 - 2x & 0 \le x \le \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \qquad \text{Eq. (4-23a)}$$

$$N_2(x) = \begin{cases} 2x & 0 \le x \le \frac{1}{2} \\ 2 - 2x & \frac{1}{2} \le x \le 1 \\ 0 & \text{otherwise} \end{cases} \qquad \text{Eq. (4-23b)}$$

$$N_3(x) = \begin{cases} 2x - 1 & \frac{1}{2} \le x \le 1 \\ 0 & \text{otherwise} \end{cases} \qquad \text{Eq. (4-23c)}$$

The functions $N_i(x)$, i = 1, 2, 3, are shown in Fig. 4-1 and are called *shape* functions, *trial* functions, or *basis* functions.

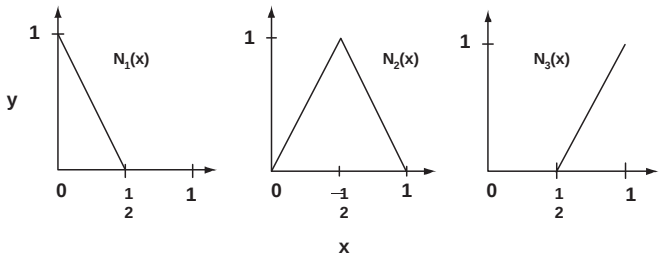


Figure 4-1. Linear 1-D shape functions for a two-element approximation.

The functions have *local support*, i.e., they vanish outside a maximum of two elements. The implementation of Dirichlet conditions is trivial, e.g., to impose u(0) = 0, simply set $u_1 = u_1(x_1) = u(0) = 0$. In a Galerkin formulation, set $w_i(x) = N_i$, and the linear shape functions take the generic form over each element shown in Fig. 4-2.

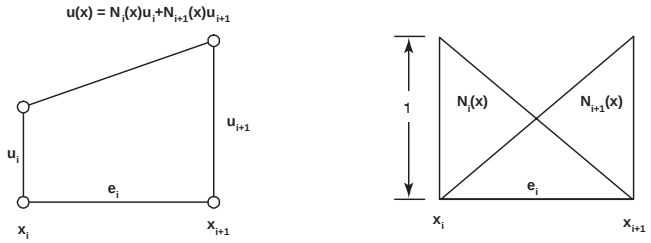


Figure 4-2. Linear element interpolation and element shape functions.

2. Quadratic and Higher Order Elements

To obtain higher order elements, we must introduce more nodes in the elements. For example, if we desire to use quadratic polynomials over an element, a function u(x) will be approximated as

$$u(x) \cong a + b x + c x^2 \qquad 0 \le x \le h \qquad \text{Eq. (4-24)}$$

which contains three unknown parameters. To determine the shape functions we place three nodes within the element, one at each end of the interval and one at the midpoint. Setting the nodes at $x_1 = 0$, $x_2 = h/2$, and $x_3 = h$, the shape functions become

$$N_1(x) = 1 - \frac{3x}{h} + \frac{2x^2}{h^2}, \quad N_2(x) = \frac{4x}{h}\left(1 - \frac{x}{h}\right), \quad N_3(x) = \frac{x}{h}\left(\frac{2x}{h} - 1\right)$$

Eq. (4- 25)

when $0 \le x \le h$ and zero otherwise.

Figure 4-3 shows the local quadratic shape functions. A finite element approximation based on quadratic elements will be more accurate than one based on linear elements. A cubic element consists of two interior nodes located at a distance of h/3 to each end.

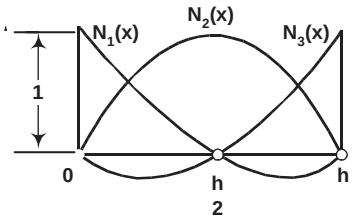


Figure 4-3. One-dimensional quadratic shape function.

Choosing to approximate the function u with a polynomial of degree n, so that the element has n + 1 nodes located at $x_i = (i-1)h/n$, i = 1, 2, …, n + 1, the shape functions can be readily obtained using Lagrange's formula

$$N_i(x) = \begin{cases} \dfrac{(x-x_1)(x-x_2)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_{n+1})}{(x_i-x_1)(x_i-x_2)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_{n+1})} & \text{for } 0 \le x \le h \\ 0 \quad \text{otherwise} & \text{for } i = 1, \cdots, n+1 \end{cases} \qquad \text{Eq. (4-26)}$$

It is easy to recover the expressions for linear shape functions when n = 1 and for quadratic elements when n = 2.

There are two ways in which a finite element approximation to any problem can be improved. The first consists in increasing the number of elements used in the mesh, therefore decreasing the size of h and, consequently, the error. This is called the h-method and relies on decreasing the size of the mesh to achieve better accuracy, utilizing always the same element. The second possibility is to keep the number of elements fixed and to increase the degree of the interpolation polynomials in the elements. In this way the number of nodes is increased and so is the order of the element. This is called the p-method. Of course, a combination of both can also be used, and this is referred to as the h-p method.

There is a very significant improvement going from linear to quadratic elements. However, the gains going from quadratic to cubic elements are marginal. On the other hand, the calculation cost increases considerably as we increase the order of the elements. To obtain the element stiffness matrices requires significantly more operations for the higher order elements. However, more important is the fact that the bandwidth of the coefficient matrix becomes larger with higher order elements. In many cases a few quadratic elements will yield solutions of much better accuracy than a much larger number of linear elements, and their use is therefore desirable.

There are other important families of elements based on piecewise polynomial interpolation that are not of the Lagrangian type. An example is Hermite polynomials, which are based on interpolating derivatives as well as the function at the nodes. The cubic Hermite two-noded element is an element for which both the function and its first derivative are continuous over the entire domain Ω. This element is widely used in finite element simulations of beam bending in solid mechanics [4.05]. It has also been successfully used to simulate transport in aquifers [4.06]. Smooth finite element approximations can also be obtained using splines as shape functions.

