Finite Difference Method

1 Introduction

The finite difference method is the oldest approach to the numerical solution of the equations of fluid dynamics. The fundamental idea is straightforward and implementation is relatively simple. Approximations of various levels of accuracy can be readily computed by the use of different finite difference formula. Consistency, stability and error analysis can be carried out.

The momentum conservation and continuity equations involve not only the various components of velocity but also first and second order partial derivatives of these components. The finite difference method consists in replacing the derivatives in the governing equations by finite difference approximations. While the original equations apply to the infinite of points in the fluid constituting the flow domain, the corresponding finite difference analogues apply only at finite collection of discrete points in a mesh of nodes constructed for the purpose. As a result, the original initial-boundary value problem for a set of non-linear partial differential equations is transformed into a set of coupled, simultaneous, non-linear algebraic equations. Since the resulting set of equations is non-linear, iterative solutions methods must be used to solve them.

2 Finite Difference Approximations

The finite difference method is inspired in elementary calculus. Recall that the derivative of a function $f(x)$ with respect to the argument $\frac{df}{dx}$ is defined as

$$\frac{df}{dx} = \lim_{\Delta x \to 0} \frac{\Delta f}{\Delta x} = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{(x + \Delta x) - x}$$

where the equality between df/dx and $\Delta f/\Delta x$ is exact only in the limiting sense.

The error or difference between the values of df/dx and $\Delta f/\Delta x$ can be estimated from the Taylor series expansion of the function $f(x) = f(x_i)$ in the vicinity of a given point x_i, which is given by

$$f(x + \Delta x) = f(x) + (\Delta x)\frac{df}{dx}|_i + \frac{(\Delta x)^2}{2!}\frac{d^2 f}{dx^2}|_i + ... + \frac{(\Delta x)^n}{n!}\frac{d^n f}{dx^n}|_i$$

Note that the formula predicts the value of the function at the neighboring location using only knowledge of the value of the function and its derivatives at $x = x_i$.

Upon rearrangement, the above yields

$$\frac{f(x + \Delta x) - f(x)}{\Delta x} = \frac{df}{dx}|_i + \frac{\Delta x}{2!}\frac{d^2 f}{dx^2}|_i + ... + \frac{\Delta x^{n-1}}{n!}\frac{d^n f}{dx^n}|_i$$

showing that, when Δx is small, the leading term in the approximation error is of the order of Δx and that the error tends to zero as Δx gets smaller. An identical result is obtained when considering partial derivatives.

Consider now a scalar function of position $\phi(x, y, z)$ in a three dimensional rectangular Cartesian system of coordinates and focus on the variations of ϕ with distance along the $x-$coordinate direction at constant values of y and z. Let x_i be an given point along the $x-$axis and $x_{i-1} = x_i - \Delta x$ and $x_{i+1} = x_i + \Delta x$ two neighboring points. The following truncated approximations to the first partial derivative at $x = x_i$ are readily obtained from the Taylor series by discarding all the error terms starting with the leading term:

$$\frac{\partial \phi}{\partial x}|_i \approx \frac{\phi(x_{i+1}) - \phi(x_i)}{x_{i+1} - x_i} = \frac{\phi_{i+1} - \phi_i}{x_{i+1} - x_i}$$

called the forward difference (FD) formula,

$$\frac{\partial \phi}{\partial x}|_i \approx \frac{\phi(x_i) - \phi(x_{i-1})}{x_i - x_{i-1}} = \frac{\phi_i - \phi_{i-1}}{x_i - x_{i-1}}$$

called the backward difference (BD) formula, and

$$\frac{\partial \phi}{\partial x}|_i \approx \frac{\phi(x_{i+1}) - \phi(x_{i-1})}{x_{i+1} - x_{i-1}} = \frac{\phi_{i+1} - \phi_{i-1}}{x_{i+1} - x_{i-1}}$$

the central difference (CD) formula. One can readily show that the leading error term in the first two formulae is proportional to Δx while in the third formula is proportional to $(\Delta x)^2$.