C. Two-dimensional Elements

1. Triangular Elements

The simplest two-dimensional figure that defines an area is the triangle. The simplest triangular element is obtained defining a linear interpolation field of the form

$$u(x,y) \cong a + b\,x + c\,y \qquad \text{Eq. (4-27)}$$

and placing the nodes at the corners of the triangle. The shape functions can be written in terms of the nodal coordinates as

$$N_1(x,y) = \frac{1}{2A}\left[x_2y_3 - x_3y_2 + (y_2 - y_3)\,x + (x_3 - x_2)\,y\right]$$

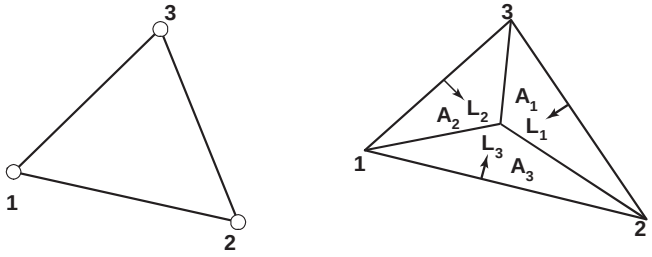
$$N_2(x,y) = \frac{1}{2A}\left[x_3y_1 - x_1y_3 + (y_3 - y_1)\,x + (x_1 - x_3)\,y\right]$$

$$N_3(x,y) = \frac{1}{2A}\left[x_1y_2 - x_2y_1 + (y_1 - y_2)\,x + (x_2 - x_1)\,y\right] \qquad \text{Eq. (4-28)}$$

where the area A is given by

$$2A = (x_2y_3 - x_3y_2) + (x_3y_1 - x_1y_3) + (x_1y_2 - x_2y_1) \qquad \text{Eq. (4-29)}$$

and the nodes are numbered counterclockwise as in Fig. 4-4. These elements are discussed by Pepper and Heinrich [4.07]



(a) element nodal numbering (b) natural coordinate system

Figure 4-4. Linear triangular element (a) and area natural coordinate system (b).

The shape functions can be more easily obtained if we use area coordinates. Joining any point P in the triangle to the vertices of the triangle, the three areas, A_1, A_2, and A_3, can be defined as shown in Fig. 4-5.

A coordinate system that uniquely represents every point in the triangle is given by

$$L_i = \frac{A_i}{A} \qquad i = 1,2,3 \qquad \text{Eq. (4-30)}$$

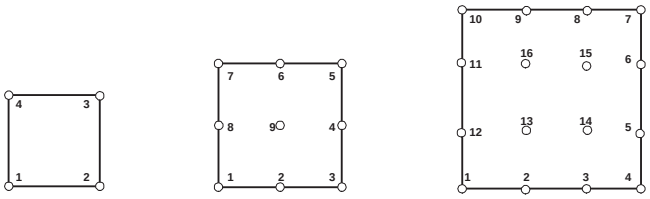
If the nodes are uniformly distributed along the element sides, the shape functions are easily constructed in this coordinate system, also called the *natural* coordinate system for the triangle.

The shape functions in natural coordinates are independent of the shape of the triangle. This is particularly appealing when we are dealing with highly irregular geometries that may require a large variety of very differently shaped triangles. The ability of the triangle to discretize any kind of geometric figure with relative ease is the main reason for the wide use of triangular elements. From this point of view, triangular elements are always better than quadrilateral elements. Very powerful mesh generators have been developed based on the triangular geometry that can automatically discretize extremely complex regions. Although in the last years much progress has been made in this area using quadrilateral elements, these mesh generators still lack the versatility and degree of automation of those based on triangles.

2. Quadrilateral Elements

A quadrilateral element is defined by four corner points and therefore is no longer linear, which makes it more complex than a triangular element. However, there is a greater variety of quadrilateral elements, and in general, they can offer many advantages over the use of triangles.

The simplest way to obtain rectangular elements consists in taking the product (also referred to as the tensor product) of one-dimensional elements. In this fashion we generate the family of Lagrangian elements that are bilinear, biquadratic, etc., and contain 2^2, 3^2, 4^2, … nodes as shown in Fig. 4-5. To obtain the shape functions we only need to know the form of the shape functions in one dimension. The two-dimensional function at a node is obtained as the product of the one-dimensional functions that would correspond to that node in the x and y directions, respectively.



(A) Bilinear **(b) Biquadratic** **(c) Bicubic**

Figure 4-5. 2-D quadrilateral elements, (a) bilinear, (b) biquadratic, and (c) bicubic.

For the 4-node bilinear and nine-node biquadratic elements, the shape functions are defined in Table 1.

Another important family of rectangular elements is known as the *serendipity* elements. These elements differ from the Lagrangian family in that they do not contain any interior nodes. Examples of the eight-noded quadratic and the 12-noded cubic elements are shown in Fig. 4-6.

Table 1.
Bilinear and biquadratic element shapes for rectangular element

Bilinear

$$N_1(x,y) = \frac{1}{4ab}(b-x)(a-y)$$

$$N_2(x,y) = \frac{1}{4ab}(b+x)(a-y)$$

$$N_3(x,y) = \frac{1}{4ab}(b-x)(a+y)$$

$$N_4(x,y) = \frac{1}{4ab}(b-x)(a+y)$$

Biquadratic

$$N_1(x,y) = \overline{N}_1(x)\overline{N}_1(y) = \frac{1}{4a^2b^2}xy(x-a)(y-b)$$

$$N_2(x,y) = \overline{N}_2(x)\overline{N}_1(y) = \frac{1}{2a^2b^2}\left(a^2-x^2\right)y\,(y-b)$$

$$N_3(x,y) = \overline{N}_3(x)\overline{N}_1(y) = \frac{1}{4a^2b^2}xy(x+a)(y-b)$$

$$N_4(x,y) = \overline{N}_3(x)\overline{N}_2(y) = \frac{1}{2a^2b^2}x(x+a)\left(b^2-y^2\right)$$

$$N_5(x,y) = \overline{N}_3(x)\overline{N}_3(y) = \frac{1}{4a^2b^2}xy(x+a)(y+b)$$

$$N_6(x,y) = \overline{N}_2(x)\overline{N}_3(y) = \frac{1}{2a^2b^2}\left(a^2-x^2\right)(y+b)$$

$$N_7(x,y) = \overline{N}_1(x)\overline{N}_3(y) = \frac{1}{4a^2b^2}(x-a)(y+b)$$

$$N_8(x,y) = \overline{N}_1(x)\overline{N}_2(y) = \frac{1}{2a^2b^2}x(x-a)\left(b^2-y^2\right)$$

$$N_9(x,y) = \overline{N}_2(x)\overline{N}_2(y) = \frac{1}{a^2b^2}\left(a^2-x^2\right)\left(b^2-y^2\right)$$

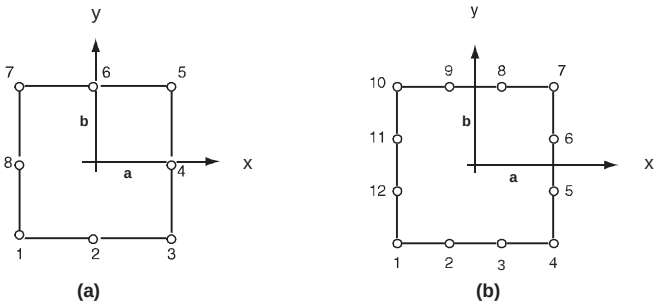


Figure 4-6. Serendipity elements for (a) quadratic and (b) cubic.

The shape functions for these elements were found by a trial and error process, thus the name serendipity. However, with the development of blending function interpolation techniques, a systematic way to construct the shape functions is now available (See Heinrich and Pepper [4.08].). For the

$$u(x,y,t) = \sum_j N_j(x,y)\,u_j(t),\ v(x,y,t)$$
$$= \sum_j N_j(x,y)\,v_j(t),\ p(x,y,t)$$
$$= \sum_j M_k(x,y)\,p_k(t)$$

In order to satisfy the LBB condition, the pressure must be interpolated with a lower order polynomial than the velocity components. Figure 4-23 shows the simplest and most often used combination of such mixed interpolation. The combinations 1, 3 and 5 involve discontinuous pressure fields. The evidence from numerical experiments indicates that these approximations to the pressure field result in improved accuracy over those combinations using continuous pressures.

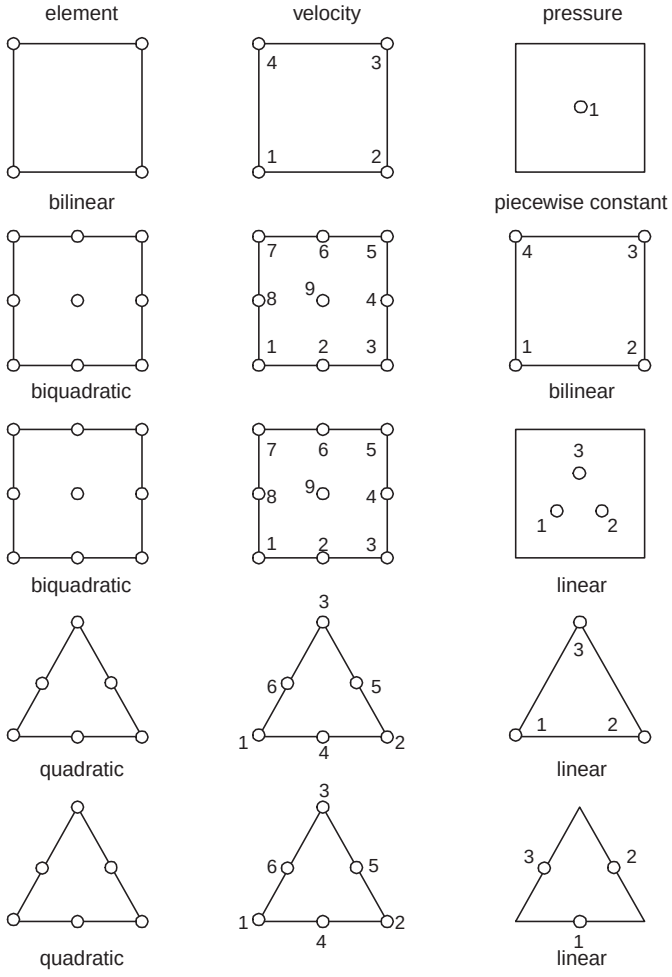


Figure 4-23. Elements for mixed interpolation of velocity and pressure.

Substituting into the weak formulation given by Eqs. (4-77)–(4-79), and setting the weighting functions U, V, and P equal to the shape functions, N_i, N_i, and M_i, respectively, yields the discrete Galerkin equations. For velocity interpolated to n-noded elements and the pressure using m points,

$$\mathbf{M}\dot{\mathbf{d}} + \mathbf{K}\mathbf{d} + \mathbf{F} = \mathbf{0} \qquad \text{Eq. (4-82)}$$

The dimension of the system is (2n + m) x (2n + m). To simplify the notation, the degrees of freedom are ordered so that

$$\mathbf{d} = \left[u_1 u_2 \ldots u_n v_1 v_2 \ldots v_n p_1 \ldots p_m\right]^T = \begin{bmatrix}\mathbf{u}\\ \mathbf{v}\\ \mathbf{p}\end{bmatrix}$$

For the bilinear velocity-constant pressure and biquadratic velocity-linear pressure elements we have

$$\mathbf{d} = \left[u_1 u_2 u_3 u_4 v_1 v_2 v_3 v_4 p_1\right]^T$$

and

$$\mathbf{d} = \left[u_1 u_2 \ldots u_9 v_1 v_2 \ldots v_9 p_1 p_2 p_3\right]^T$$

respectively.