Second order accurate approximations to the first derivative are useful in deriving finite difference formulae for higher order derivatives. The following second order accurate central difference formulae constitute approximations for the first derivatives at the mid-node locations $x_{i+\frac{1}{2}} = \frac{1}{2}(x_{i+1} + x_i)$ and $x_{i-\frac{1}{2}} = \frac{1}{2}(x_{i1} + x_{i-1})$,

$$\frac{\partial \phi}{\partial x}|_{i+\frac{1}{2}} \approx \frac{\phi_{i+1} - \phi_i}{x_{i+1} - x_i}$$

and

$$\frac{\partial \phi}{\partial x}|_{i-\frac{1}{2}} \approx \frac{\phi_i - \phi_{i-1}}{x_i - x_{i-1}}$$

Higher order accuracy formulae are readily obtained by considering additional neighboring points. For instance, fitting a cubic polynomial to the four uniformly spaced neighboring points $(x_{i-2}, x_{i-1}, x_i, x_{i+1})$ yields the third order accurate approximation

$$\frac{\partial \phi}{\partial x}|_i \approx \frac{2\phi_{i+1} + 3\phi_i - 6\phi_{i-1} + \phi_{i-2}}{6\Delta x}$$

and fitting the five uniformly spaced neighboring points $(x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2})$ with a fourth order polynomial yields the fourth order accurate approximation

$$\frac{\partial \phi}{\partial x}|_i \approx \frac{-\phi_{i+2} + 9\phi_{i+1} - 8\phi_{i-1} + \phi_{i-2}}{12\Delta x}$$

Higher accuracy, compact approximations can be obtained using Pade' schemes which involve values of the derivatives at neighboring points. For example, a Pade'-6 (sixth order accurate) scheme is given by

$$\frac{1}{3}\frac{\partial \phi}{\partial x}|_{i+1} + \frac{\partial \phi}{\partial x}|_i + \frac{1}{3}\frac{\partial \phi}{\partial x}|_{i-1} = \frac{14}{9}\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} + \frac{1}{9}\frac{\phi_{i+2} - \phi_{i-2}}{4\Delta x}$$

which constitutes a tridiagonal system of equations for the values of the derivatives at the three locations (x_{i-1}, x_i, x_{i+1}). Many other possible finite difference formulae exist.

Non-uniformly spaced points are a must in CFD since the characteristics of the flow vary from point to point. In some areas the velocities may change rapidly with distance while in others that will not be the case. One is must then use many mesh points where the flow changes rapidly with position and not so many where the flow changes little.

The use of mesh expansion ratios is common practice in CFD. Consider three neighboring points in a non-uniform grid of points (x_{i-1}, x_i, x_{i+1}). The corresponding mesh spacings are $\Delta x_i = x_i - xi - 1$ and $\Delta x_{i+1} = x_{i+1} - xi$. The mesh expansion ratio r is defined as

$$r = \frac{\Delta x_{i+1}}{\Delta x_i}$$

One can show that the effect of mesh refinement on the truncation error when using non-uniform grid is the same as for a uniform grid, but that for a fixed, given number of grid points, the errors are almost always smaller when using non-uniform spacing.

Finite difference approximations for second derivatives can be readily obtained using the above. For instance

$$\frac{\partial^2 \phi}{\partial x^2}|_i \approx \frac{\frac{\partial \phi}{\partial x}|_{i+\frac{1}{2}} - \frac{\partial \phi}{\partial x}|_{i-\frac{1}{2}}}{\frac{1}{2}(x_{i+1} - x_{i-1})}$$

when the mesh spacing is uniform, this reduces to the well know expression

$$\frac{\partial^2 \phi}{\partial x^2}|_i \approx \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{(\Delta x)^2}$$

Higher order accuracy formulae are readily obtained. For instance fitting the five uniformly spaced neighboring points $(x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2})$ with a fourth order polynomial yields the fourth order accurate approximation

$$\frac{\partial^2\phi}{\partial x^2}|_i \approx \frac{-\phi_{i-1} + 16\phi_{i-1} - 30\phi_i + 16\phi_{i+1} - \phi_{i_2}}{12(\Delta x)^2}$$