The mass matrix **M** is defined by

$$\mathbf{M} = \begin{bmatrix}\mathbf{A} & 0 & 0\\ 0 & \mathbf{A} & 0\\ 0 & 0 & 0\end{bmatrix} \qquad \text{Eq. (4-83)}$$

where **A** is an n x n mass matrix given by

$$\left[a_{ij}\right] = \left[\int_\Omega N_i N_j \, d\Omega\right] \qquad \text{Eq. (4-84)}$$

The stiffness matrix **K** contains only the linear part of the operator, i.e., the viscous terms, the pressure terms, and the continuity constraint, and is given by

$$\mathbf{K} = \begin{bmatrix}\mathbf{B} & 0 & \mathbf{C}_x\\ 0 & \mathbf{B} & \mathbf{C}_y\\ -\mathbf{C}_x^T & -\mathbf{C}_y^T & 0\end{bmatrix} \qquad \text{Eq. (4-85)}$$

Here **B** is an n x n matrix defined by

$$\left[b_{ij}\right] = \left[\int_\Omega \frac{1}{\mathrm{Re}}\left(\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y}\right) d\Omega\right] \qquad \text{Eq. (4-86)}$$

and corresponds to the Laplacian operator. The matrices $\mathbf{C}_x$ and $\mathbf{C}_y$ are both n x m and have the form

$$\left[\left(c_x\right)_{ik}\right] = \left[-\int_\Omega \frac{\partial N_i}{\partial x} M_k \, d\Omega\right],\ \left[\left(c_y\right)_{ik}\right] = \left[-\int_\Omega \frac{\partial N_i}{\partial y} M_k \, d\Omega\right] \qquad \text{Eq. (4-87)}$$

The function **F** contains the convective terms and is given by

$$F = \begin{bmatrix} F_x \\ F_y \\ 0 \end{bmatrix} \qquad \text{Eq.(4-88)}$$

with the n x 1 vectors F_x and F_y defined by

$$\left[(f_x)_i\right] = \left[\int_\Omega N_i \left\{ \left(\sum_j N_j u_j\right)\left(\sum_k \frac{\partial N_k}{\partial x} u_k\right) + \left(\sum_j N_j v_j\right)\left(\sum_k \frac{\partial N_k}{\partial y} u_k\right)\right\} d\Omega\right]$$

$$\left[(f_y)_i\right] = \left[\int_\Omega N_i \left\{ \left(\sum_j N_j u_j\right)\left(\sum_k \frac{\partial N_k}{\partial x} v_k\right) + \left(\sum_j N_j v_j\right)\left(\sum_k \frac{\partial N_k}{\partial y} v_k\right)\right\} d\Omega\right]$$

Eq. (4-89)

To calculate steady state flows, a direct nonlinear or a time-dependent solution can be used until the solution no longer changes in time. The second approach is preferred when physical instabilities may occur that can change the mode of circulation. A direct iterative solution will normally not be able to branch off to a new mode that is physically more stable. When a time-dependent algorithm is chosen, there are two basic ways to deal with the nonlinear terms.

1) One method is treating the convective terms explicitly, that is, evaluating the nonlinear terms at each time step using the latest computed values and assembling them into the forcing term in the right-hand-side. This introduces a stability condition where the Courant number must be less than 1 for every element in the mesh. If the evolution of the flow field in time is important, this does not constitute a restriction.

2) A second method is using an unconditionally stable time integrating algorithm combined with a Newton-Raphson iteration within each time step. This allows for the use of larger time steps in order to achieve steady state more rapidly. However, it can become expensive if the time evolution must be calculated accurately.

When time-dependent algorithms, such as discussed in points 1) and 2) above, and bilinear elements are used, it suffices to weight the convective terms in the right-hand-side with the weighting functions

$$w_i = N_i + \frac{\alpha \bar{h}}{2|V|}\left(u\frac{\partial N_i}{\partial x} + v\frac{\partial N_i}{\partial y}\right)$$

where α = coth $\gamma/2 - 2/\gamma$ as before and γ is now the cell Reynolds number given by

$$\gamma = Re|V|\bar{h}$$

The steady state solution to flow over the backward facing step, shown in Fig. 4-22, is shown for Re = 900, based on the entry length H/2. A mesh of 3000 rectangular bilinear elements is used with piecewise constant pressure, in a mixed formulation. The mesh is irregularly spaced for better resolution close to the solid boundaries and at the entry. Referring to Fig. 4-22, L_1 = 3H and L_2 = 19H. The solution was obtained using a backward-implicit time-stepping scheme combined with a Newton-Raphson iteration within each time step and Petrov-Galerkin weighting of the convective terms. The steady state streamfunction is shown in Fig. 4-24a. A second recirculation cell is observed at the top wall that was not present at the low Re flow. The pressure field is shown in Fig. 4-24b.

Flow over a backward facing step has become one of the benchmark problems to test numerical models [4.42], [4.43]. Another very popular problem to test methods is the cavity-driven flow introduced by Burggraf [4.44]. There exists a great amount of information and solutions to this problem in the literature.

In some cases, particularly if the data are not smooth, the calculated pressure field oscillates from one element to the next in what is usually called a “checkerboard” mode. The oscillations occur because the discretization space of the pressure admits functions that are not constant, but for which the discrete divergence operator vanishes.

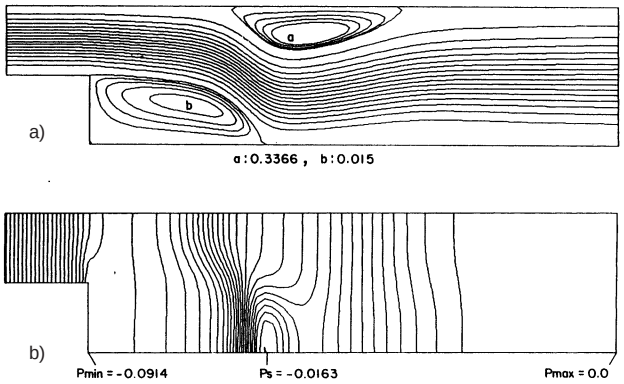


Figure 4-24. Flow over a backward facing step at Re = 900: (a) streamfunction contours and (b) pressure contours.

2. Fractional Step Method

The method of fractional steps has been used extensively, especially in finite difference discretizations. Some selected references are Chorin [4.45], Yanenko [4.46], Schneider et al [4.47], Donea et al [4.48], and Quartapelle [4.49]. The method is designed for the solution of the time-dependent equations and consists in performing the solution in two steps as follows.

First, split the velocity components into two parts as

$$u = u^* + u', \quad v = v^* + v'$$

where u* and v* satisfy the momentum equations without the pressure terms, i.e., using the vector notation for conciseness,

$$\frac{\partial u^*}{\partial t} = -V\cdot\nabla u + \frac{1}{Re}\nabla^2 u$$

$$\frac{\partial v^*}{\partial t} = -V\cdot\nabla v + \frac{1}{Re}\nabla^2 v$$

Eq. (4-90)

Using an explicit forward Euler scheme to solve for u and v in Eq. (4-90),

$$\frac{u^*_{n+1} - u^n}{\Delta t} = -V^n \cdot \nabla u^n + \frac{1}{Re}\nabla^2 u^n$$

$$\frac{v^*_{n+1} - v^n}{\Delta t} = -V^n \cdot \nabla v^n + \frac{1}{Re}\nabla^2 v^n \qquad \text{Eq. (4-91)}$$

The velocity components calculated from Eq. (4-91) will not satisfy the continuity equation. Therefore corrections u′ and v′ need to be computed with the help of the pressure field. To do this, one must first find the pressure field. Differentiating the u component in Eq. (4-90) with respect to x and the v component with respect to y and adding,

$$\nabla^2 p = \frac{\partial}{\partial x}\left(-V \cdot \nabla u + \frac{1}{Re}\nabla^2 u\right) + \frac{\partial}{\partial y}\left(-V \cdot \nabla v + \frac{1}{Re}\nabla^2 v\right) \qquad \text{Eq. (4-92)}$$

Substituting the expressions in Eq. (4-91) into Eq. (4-92),

$$\nabla^2 p^{n+1} = \frac{1}{\Delta t}\left(\frac{\partial u^*_{n+1}}{\partial x} + \frac{\partial v^*_{n+1}}{\partial y}\right) \qquad \text{Eq. (4-93)}$$

which must be solved for the pressure. Once p^{n+1} is known, from the complete discretized equations,

$$\frac{u^{n+1} - u^n}{\Delta t} = -V^n \cdot \nabla u^n + \frac{1}{Re}\nabla^2 u^n - \frac{\partial p^{n+1}}{\partial x} \qquad \text{Eq. (4-94)}$$

$$\frac{v^{n+1} - v^n}{\Delta t} = -V^n \cdot \nabla v^n + \frac{1}{Re}\nabla^2 v^n - \frac{\partial p^{n+1}}{\partial y} \qquad \text{Eq. (4-95)}$$

Substituting Eqs. (4-90) and (4-91)

$$u'_{n+1} = -\Delta t \frac{\partial p^{n+1}}{\partial x}, \qquad v'_{n+1} = -\Delta t \frac{\partial p^{n+1}}{\partial y}$$

The new velocity components $u^{n+1} = u^*_{n+1} + u'_{n+1}$ and $v^{n+1} = v^*_{n+1} + v'_{n+1}$ satisfy Eqs. (4-94) and (4-95) as well as the incompressibility condition.

It is not hard to imagine that many variations of the present methodology can be obtained once this basic form is understood. The pressure may be interpolated using the same shape functions as for the velocity components. This has led some authors to call them “equal-order interpolation” schemes.

The advantage of this method is that only the solution of the Poisson equation for the pressure must be solved implicitly. The disadvantage is that the time step can become extremely small to satisfy stability, and calculations become prohibitively lengthy.

3. Penalty Function Formulation

The basic idea of the penalty method consists in expressing the pressure through the pseudoconstitutive relation

$$p = p_s - \lambda \nabla \cdot V \qquad \text{Eq. (4-96)}$$

in which λ is a large number. Equation (4- 96) is then substituted into the momentum equations,

$$\frac{\partial u}{\partial t} + V \cdot \nabla u = -\frac{\partial p_s}{\partial x} + \lambda \frac{\partial}{\partial x}\left(\nabla \cdot V\right) + \frac{1}{Re}\nabla^2 u \qquad \text{Eq. (4-97)}$$

$$\frac{\partial v}{\partial t} + V \cdot \nabla v = -\frac{\partial p_s}{\partial y} + \lambda \frac{\partial}{\partial y}\left(\nabla \cdot V\right) + \frac{1}{Re}\nabla^2 v \qquad \text{Eq. (4-98)}$$

and the continuity equation is no longer necessary. The discretizations of Eqs. (4-97) and (4-98) result in the solution of a discrete system of equations that involve only the velocity degrees of freedom, which is about 15% smaller for bilinear elements. Furthermore, there is no occurrence of zero diagonal elements of the final linear matrices, although the system will remain ill conditioned due to the large value of the penalty parameter. Hence direct equation solvers are required.

For constant density flows, the static component p_s in Eq. (4-96) is eliminated through a redefinition of the pressure, so that the hydrostatic pressure due to gravity is canceled out. For this reason, the body force terms are not included even though one of these directions might represent the vertical direction aligned with gravity. In this case, the penalty formulation may be written as

$$p = -\lambda \nabla \cdot V \qquad \text{Eq. (4-99)}$$

which is the standard form found in most references. However, this form of the penalty method is incorrect when applied to stratified flows and flows with free surfaces.