3 Finite Difference Approximation of the Diffusion Term

To illustrate the use of the above, consider the diffusion term in the conservation of momentum equations $\frac{\partial}{\partial x}(\Gamma\frac{\partial\phi}{\partial x})$. A very frequently used, second order accurate finite difference approximation of this term at location x_i is given by

$$\frac{\partial}{\partial x}(\Gamma\frac{\partial\phi}{\partial x})|_i \approx \frac{(\Gamma\frac{\partial\phi}{\partial x})|_{i+\frac{1}{2}} - (\Gamma\frac{\partial\phi}{\partial x})|_{i-\frac{1}{2}}}{\frac{1}{2}(x_{i+1} - x_{i-1})} \approx \frac{\Gamma_{i+\frac{1}{2}}(\frac{\phi_{i+1}-\phi_i}{x_{i+1}-x_i}) - \Gamma_{i-\frac{1}{2}}(\frac{\phi_i-\phi_{i-1}}{x_i-x_{i-1}})}{\frac{1}{2}(x_{i+1} - x_{i-1})} =$$

$$= \frac{\Gamma_{i+\frac{1}{2}}(\frac{\phi_{i+1}-\phi_i}{\Delta x}) - \Gamma_{i-\frac{1}{2}}(\frac{\phi_i-\phi_{i-1}}{\Delta x})}{\Delta x} = \Gamma(\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{(\Delta x)^2})$$

4 Boundary Conditions

For fluids that are entirely confined by solid walls the no slip condition is applied at all boundaries. At an inlet, the input velocity must be specified while at an outlet one uses an outflow condition ($d\mathbf{u}/dn = 0$) or sets a reference value for the pressure (say $p = 0$).

For the scalar variable ϕ, its actual values may be specified (BC of Dirichlet type or first kind) or the value of its normal derivative may be specified (BC of Neumann type or 2nd kind) or a linear combination of the above may be specified (BC of Robin type or 3rd kind).

5 Solution of the Algebraic Equations

For purposes of illustrating the essence of the solution techniques employed when working with finite difference methods, consider the boundary value problem consisting of finding an approximation to the function $\phi(x)$ for $x \in [0, L]$ satisfying

$$-\frac{d^2\phi}{dx^2} = f(x)$$

where $f(x)$ is a given function and subject to

$$\phi(0) = 0$$
$$\phi(1) = 0$$

Introducing a mesh of uniformly spaced nodes $i = 1, 2, ..., N$, with spacing Δx the finite difference analogue of the above problem for node i is readily obtained as

$$\frac{-\phi_{i-1} + 2\phi_i - \phi_{i+1}}{(\Delta x)^2} = f(x_i)$$

or rearranging

$$-\phi_{i-1} + 2\phi_i - \phi_{i+1} = f(x_i(\Delta x)^2$$

The complete set of equations for all nodes in the mesh constitutes a system of coupled, simultaneous, linear algebraic equations. Note that the algebraic equation system obtained has a very peculiar form in which every equation involves only three neighboring nodes. Algebraic systems of equations with this type of structure are called Tri-Diagonal since when the equations are written in matrix form, non-zero coefficients appear only along the main diagonal and the first super and sub diagonal rows, with the terms appearing along the diagonal being always numerically larger that those elsewhere.

A very efficient direct (i.e. non-iterative) method exist for their solution (e.g. LU decomposition; Tri-Diagonal Matrix Algorithm (TDMA). In this method,the original matrix is first manipulated row by row in order to obtain an upper triangular matrix (i.e. non-zero coefficients only at and above the diagonal); this is the triangularization step. The second step is simply one of back substitution starting with the last equation in the set and continuing until the first one is reached. Even though, multidimensionality and the non-linear terms in the flow equations make the application of direct methods impractical in general, the TDMA procedure is of great importance in CFD where it is used as linchpin in iterative solution procedures.