The two-dimensional, steady state Stokes equations are written as,

$$\mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = \frac{\partial p}{\partial x}$$

$$\mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) = \frac{\partial p}{\partial y} \qquad \text{Eq. (4-100)}$$

The Galerkin penalty function formulations of Eq. (4-100) are

$$\left[\int_\Omega \mu\left(\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y}\right) u_j \, d\Omega\right] + \left[\int_\Omega \lambda \frac{\partial N_i}{\partial x}\left(\frac{\partial N_j}{\partial x}u_j + \frac{\partial N_j}{\partial y}v_j\right) d\Omega\right] = 0$$

$$\left[\int_\Omega \mu\left(\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y}\right) v_j \, d\Omega\right] + \left[\int_\Omega \lambda \frac{\partial N_i}{\partial y}\left(\frac{\partial N_j}{\partial x}u_j + \frac{\partial N_j}{\partial y}v_j\right) d\Omega\right] = 0 \qquad \text{Eq. (4-101)}$$

where all the line integrals are disgarded since they are irrelevant to this discussion. The final system of linear equations can be written as

$$\left[\mu \mathbf{K}_1 + \lambda \mathbf{K}_2\right] \mathbf{d} = \mathbf{F} \qquad \text{Eq. (4-102)}$$

where the vector **F** is generated from the boundary conditions and, in this case, the vector **d** contains only velocity degrees of freedom. Assume now that the matrix $\mathbf{K}_2$ is nonsingular and that λ is increased more and more in an effort to satisfy incompressibility better. Because μ and $\mathbf{K}_1$ remain constant as λ increases, they become negligible, and the solution to Eq. (4-102) can be written as

$$\mathbf{d} = \frac{1}{\lambda} \mathbf{K}_2^{-1} \mathbf{F} \qquad \text{Eq. (4-103)}$$

Because $\mathbf{K}_2$ and **F** are also constant, $\mathbf{d} \to 0$ as $\lambda \to \infty$. This phenomenon is called "locking" and is directly related to satisfaction (or the lack thereof) of the LBB condition.

The penalty matrix $\mathbf{K}_2$ must be singular. This is achieved using selective reduced integration of the penalty term and a quadrature rule with degree of precision lower than required to guarantee an optimal convergence rate in the matrix $\mathbf{K}_1$. Mathematical techniques to analyze elements to be used with the reduced integration penalty term are discussed in Oden et al [4.50], Carey and Krishnan [4.51], and Idelsohn et al [4.21].

Computational experience shows that penalty calculations must be performed using double-precision 64 bit words. Under these circumstances, a penalty parameter λ between 10^7 and 10^9 will be adequate in most practical situations.

4. Calculation of Pressure

In penalty calculations the pressure P may be recovered regardless of whether P represents the modified pressure or the total pressure. For bilinear interpolation of the velocities,

$$P^e = -\frac{\lambda}{A^e} \iint_e \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) de \qquad \text{Eq. (4-104)}$$

where the superscript e denotes the restriction to the element under consideration and A^e is the element area. In this case, a one-point reduced integration quadrature formula is employed to evaluate Eq. (4-104). The pressures are piecewise constant pressures – the nodal pressures are obtained with the least squares procedure.

5. Open Boundaries

The issue of boundary conditions at outflow boundaries has been treated by many authors [4.52], although in a rather ad hoc way. An analysis performed by Heinrich et al [4.53] points out that pressure boundary conditions should not be applied - only a reference pressure must be prescribed at some point in the domain. Therefore the difficulties at the open boundaries can also be eliminated by removing the pressure from the boundary condition, so that only

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial x} = 0$$

are used. To achieve this, the line integrals

$$\int_\Gamma U p n_x \, d\Gamma \qquad \int_\Gamma V p n_y \, d\Gamma$$

must be discretized and retained in the weighted residuals formulation. If a mixed formulation is being used, these integrals will contribute to the matrices $\mathbf{C}_x$ and $\mathbf{C}_y$ in Eq. (4-87), but no contribution will be added to $\mathbf{C}_x^T$ and $\mathbf{C}_y^T$ in the last row. Thus **K** will become an asymmetric matrix even in the case of Stokes flow.

If a penalty formulation is used, retrieving the integrals is only possible in a time-dependent formulation where the integrals are evaluated explicitly on the right-hand side. To try to include them implicitly by means of the penalty term will result in an ill-conditioned stiffness matrix.

For a similar benchmark problem involving heat fluxes along channel boundaries that has been solved with a variety of existing numerical models, the interested reader should look at the work by Blackwell and Pepper [4.43].

B. Free Surface Flows

Consider the flow of a fluid with a free surface that is subjected to small-amplitude gravity waves. Assume that the flow is isothermal. The fluid layer is two-dimensional with a reference depth h_0, as shown in Fig. 4-25. The deviation of the depth from the reference depth is denoted by h, so that the position of the free surface S(x, y, t) is

$$S = y - h_0 - h = 0 \qquad \text{Eq. (4-105)}$$

Rewrite the governing equations of motion as

$$\nabla \cdot V = 0 \qquad \text{Eq. (4-106)}$$

$$\frac{DV}{Dt} = -\frac{1}{\rho_0} \nabla p + \nu \nabla^2 V + B \qquad \text{Eq. (4-107)}$$

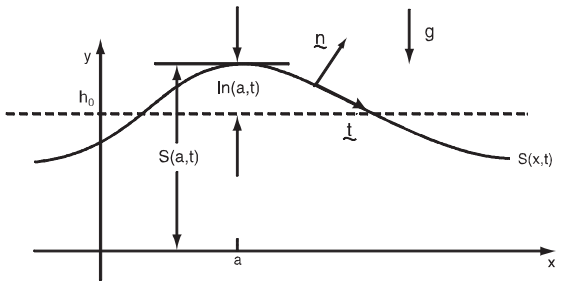


Figure 4-25. Free surface flow in two dimensions.

The boundary conditions at the free surface express continuity of stresses and are given by

$$\sigma_{ij} n_j = \left(-p_a + 2\sigma G\right) n_i \,,$$

$$\sigma_{ij} t_j = 0 \qquad \text{on} \quad S \qquad \text{Eq. (4-108)}$$

where p_a is the pressure at the interface, e.g., atmospheric pressure at a water-air interface, and is assumed to be constant; σ is the surface tension; G the mean surface curvature; **n** the unit outward normal vector and **t** the unit tangent vector, as shown in Fig. 4-25.

A kinematic condition to describe the surface motion is also needed, which is

$$\frac{DS}{Dt} = \frac{\partial S}{\partial t} + V \cdot \nabla S = 0 \qquad \text{Eq. (4-109)}$$

Using Eq. (4-105), this expression becomes

$$\frac{\partial h}{\partial t} + u\frac{\partial h}{\partial x} - v = 0 \qquad \text{Eq. (4-110)}$$

Let $H/y_c = 2/3$, where H is the total head and y_c is called the critical depth. Using $\overline{U} = \sqrt{g y_c}$, the velocity of infinitesimal gravity waves as the characteristic velocity, and y_c as the characteristic length, the relevant nondimensional numbers are the Froude number, Fr, and Reynolds number, Re, defined as

$$Fr = \frac{U^2}{g y_c} \qquad Re = \sqrt{\frac{8g}{27}}\, H^{3/2} / \nu$$

The static pressure p_s becomes

$$p_s = \rho g\left(h + h_0 - y\right) \qquad \text{Eq. (4-111)}$$

Hence, the pressure gradients are

$$\frac{\partial p_s}{\partial x} = \rho g \frac{\partial h}{\partial x}\ , \quad \frac{\partial p_s}{\partial y} = -\rho g \qquad \text{Eq. (4-112)}$$

The solution of free surface problems becomes complicated by the fact that the location of the free surface must be calculated as part of the problem. At every time step the location of the free surface is recalculated so as to satisfy dynamic equilibrium at the end of each time step, following the basic idea of Rushak [4.54] to modify the mesh.

There is much literature on free surface flows in which surface tension is important. Finite element algorithms for such flows have been proposed by, among others, Nickell et al [4.55], Kistler and Scriven [4.56] and Chippada et al [4.57].

C. Flows in Rotating Systems

There is an interesting analogy between convective flows and Taylor-Couette flows. To this purpose, rewrite the governing equations in the form

$$\frac{\partial u_i}{\partial x_i} = 0 \qquad \text{Eq. (4-113)}$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \varepsilon_1 \frac{\partial^2 u_i}{\partial x_j \partial x_j} + f_i \qquad \text{Eq. (4-114)}$$

$$\frac{\partial \theta}{\partial t} + u_j \frac{\partial \theta}{\partial x_j} = \varepsilon_2 \frac{\partial^2 \theta}{\partial x_j \partial x_j} \qquad \text{Eq. (4-115)}$$

In the case of an isothermal flow without body forces, $f_i = 0$ and Eq. (4-114) is either irrelevant or it may model the transport of a species that does not affect the fluid motion. In this case, we set $\varepsilon_1 = 1/Re$. If a Boussinesq fluid is being modeled, Eq. (4-115) becomes the energy equation and θ the temperature. There are two distinct cases:

1) In the case of natural convection the appropriate parameters are the Prandtl and Rayleigh numbers. Set $e_1 = Pr$, $e_2 = 1.0$, $f_1 = 0$, and $f_2 = PrRa\theta$.

2) For free and forced convection, the flow is characterized by the Reynolds, Péclet, and Grashof or Froude numbers. Therefore, set $e_1 = 1/Re$, $e_2 = 1/Pe$, $f_1 = 0$, and $f_2 = \theta/Fr$.

Finally, if we are modeling an isothermal flow in a rotating system (of the Taylor-Couette type), the only relevant parameter is the Taylor number, Ta, given by

$$Ta = \frac{\Omega^2 R d^3}{\nu^2} \qquad \text{Eq. (4-116)}$$

where Ω is the angular velocity, R is the outer radius, and d the characteristic thickness of the fluid layer. In this case set $\varepsilon_1 = \varepsilon_2 = 1.0$, $f_1 = 0$, and $f_2 = Ta\theta^2$. Equation (115) becomes the equation for the circumferential component of the velocity.

In general, a Petrov-Galerkin discretization will be necessary to guarantee stability of the algorithm. Using isoparametric bilinear elements and the penalty formulation with penalty parameter λ, the weak form of Eqs. (4-113)–(4-115) become

$$\int_\Omega N_i\left[\frac{\partial u}{\partial t} + \varepsilon_1\left(\frac{\partial N_i}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial u}{\partial y}\right)\right] d\Omega + R\int_\Omega \lambda \frac{\partial N_i}{\partial x}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) d\Omega$$
$$= \int_\Omega \left(N_i + P_{i1}\right)\left(u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right) d\Omega - \int_{\Gamma_0} N_i p\, n_x\, d\Gamma$$

Eq. (4-117)

$$\int_\Omega N_i\left[\frac{\partial v}{\partial t} + \varepsilon_1\left(\frac{\partial N_i}{\partial x}\frac{\partial v}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial v}{\partial y}\right)\right] d\Omega + R\int_\Omega \lambda \frac{\partial N_i}{\partial y}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) d\Omega$$
$$= \int_\Omega \left[\left(N_i + P_{i1}\right)\left(u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right) + N_i f_2\right] d\Omega - \int_{\Gamma_0} N_i p\, n_y\, d\Gamma$$

Eq. (4-118)

$$\int_{\Omega}\left[N_i\frac{\partial\theta}{\partial t}+\varepsilon_2\left(\frac{\partial N_i}{\partial x}\frac{\partial\theta}{\partial x}+\frac{\partial N_i}{\partial y}\frac{\partial\theta}{\partial y}\right)\right]d\Omega$$

$$=\int_{\Omega}\left(N_i+P_{i2}\right)\left(u\frac{\partial\theta}{\partial x}+v\frac{\partial\theta}{\partial y}\right)d\Omega+\int_{\Gamma_1}N_i q\,d\Gamma$$

Eq. (4-119)

Here N_i denotes the bilinear shape functions, and P_{ij} are the Petrov-Galerkin perturbations applied to the convective terms only. The open portions of the boundary are denoted by Γ_0, and those for which a heat flux q is prescribed, by Γ_2.