In an iterative method one assumes an initial guess for the solution ϕ_i^{old} and computes an improved guess ϕ_i^{new} using the discretization formula. The process is then repeated until the computed values at all grid points change little with every subsequent iteration (convergence of the iterations). Typically, at every iteration once searches the entire mesh for the largest relative difference between iterates, i.e.

$$\epsilon_{MAX} = MAX(\epsilon_{i,j}) = MAX(\frac{|\phi_i^{new} - \phi_i^{old}|}{|\phi_{ref}^{old}|})$$

where ϕ_{ref}^{old} is a suitably selected reference value. Iterations stop once a predetermined level of tolerance is reached, i.e.

$$\epsilon_{MAX} \leq \epsilon_{TOL}$$

The iteration procedure is best illustrated by an example. Consider again the problem above. Using the finite difference method, the following discrete analogue was obtained

$$-\phi_{i-1} + 2\phi_i - \phi_{i+1} = f(x_i(\Delta x)^2$$

From this, different iterations schemes can be readily constructed.

In the Jacobi iteration procedure, the iteration formula is simply obtained by solving the above for ϕ_i

$$\phi_i^{new} = \frac{1}{2}[\phi_{i-1}^{old} + \phi_{i+1}^{old} + f(x_i)(\Delta x)^2]$$

were the superscript *new* is used to denote the improved guess. An improved guess for all nodes in the FD mesh can be obtained by visiting the nodes sequentially and calculating the new guess with the formula above.

In the Gauss-Seidel iteration procedure one takes advantage of the fact that nodes are always visited in sequence (say, from left to right) and uses the latest values stored in the memory of the computer, as soon as these become available. The iteration formula in this case is

$$\phi_i^{new} = \frac{1}{2}[\phi_{i-1}^{new} + \phi_{i+1}^{old} + f(x_i)(\Delta x)^2]$$

In the Successive Over-relaxation (SOR) iteration procedure one improves on the G-S procedure by relaxing the value of the iterated variable. The iteration formula in this case is

$$\phi_i^{new} = \phi_i^{old} + \frac{\omega}{2}[\phi_{i+1}^{new} + \phi_{i+1}^{old} - 2\phi_i^{old} + f(x_i)(\Delta x)^2]$$

where ω $(1 < \omega < 2)$ is the relaxation factor. Note that the SOR formula reduces to the Gauss-Seidel scheme when $\omega = 1$.

6 The Convection-Diffusion Equation and Upwinding

Consider the following one-dimensional convection-diffusion equation for the transported quantity $\phi(x)$

$$\rho u \frac{d\phi}{dx} = \Gamma \frac{d^2\phi}{dx^2}$$

for $x \in [0, L]$ subject to suitable boundary conditions such as

$$\phi(0) = 1$$
$$\phi(L) = 1$$

where ρ is the density of the fluid, $u > 0$ is the velocity and Γ the diffusivity of ϕ in the fluid, and all are assumed constant.

Using the method of finite differences and central differencing on an uniform mesh (spacing Δx), consisting of N nodes located at $x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_N$ yields the following discrete analogue

$$\rho u(\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}) = \Gamma(\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{(\Delta x)^2})$$

This is again a tridiagonal system of linear algebraic equations. The effect of convection appears as an additional term in the equations for the coefficients of ϕ_{i-1} and ϕ_{i+1} and the equations can be solved as before, using the TDMA method or some iterative procedure.

The above scheme works fine as long as the Peclet number of the system, defined as

$$Pe = \frac{\rho u L}{\Gamma}$$

and representing the ratio of convective to diffusive transports, is small (say < 1).

When the value of Pe increases, ϕ varies rapidly in the vicinity of the downstream boundary and the scheme above can result in meaningless oscillations in the computed values of ϕ. A simple remedy for this problem is the technique called upwinding. This consists in taking into account the direction of the flow when discretizing the convective term. One should expect that, at large flow velocities, the value of ϕ at any given location should be influenced most by the value of ϕ at the nodal location immmediately upstream of the flow, i.e. for the above case with $u > 0$,

$$\rho u (\frac{\phi_i - \phi_{i-1}}{\Delta x}) = \Gamma (\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{(\Delta x)^2})$$

If instead, $u < 0$, the corresponding formula would be

$$\rho u (\frac{\phi_{i+1} - \phi_i}{\Delta x}) = \Gamma (\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{(\Delta x)^2})$$

Upwinding prevents the development of oscillations. However, since the finite difference is one sided, it is only first order accurate and it has the effect of introducing a non-physical diffusive effect in the solution called false diffusion. Discretization schemes of higher order of accuracy have been developed based on the notion of upwinding (e.g. the exponential scheme).