The spatial discretization leads to the system of ordinary differential equations

$$\mathbf{M}\dot{\mathbf{d}}+\mathbf{K}\mathbf{d}=\mathbf{F}(\theta)-\mathbf{N}(\mathbf{d}) \qquad \text{Eq. (4-120)}$$

$$\mathbf{C}\dot{\theta}+\mathbf{D}\theta=-\mathbf{G}(\mathbf{d},\theta) \qquad \text{Eq. (4-121)}$$

where **M** and **C** are the mass matrices for the momentum and energy equations, respectively; **d** is the vector of velocity degrees of freedom; **K** contains the contributions of the viscous and penalty terms; **N** is the vector containing the nonlinear convective terms in the momentum equation; **F** contains the contributions of the body forces and the prescribed boundary conditions; **D** is the heat diffusion matrix; and **G** contains the convective terms and contributions from the boundary conditions in the energy equation.

The nonlinear convective terms and the body force terms are calculated explicitly in the right-hand side using the latest available values of the dependent variables. This has the advantage that the system matrices (**M** + γΔt**K**) and (**C** + γΔt**D**) on the left-hand side are constant throughout the calculation, provided the time step remains constant. Therefore the matrices can be factored at the beginning of the calculation using an LU decomposition method. To advance the solution in time after the first time step, it suffices to update the right-hand side and perform a forward and backward substitution, thereby reducing the computational time significantly. Furthermore, these matrices are symmetric, enhancing computational efficiency.

D. Isothermal Flow Past a Circular Cylinder

Flow past a circular cylinder is a well-defined example of a time-dependent flow. When Re exceeds about 60, vortices are shed from the two recirculating cells behind the cylinder. Brooks and Hughes [4.58] performed calculations at Re = 100 based on the cylinder diameter. The domain and mesh used in their work are shown in Fig. 4-26(a-c). The mesh contains 1510 nodes and 1436 isoparametric bilinear elements; the time step was fixed at Δt = 0.03. The simulation was started from zero velocities, and a unit horizontal inlet velocity was applied at t = 0. A symmetric steady state solution is obtained unless the flow is perturbed [4.59]. A perturbation was accomplished by imposing small forces of magnitude 0.0001 to the boundary layer nodes between time t = 54 and t = 58.5. The perturbation effectively destabilized the flow, and vortex shedding began at time *t* = 96. Stream function plots at times *t* = 102 and *t* = 144 are shown in Fig. 4-26(d-f), together with a pressure plot in the neighborhood of the cylinder at time t = 144.

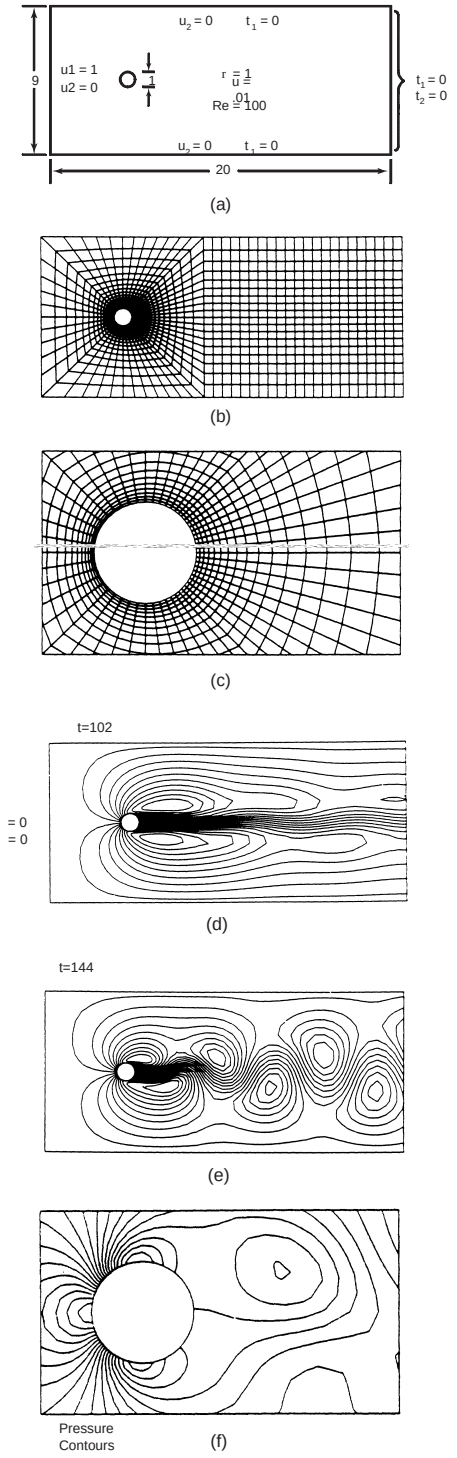


Figure 4-26. Flow past a circular cylinder at Re = 100: (a) Domain and boundary conditions, (b) finite element mesh, (c) detail of the mesh next to the cylinder, (d) streamlines at *t* = 100, (e) streamlines at *t* = 144, and (f) pressure contours close to the cylinder at *t* = 144.